Chapter 10
Different Views on the Finger—Score-Level Fusion in Multi-Perspective Finger Vein Recognition

Bernhard Prommegger, Christof Kauba and Andreas Uhl

Abstract In finger vein recognition, the palmar view of the finger is used almost exclusively, with some exceptions where the dorsal view is utilised. Only little attention has been paid to all other views around the finger's longitudinal axis. We established a multi-perspective finger vein dataset comprising of views all around the finger's longitudinal axis, captured using our self-developed rotating multi-perspective finger vein capture device. The performance of the single views is evaluated using common finger vein recognition algorithms. Based on these single view scores, several score-level fusion experiments involving different fusion strategies are carried out in order to determine the best performing set of views and feature extraction methods to be fused in terms of recognition accuracy while minimising the number of views involved. Our experimental results show that the recognition performance can be significantly improved over the best performing single view one with as few as two views and two-feature extraction methods involved.

Keywords Finger vein recognition · Multi-perspective fusion · Biometric fusion · Score-level fusion · Multi-algorithm fusion · Multi-perspective finger vein capture device · Finger vein dataset

B. Prommegger (✉) · C. Kauba · A. Uhl
Department of Computer Sciences, University of Salzburg,
Jakob-Haringer-Str. 2, 5020 Salzburg, Austria
e-mail: bprommeg@cs.sbg.ac.at

C. Kauba
e-mail: ckauba@cs.sbg.ac.at

A. Uhl
e-mail: uhl@cs.sbg.ac.at

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_10

10.1 Introduction

Finger vein recognition as one representative of vascular pattern biometrics deals with the vascular pattern inside the fingers of a human. Since one of the first mentions of finger veins as a biometric trait in academia by Kono [1] in 2000, they have received much attention not only from academia but also from industry. Commercial off-the-shelf (COTS) finger vein capture devices, as well as most research papers solely, use the palmar (front side of the finger) view in combination with light transmission (the light source and the image sensor are placed on opposite sides of the finger) as illumination source. Multi-perspective finger vein recognition deals with two or more arbitrary perspectives around the finger's longitudinal axis. Despite the advantages of multi-perspective finger vein biometrics over single view ones, these additional perspectives have not got much attention so far. Moreover, there is no publicly available multi-perspective finger vein dataset yet.

This chapter is based on our previous work [2] where we designed a novel, multi-perspective finger vein capture device in order to establish the first multi-perspective finger vein data set. This dataset comprises of images captured all around the finger's longitudinal axis in 1° steps. Based on this dataset, each of the different views has been evaluated individually and some simple fusion experiments have been conducted. The main focus of this chapter is on the fusion of multiple perspectives and feature extraction methods in order to determine the best performing combination in terms of recognition accuracy by employing a more advanced multi-sample score-level fusion scheme as well as by applying further fusion strategies in terms of view and feature combinations. We analyse all possible pairs and triples of perspectives and all possible combinations of the used feature extraction methods. In addition, we combine the best results of our multi-perspective and multi-algorithm fusion experiments to one single combined fusion. Our main goal is to minimise the number of views and feature extraction methods involved, while maximising the recognition accuracy. A typical multi-perspective finger vein capture device contains one image sensor and one light source situated at the right position per desired view. The more views are to be captured, the more camera and illumination modules have to be equipped, thus increasing the production costs, the complexity and the overall size of the finger vein capture device. If the number of desired perspectives is further increased, the construction of a suitable capture device is no longer feasible without the need of rotating parts. Our current multi-perspective finger vein capture device is such a rotating device, making it more susceptible to malfunctions and external influences than a capture device containing no rotating parts. Moreover, the capturing time is increased as the capture device has to rotate all around the finger. Hence, it is beneficial to reduce the number of different views to be captured to a minimum in order to reduce the complexity and production costs of the biometric capture device and to avoid the need for a rotating device while still preserving the advantages of a multi-perspective capture device.

The rest of this chapter is structured as follows: Sect. 10.2 starts with a description of multi-perspective finger vein biometrics including related work regarding other

views than the palmar and dorsal one in finger vein recognition. Our multi-perspective finger vein capture device design is described in Sect. 10.3. Section 10.4 introduces our multi-perspective finger vein dataset captured with the aforementioned device. Section 10.5 gives an overview of biometric fusion in general followed by related work on biometric fusion in finger vein recognition. Section 10.6 explains our experimental set-up, including the finger vein recognition tool chain as well as the fusion framework we utilised and lists the experimental results, followed by a results discussion. Section 10.7 concludes this paper an gives and outlook on future work.

10.2 Multi-perspective Finger Vein Biometrics

The majority of the available finger vein recognition schemes as well as all available COTS finger vein capture devices deal with the palmar (also called ventral) view of the finger. There are only some exceptions where the dorsal view is used. Raghavendra and Busch [3] proposed the first dorsal finger vein acquisition and a complete recognition tool chain including several different feature extraction schemes. In the scope of the PROTECT project (http://www.projectprotect.eu), we acquired the first publicly available dorsal finger vein dataset [4] using the predecessor of our open-source finger vein capture device. In [5], we established a larger dorsal finger vein dataset captured using both of our proposed open-source finger vein capture devices, which design is decribed in Chap. 3 of this book [6].

There are more views around the finger than the palmar and dorsal one that can be captured. A single finger is an elliptical cylinder-shaped object, hence, there are all possible views around its longitudinal axis (360° of rotation) available. Multi-perspective finger vein recognition describes the use of two or more of these perspectives around the finger's longitudinal axis. Multi-perspective finger vein recognition has several advantages over the single perspective one: The vein patterns of the palmar and dorsal view as well as of the perpendicular views are independent from each other [7]. By fusing more than one perspective that is independent enough from each other (i.e. the rotation angle between the single perspectives has to differ enough for the perspectives to be independent of each other), the overall recognition performance can be increased easily. Tome et al. [8, 9] showed that finger vein and hand vein recognition systems are susceptible to a simple type of presentation attack. By using a paper printout of the vein pattern, they were able to successfully spoof several finger vein capture devices. This paper printout is a flat, 2D representation of the vein pattern. If a biometric capture device takes finger vein images from different perspectives, such simple 2D printout attack finger vein presentation will not be identified as bona fide finger vein presentation. Thus, a multi-perspective finger vein capture device is successfully able to prevent this kind of presentation attack. However, multi-perspective finger vein recognition bears some disadvantages too: The biometric capture devices get more complex, either more than one camera and illumination module are needed, or the capture device has to be build in a rotating manner. This leads to higher production costs of multi-perspective capture devices

and especially rotating capture devices are more error prone due to the moving parts. Another disadvantage is the bigger size of a multi-perspective capture device compared to single perspective ones. The multiple image sensors/illuminator modules or the rotating parts need more space than just a single image sensor in combination with one illumination module.

Lu et al. [10] proposed a multi-perspective finger vein recognition system using two cameras. The cameras are placed at an angle of 60° next to each other, each camera is located 30° apart from the palmar view. They applied feature—as well as score-level fusion using the two views captured simultaneously by the two cameras and were able to improve the recognition performance of the single view ones. Zhang et al. [11] employed a binocular stereoscopic vision device to do 3D point cloud matching of hand veins and knuckle shape. Their capture device set-up consist of two cameras, placed in a relative position of about 45° next to each other, each one equipped with an NIR-pass filter. There is only a single light transmission illuminator placed underneath the palm of the hand. The 3D point clouds are generated by extracting information from the edges of the hand veins and knuckle shapes and then compared utilising a kernel correlation method, especially designed for unstructured 3D point clouds. The authors claim that their proposed method is faster and more accurate compared to 2D vein recognition schemes. In [12] the authors propose a 3D hand vein capturing system based on a rotating platform and a fixed NIR camera. The camera is located above the hand, the hand is put on a handle with an integrated light transmission illuminator. This handle is mounted on a rotating plate. Then the plate rotates around the z-axis. However, the degree of rotation is limited due to the limited movement of the hand in this position. A 3D point cloud is generated from the single view images and matched using kernel correlation. This should help to overcome hand registration and posture change problems present in hand vein recognition if only 2D vein patterns/images are available.

Nevertheless, true multi-perspective finger vein recognition (evaluating more than two different views around the finger) has not been investigated so far, except for our previous work [2]. One reason herefore might be the lack of available multi-perspective finger vein datasets. In order to acquire such a dataset a suitable biometric capture device, able to capture the different views to be acquired, is essential. Capturing these additional perspectives could be done by utilising either a COTS capture device or one of the capture devices proposed in other works by simply turning the finger around its longitudinal axis. However, it is difficult to position the finger in the correct rotational angle. Thus, rotating the finger itself implies the disadvantage of an inaccurate rotation angle and deviations in the rotation angle across different iterations, leading to a low repeatability and a low quality dataset. In order to acquire a suitable multi-perspective finger vein dataset comprising of images captured in several, defined perspectives, either a biometric capture device comprising of several cameras and illumination modules, able to capture more than one view simultaneously, or a rotating biometric capture device able to capture these views consecutively, is necessary. If only a limited number of perspectives are involved, a suitable biometric capture device can be built without any rotating parts, just by equipping an individual image sensor and an associated illumination module per desired

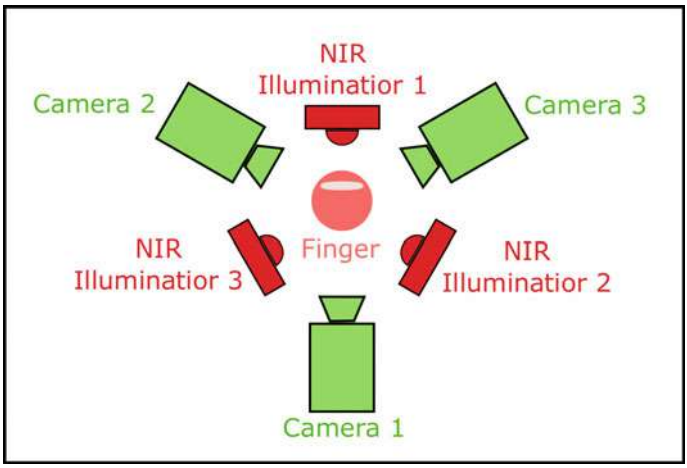


Fig. 10.1 Multi-perspective finger vein set-up exhibiting three different perspectives based on three image sensors and three illuminator modules

view (an example with three different views is shown in Fig. 10.1). The illumination intensity has to be adjusted per view as the path to penetrate the finger is different for each individual view, requiring a stronger or weaker illumination depending on the distance. If more perspectives are desired, rotating the capture device around the finger while the finger remains in a fixed position during the acquisition process is the only feasible option.

The design and construction of a practicable biometric capture device is a complex task. Furthermore, the actual data acquisition is a tedious and time-consuming work. In our previous paper [2], we proposed a rotating multi-perspective finger vein capture device that is able to capture the finger all around its longitudinal axis (360°). We established a multi-perspective finger vein dataset consisting of 252 individual fingers. Based on this dataset, we evaluated the different views around the finger in 5° steps and concluded that the palmar followed by the dorsal one achieve the best single view recognition performance. Moreover, we applied a simple score-level fusion strategy and showed that the recognition performance can be improved by fusing more than one view. This chapter is an extension of our previous work. Based on our proposed multi-perspective finger vein capture device, we refine and extend our previous results by the following:

- Improving the recognition tool chain to improve the single view results, especially the ROI extraction and by including a new recognition scheme proposed by Matsuda et al. [13].
- Employing an advanced score-level fusion framework (BOSARIS [14]).
- Exploring different fusion strategies in terms of which views to include in the fusion.
- Evaluating multi-algorithm fusion per view (fusion is done at score level).

- Combining multi-perspective and multi-algorithm fusion.

The purpose of our evaluations is to maximise the recognition performance while minimising the number of single views involved. If only a limited number of views is involved, the capture device can be built without the need for any rotating parts just by equipping an individual image sensors and an illumination modules per desired view. A biometric capture device which relies on rotating parts is more error prone and more susceptible to external influences, the rotation speed can vary due to increased friction or it can be completely blocked if the finger is not properly inserted. The rotating parts exhibit a higher wear than non-moving parts and are thus more prone to failures. Moreover, the acquisition time of a rotating capture device is higher compared to a non-rotating one as the device needs to rotate around the finger in order to capture the different views. Furthermore, a capturing device exhibiting a closed box design, where the capture subject has to put his finger into a "black hole" poses psychological disadvantages and leads to discomfort. Hence, in practical applications of multi-perspective finger vein biometrics only a capture device built in a non-rotating and open manner is feasible. Consequently, we aim to identify the best combination of two or three views to include in the fusion in order to build such a multi-perspective finger vein capture device based on fixed, non-moving parts only. Figure 10.1 shows the schematic principle of such a capture device for three perspectives: it consists of three independent image capturing pairs, each consisting of its own NIR illumination module and NIR camera.

10.3 Multi-perspective Finger Vein Capture Device

In order to acquire a multi-perspective finger vein dataset, we designed a custom finger vein capture device tailored to this purpose. For more details on the general principle of a finger vein scanner and the vascular pattern recognition basics, the interested reader is referred to our open finger vein scanner chapter [6] and the introductory chapter [15] of this book, respectively. Our multi-perspective finger vein capture device is able to capture images from all around the finger's longitudinal axis (360°). An illustration of the unwrapped finger vein capture device with all its parts labelled can be seen in Fig. 10.2. Its outside dimensions (of the aluminium frame including the rotating part) are $258 \times 325 \times 455$ mm (width $\times$ height $\times$ depth). The rotating part (rotator) has a diameter of 380 mm. The device consists of an aluminium frame, where the rotation motor and the control board are located and a rotator, which rotates around the finger. The rotating part is connected to a stepping motor by two cogwheels. These cogwheels have a gear ratio of 1:5/3 (motor to rotor). The stepping motor (SY42STH47-1684A [16]) which drives the rotator has 200 steps per full rotation (1.8° per single step). We use a micro-stepping of $1/16$, thus one step corresponds to 0.0675°. Hence, it is possible to capture a maximum of 5333 different perspectives of the finger. Located on the right side of the device is the image sensor, an IDS Imaging UI-1240ML-NIR industrial NIR-enhanced camera

[17]. It has a max. resolution of 1280 × 1024 pixels, a max. frame rate of 25 fps and is equipped with a Fujiflim HF9HA-1b 9 mm 2/3" wide-angle lens [18]. To reduce the influence of ambient light, an additional NIR long-pass filter (MIDOPT LP780 [19], with a cut-off wavelength of about 750 nm and a useful range of 780–1000 nm) is mounted on top of the camera lens. The illumination module is located on the opposite side of the image sensor (the left side in Fig. 10.2). Our multi-perspective finger vein capture device is based on the light transmission principle. Instead of typical NIR LEDs the illumination module consists of five NIR laser modules with a peak emission wavelength of 808 nm placed in a strip. Laser diodes have several advantages over LEDs, especially, if the finger is not placed directly on top of the illumination module as mentioned in Chapter [6]. Due to the rotating principle of the biometric capture device, it is not possible for the finger to touch the illumination module, which prevents the use of LEDs without impacting the image quality. Each laser module consists of a NIR laser diode, a control PCB for the laser diode and a housing with a focus-adjustable lens. The plane of focus of the laser modules is set at the axis of rotation where the finger is placed, leading to the highest possible amount of illumination at the position of the finger. Each of the laser modules can be brightness controlled separately (by adjusting the operating current) and independently, enabling a uniform illumination along the whole finger. The finger is put into the capture device at its axis of rotation (in the centre of the image in Fig. 10.2). A fingertip stabiliser (a custom 3D printed part which inside is shaped like the outside of a fingertip) is located at the inside bottom of the rotating part and a height-adjustable finger trunk stabiliser, which is basically a wooden plate with a hole in the middle is located above the rotating part. These finger stabilisers help to reduce finger movements during one acquisition run to a minimum. The finger is put into the capture device so that its tip is inside the fingertip stabiliser, pushing the height-adjustable plate down. Afterwards, this individual finger height is fixed using four screws on the top of the scanner and remains fixed until a new finger is to be captured. All parts except the stepping motor, the camera including the lens and NIR long-pass filter) are self-designed and manufactured by ourselves, including several 3D printed parts, the wooden housing of the rotating part, the housing of the control board, the control board itself and the aluminium frame.

The acquisition process is semi-automated. At first, the subject has to put the finger into the device. Then the height of the finger trunk stabiliser plate has to be adjusted and the operator initiates one capturing run (360° around the finger's longitudinal axis), starting the automated part of the acquisition process.

During this automated data acquisition part, the illumination for each laser module is set automatically by the help of an automated brightness control algorithm. This algorithm tries to achieve a sufficient and uniform illumination along the finger in order to obtain an optimal image contrast. It evaluates the average grey level of the image area around the centre of each laser module i ($GL^i_{current}$) and compares this value to a predefined target grey level (GL^i_{target}). If there is a deviation between these two values, the operating current of the corresponding laser module is adjusted: $I^i_{corr} = \frac{GL^i_{target} - GL^i_{current}}{GL_{\max}} \cdot \frac{I_{\max}}{2 \cdot n}$, where $GL_{\max}$ is the maximum grey value (255 for 8 bit

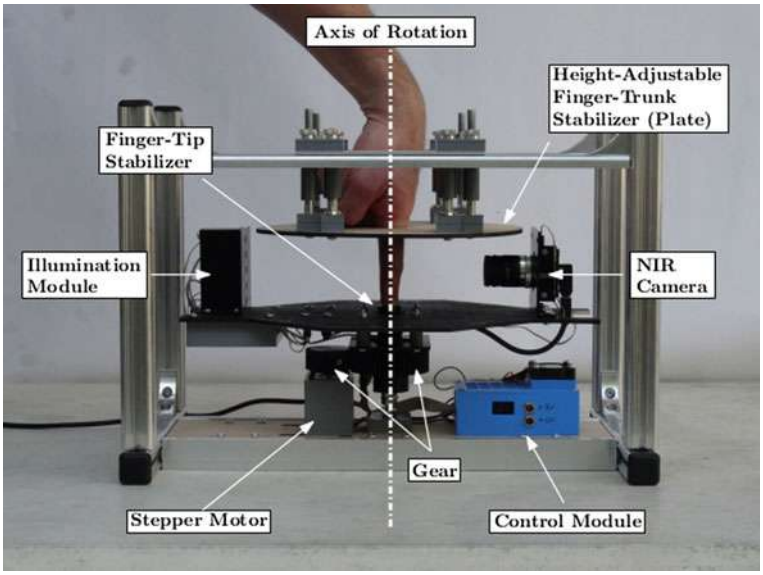


Fig. 10.2 Self-designed multi-perspective finger vein capture device (image originally published in [2], ©2018 IEEE)

images) and n is the number of the current iteration. Initially, all laser modules are set to half of their maximum operating current $I_{\max}$ (corresponding to its maximum intensity). The algorithm finishes in at most $\log_2(I_{\max})$ steps.

After the optimal intensity level for each laser module is set, the video sequence recording is started. The rotator starts to rotate around the finger and an indicator LED is turned on to synchronise the video stream. The rotation is stopped when the rotator reaches its start position again and at this point the indicator LED is turned off. A few frames later the video sequence recording is stopped too. The videos are recorded in the MP4 container format using the MJPG video codec with a frame rate of 15 fps and YUV colour space. The speed of the rotation and the video frame rate are synchronised such that a defined resolution (in degree) of images per full rotation (video frames) is met and the desired degree steps can later be extracted from single, individual frames without the need for temporal interpolation. The set illumination intensity remains the same for the whole capturing run until all perspectives are captured. This ensures the compatibility and comparability of the single, individual perspectives to each other. The different projections in 1° steps corresponding to single video frames are then extracted out of the video sequence. The capture device's indicator LED is utilised to synchronise the video frames with the beginning and the end of the rotation. In theory, there should be 361 images per full rotation run (0° and 360° is captured separately). Due to slight variations in the rotation speed and the video frame rate, there are between 357 and 362 frames instead of 361. Thus, it

became necessary to map the frame with the minimum deviation from the desired rotational angle to the corresponding perspective, resulting in a maximum deviation of 0.5° from the desired rotation angle.

10.4 Multi-perspective Finger Vein Dataset

With the help of our self-designed multi-perspective finger vein capture device, we established a multi-perspective finger vein dataset in order to be able to conduct our multi-perspective score-level fusion experiments. This dataset currently consists of 63 subjects, 4 fingers per subject (index and middle finger of the left and right hand) and 5 runs per finger. The thumb and the pinky finger were not included as they are too short compared to the index and middle. The ring finger was skipped as well as it turned out to be too uncomfortable for the subjects to put it in the capture device for the whole capturing process. The finger was removed and inserted in the device again after each run. During each run, a video sequence of a full 360° rotation with a target resolution of 1° (each frame corresponds to a 1° step) is captured. Figure 10.3 shows the capture device during the data acquisition process. The acquisition process takes approximately 45 s per capture attempt, hence it takes about 15 min to capture a single subject, including all four fingers, 5 runs per finger. The whole dataset consists of $63 \times 4 \times 5 \times 361 = 454{,}860$ images in total. The extracted video frames have a resolution of 1024×1280 pixels and are 8-bit greyscale images stored in png format.

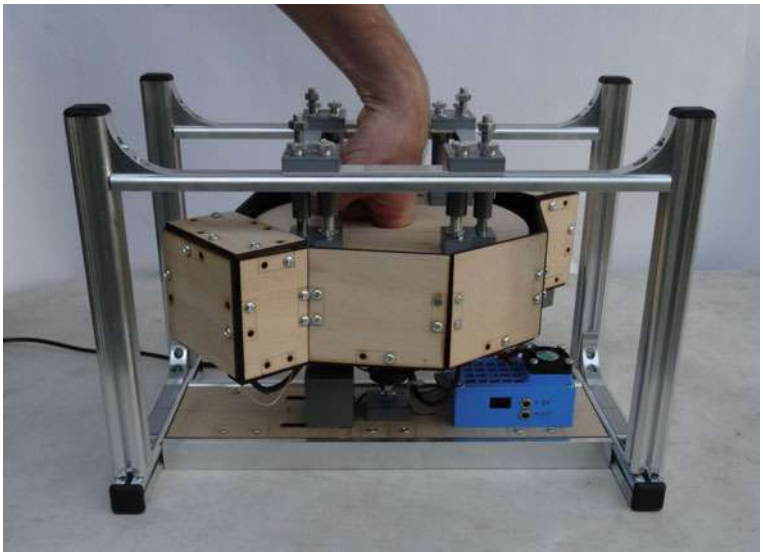


Fig. 10.3 Data acquisition with the multi-perspective finger vein capture device (image originally published in [2], ©2018 IEEE)

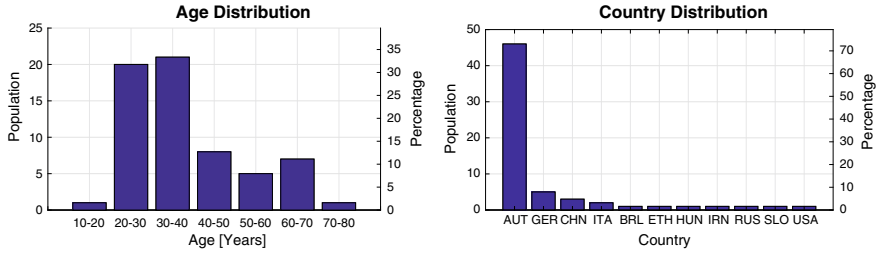


Fig. 10.4 Age (left, image originally published in [2], ©2018 IEEE) and country of origin distribution (right) for the multi-perspective finger vein dataset

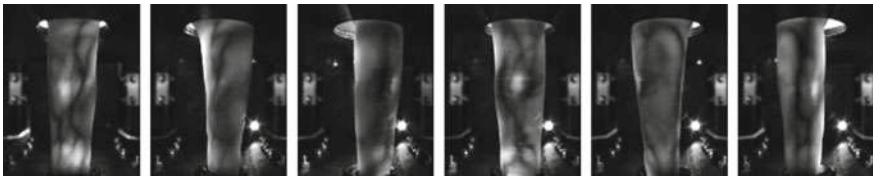


Fig. 10.5 Multi-perspective finger vein dataset example images, from left to right: 0°, 60°, 120°, 180°, 240°, 300° (image originally published in [2], ©2018 IEEE)

The finger is always located in the centre area of the image, thus the images are then cropped to 650 × 1280 pixels to retain the usable finger area only. Figure 10.5 shows some example images in different perspectives from 0° to 300°. It can be clearly seen that the visible vein lines vary among the different perspectives. The black part at the centre top area in the images results from the finger trunk stabilisation plate, which is pushed in further or less depending on the length of the finger.

The gender distribution of the 63 subjects is almost balanced with 27 (42.7%) female and 36 (57.3%) male subjects. The subjects represent a good cross section among all different age groups, as the age distribution, depicted in Fig. 10.4 left, shows. There is only a slight overhang among the 20–40 year old subjects. The youngest subject was 18 and the oldest one 79 years old. The subjects are from 11 different countries (Austria, Brazil, China, Ethiopia, Hungary, Iran, Italy, Russia, Slovenia, USA) while the majority of subjects are white Europeans (73%). The origin country distribution is depicted in Fig. 10.4 right. The dataset is available for research purposes and can be downloaded at http://wavelab.at/sources/PLUSVein-FR/.

10.5 Biometric Fusion

Like every typical biometric recognition system, a finger vein recognition system consists of five steps/modules: image acquisition, preprocessing, feature extraction, comparison and the final decision. This recognition tool chain is depicted in Fig. 10.6.

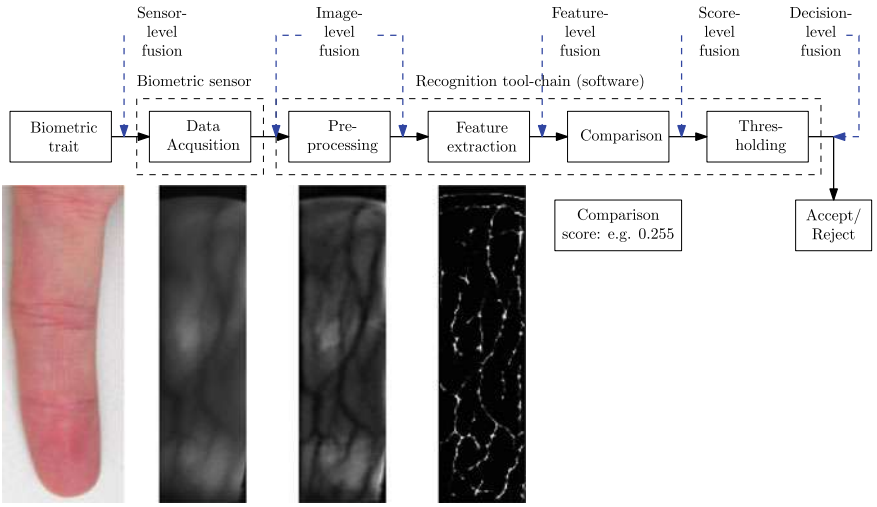


Fig. 10.6 Basic components of a biometric recognition system including the different levels of fusion by taking the example of finger veins (second row)

There are two modes, enrolment and authentication. Authentication includes both, verification as well as identification. During enrolment one or several finger vein images are captured and the extracted biometric templates are stored in a database. During authentication a new template is extracted from a newly captured image and compared against one or more templates stored in the database. The result is a comparison score. Finally the decision module outputs for the capture subject an "accept" or "reject" depending on the evaluation of the comparison score against a threshold.

According to the ISO/IEC TR 24722:2015 standard [20], biometric fusion can be regarded as a combination of information from multiple sources, i.e. sensors, characteristic types, algorithms, instances or presentations in order to improve the overall system's performance and to increase the systems robustness.[1] Biometric fusion can be categorised according to the level of fusion and the origin of input data. The different levels of fusion correspond to the components of a biometric recognition system:

- Sensor-level fusion: is also called multisensorial fusion and describes using multiple sensors for capturing samples of one biometric instance [20]. This can either be done by the sensor itself or during the biometric processing chain. An example of sensor-level fusion are finger vein images that have been captured using different wavelength of near-infrared light and fused by merging the different wavelength bands to obtain one single output image. This can be done by a single biomet-

[1]Recognition performance is just one aspect. PAD performance (robustness against presentation attacks) is another aspect to keep in mind.

ric capture device. Another example is the acquisition and fusion of fingerprint images captured using optical, electrostatic and acoustic sensors.

- Image-level fusion: during data acquisition, the biometric capture device itself might be able to capture multiple samples of the same biometric trait and combine those samples to a single output sample. Image-level fusion corresponds to fusing several images captured from the same biometric trait but not necessarily within the sensor device. Image-level fusion can also be applied after preprocessing so the input to the fusion module is the preprocessed images. One example of image-level fusion is a finger vein capture device that captures more than one finger simultaneously and combines the images from the individual fingers into a single output image, which is also called multi-instance.
- Feature-level fusion: during template creation, several meaningful features, describing the biometric trait's properties, are extracted from the preprocessed images and stored in a feature vector, commonly denoted as biometric template. Feature-level fusion combines several such feature vectors to form a new, higher dimensional feature vector which should represent a subject's biometric traits in a different and more discriminant way. Dimensionality reduction methods are beneficial in combination with feature-level fusion to extract the most significant and discriminative features and to save storage space.
- Score-level fusion: during the comparison step, two templates are compared against each other and a similarity or dissimilarity score is calculated. Score-level fusion combines two or more of those scores into a new, single score. The input scores can originate from different comparison modules. They should either be compatible with each other (e.g. all are similarity scores exhibiting the same range of possible values) or else a score normalisation technique has to be applied during the fusion.
- Decision-level fusion: the output of the decision module is a binary one, which can be interpreted as match/non-match or accept/reject. Decision-level fusion combines two or more of these binary output decisions to a single output one. Usually, majority of voting schemes are employed at decision-level fusion. Note that at the decision level, the least information is available (only a binary decision), compared to the other levels of fusion.

Regarding the origin of the input data, biometric fusion can be categorised into:

- Multi-modal fusion: multiple different types of biometric traits from the same subject is fused together. A popular example is the fusion of information from fingerprints and finger veins or iris and periocular.
- Multi-instance fusion: multiple instances of the same type of biometric trait are fused together. For example, several finger vein images from different fingers of the same subject or information from both irises of one subject are fused together.
- Multi-presentation fusion: multiple samples of the same instance of biometric trait is captured and fused, e.g. several finger veins of the same finger is captured and fused together.

- Multi-algorithmic fusion: multiple feature representations are generated using the same input data, e.g. several different finger vein features are extracted with different algorithms from the same input image and fused together.

There is no direct dependency between the origin of the input data and the level of fusion that is employed.

10.5.1 Fusion in Finger Vein Recognition

This subsection provides an overview of related work in biometric fusion involving finger veins. The first subsection discusses several single modality fusion approaches. The second subsection lists multi-modality fusion approaches which include finger veins among other biometric traits.

10.5.1.1 Single Modality (Finger Vein Only) Fusion

Table 10.1 gives an overview of related work on single modality fusion in finger vein recognition, i.e. only data from finger veins is utilised during fusion at different levels. The table lists the level of fusion applied, the origin of the input data to the fusion, the number of images and subjects contained in the used dataset, the reported biometric performance (EER if not stated otherwise) and the year of publication, sorted according to fusion level and year of publication. All the related works listed in Table 10.1 are described in the following.

Yang and Jia [21] presented a multispectral finger vein fusion approach by fusing enhanced finger vein images captured in different wavelengths. They applied an image denoising method followed by image registration and a brightness adjustment prior to the image-level fusion of images captured in six different wavelength bands. Their image-level fusion strategy operates pixel-wise and is based on an improved regional energy integration method in the spatial domain. The comparison scores are obtained by phase-only correlation. They achieved a minimum EER of 11.02% by fusing all six bands.

Guan et al. [22] applied feature-level fusion to Wavelet transform based vein image features. The high- and low-frequency Wavelet features are obtained independently and then fused by a simple nearest-neighbour rule. They did several experiments using different training set sizes and arrived at a maximum recognition rate of 94.35%. Yang and Zhang [23] proposed a feature-level scheme using global and local features. The local features are extracted using a Gabor filter framework and the global ones using 2D invariant moments. The fusion itself is performed by a weighted fusion strategy based on canonical correlation analysis. They reported a lowest FAR of 1.15% and a FRR of 2.47% for their fused features. Gupta and Gupta [24] proposed a feature-level fusion approach of two distinct binary vein features (the features are binary vein images). The first type of features is extracted using repeated

Table 10.1 Related work in single modality finger vein fusion, ordered according to fusion level and year of publication

Reference	Fusion level	Origin	Images/subjects	Performance (EER)	Year
[21]	Image	Multi-sample	5760/60	11.02%	2012
[22]	Feature	Single-sample	2044/292 (fingers)	Recognition rate: 94.35%	2009
[23]		Single-sample	640/64	FAR: 1.15%, FRR: 2.47%	2010
[24]		Single-sample	3132/156	2.98%	2015
[26]		Single-sample	1440/60	0.19%	2016
[27]	Score	Single-sample	1200/100	0.28%	2010
[28]		Multi-instance	1440/80	0.83% (fusion of 3 fingers)	2012
[29]		Single-sample	4000/50	0.011%	2012
[30]		Single-sample	4080/30	1.56%	2013
[31]		Single-sample	4260/71 (680/85)	2.63%/0.78%	2013
[32]		Single-sample	3804/634 (fingers)	2.84%	2013
[33]		Single-sample	1440/60	0.27%	2014
[2]		Multi-sample	454860/63	0.04%	2018
[35]	Decision	Single-sample	1620/54	FAR: 0.0086% at 1% FRR	2009

line tracking [25]. The second type of features is obtained by multi-scale matched filtering. A variational approach is proposed to fuse both feature extraction methods. The score calculation is conducted by first aligning the two input images with the help of an affine transformation. The affine transformation matrix is found using a gradient descent optimisation based on a sum of squared differences cost function. The authors report a minimum EER of 2.98%. Kauba et al. [26] used different binary vein feature extraction schemes and applied several advanced feature-level fusion schemes (COLLATE, STAPLE, STAPLER), which were originally proposed for segmentation of magnetic resonance imaging (MRI) brain images together with simple average and majority voting based fusion in the finger vein domain. They conducted two different sets of experiments exhibiting two different fusion strategies. In the first one, only a single feature extraction scheme was used with a set of several different feature extraction parameters per input image. The output features

obtained for the individual parameters where then fused together. In the second set, different feature extraction schemes were applied per input image and their outputs were fused. The authors showed that both strategies (single feature extractor as well as multiple feature extractors) lead to an improvement in the recognition accuracy. The best EER achieved for the first strategy was 0.29% and for the second one 0.19% compared to the best EER for the single features of 0.47%.

Zhou and Kumar [27] proposed a score-level fusion scheme for palm vein recognition based on multiple representations. They extracted four different kinds of features, two based on their proposed representations. The first ones are using Hessian phase information from the vein images, the second ones using localised Radon transform to generate a kind of orientation encoding. The other two ones are based on Ordinal Code and a Laplacian representation, respectively. These four feature representations are compared individually to get the output scores which are then fused by applying a heuristic fusion rule. The authors arrived at a minimum EER of 0.28%. Yang et al. [28] did a score-level fusion of extracted features from multiple fingers of the same subject. They used LBP based features and a Hamming distance based comparison module to generate the scores. These scores are then fused using a simple sum rule in combination with triangular norm. Their best reported EER of 0.83% was achieved by fusion ring, middle and index finger using Frank's t-norm. In [29] Kang Park used local as well as global vein features in combination with score-level fusion. The local features are extracted by the help of LBP and compared using the Hamming distance. The global ones are Wavelet transform based features which are compared using the Euclidean distance. The comparison scores are then fused with the help of a radial basis function based support vector machine. Park reported a best achieved EER of 0.0011%. Liu et al. [30] proposed a score-level fusion scheme including pixel as well as super-pixel based finger vein features. LBP, vein pattern structure based and vein minutiae based features form the pixel based features. The super-pixel based image segmentation is done using the SLIC method. Histogram, gradient and entropy features extracted from the super-pixel based segmentation are then combined and form the super-pixel based features. An Euclidean distance based comparison of both individual features is performed to calculate the comparison scores. These scores are normalised and fused by using the weighted average fusion strategy. The weights are tuned to achieve an optimal EER. They reported a minimum EER of 1.56%. Qin et al. [31] applied score-level fusion to multiple representations of the same finger vein pattern. The vein pattern is represented by three different types of features: finger vein shape based, finger vein orientation based and SIFT feature point based features. The former two are subregion partitioned and subregion compared with the help of the SIFT based features, which are treated individually, leading to three comparison scores. The scores are normalised using the Z-score normalisation and then fused by applying a weighted-sum rule based fusion as well as a support vector machine based fusion. They achieved minimum EERs of 2.63 and 0.78%. Lu et al. [32] proposed a score-level fusion scheme based on Gabor features. Usually, the individual filter responses obtained from the Gabor filter bank are weighted and/or directly combined into a single output feature. Instead, the authors extract and compare the output of each single Gabor filter channel separately.

The corresponding comparison scores are then fused using a simple weighted-sum rule. The authors were able to get an EER of 2.84% using their proposed method. Kauba et al. [33] tested different preprocessing cascades in order to improve the individual performance of the single finger vein feature extraction schemes. Binary and SIFT/SURF based features were compared individually to obtain the output scores. These scores were normalised using Min-Max normalisation and then fused using weighted sum/product/average/minimum/maximum fusion rule. The best fusion rule in terms of lowest EER was chosen accordingly. They were able to achieve a minimum EER of 0.27% with the help of score-level fusion compared to a minimum EER of 0.47% for the single features. In our previous work [2], we performed a multi-sample score-level fusion of several different perspectives around the finger. Therefore, we established a multi-perspective finger vein dataset with the help of our self-designed multi-perspective finger vein capture device, described in Sects. 10.4 and 10.3, respectively. Several different perspectives starting from 2 up to 72 were fused at score-level for 4 different kinds of extracted features using a simple sum-rule based fusion. We achieved a best overall EER of 0.039% for the fusion of 18 different views and Maximum Curvature [34] features.

Yang et al. [35] proposed a decision-level fusion approach based on three different finger vein feature representations. They extracted a topological feature, a local moment based feature and a vein shape based feature. These features were compared individually by means of a nearest cosine classifier outputting the class which the input feature belongs to. These output decisions were then fused by the help of the Dempster–Shafer algorithm. The authors reported a lowest FAR of 0.0086% at a FRR of 1%.

10.5.1.2 Multi-modality Fusion Including Finger Veins

In addition to the single modality fusion approaches, several multi-modality fusion approaches including finger veins as one of the involved biometric traits were proposed. Table 10.2 gives an overview of these approaches, including the reference to the original publication, the fusion level, the involved biometric traits, the number of subjects in the dataset used, the reported performance (EER if not stated otherwise) and the year of publication. Most approaches fuse finger-related biometrics, including fingerprint, finger texture, finger shape, finger knuckle and finger veins. There are only two approaches involving other biometrics than finger-related ones. Razzak et al. [36] fused face and finger veins and He et al. [37] fused face, fingerprints and finger veins. Both applied score-level fusion. The number of involved traits varies between at least two and at most four. Fingerprint is the most prominent one [37–46] besides finger veins that is included in the fusion followed by finger texture [38, 43, 45, 47–49] as the second most prominent one and finger shape [42, 43, 50–52] as the third one. The majority of the approaches is based on feature-level and score-level fusion, there are only two decision-level fusion approaches compared to eight

Table 10.2 Related work in finger vein fusion, multi-modality fusion involving finger veins, ordered according to fusion level and year of publication

References	Fusion level	Involved traits	Subjects	Performance (EER)	Year
[40]	Feature	Fingerprint, finger veins	40	1.85% FRR and 0.97% FAR	2011
[44]		Fingerprint, finger veins	64	1.35% FAR at 0% FRR	2012
[46]		Fingerprint, finger veins	40	1.485%	2012
[48]		Finger texture, finger veins	220	0.45%	2012
[49]		Finger texture, finger veins	220	0.435%	2014
[43]		Finger texture, finger shape, fingerprint, finger veins	100	0.00796%	2015
[45]		Finger texture, fingerprint, finger veins	300	0.415%	2016
[51]	Score	Finger shape, finger veins	816	0.075%	2010
[37]		Face, fingerprint, finger veins	510	99.8% GAR at 0.01% FAR	2010
[36]		Face, finger veins	35	5% FAR and 92.4% GAR	2010
[47]		Finger texture, finger veins	312	0.08%	2012
[52]		Finger shape, finger veins	120	4%	2013
[50]		Finger shape, finger veins	492	1.78%	2014
[42]		Finger shape, fingerprint, finger knuckle, finger veins	100	0.0319%	2014
[38]		Finger texture, fingerprint, finger veins	378	0.109%	2015
[41]	Decision	Fingerprint, finger veins	33	1.86%	2011
[39]	Feature/decision	Fingerprint, finger knuckle, finger veins	165	0.04%	2016

feature-level and eight score-level ones. All proposed fusion approaches showed a significant improvement in the recognition accuracy of the fusion compared to using finger veins only.

10.6 Experimental Analysis

This section describes the experimental part of this chapter. At first, the used subset of the dataset introduced in Sect. 10.4 is explained. Afterwards, the finger vein recognition tool chain which is employed during the experimental analysis is described. This is followed by a presentation of the fusion strategy and the applied score-level fusion framework. Afterwards, the experimental protocol to determine the FAR and FRR and consequently the recognition performance in terms of EER/FMR1000/ZeroFMR is explained. Then the results of the individual fusion strategies are given and discussed. Finally, this section is concluded with an overall results discussion.

10.6.1 Finger Vein Dataset

To reduce the amount of data during the fusion, we used a subset of the multi-perspective finger vein dataset [2] only. Not all 360 different perspectives are evaluated, but only each fifth one is considered. Thus, there is a total of 73 different perspectives ($\frac{360°}{5°/step} = 72$ plus the last one which is $360° = 0°$ again results in 73). All 63 capture subjects, 4 fingers per subject and 5 images per view and finger are considered. This results in a total of $73 \times 63 \times 4 \times 5 = 91{,}980$ images instead of 454,860 for the total dataset.

10.6.2 Finger Vein Recognition Tool chain

The finger vein recognition tool chain includes all steps of a biometric recognition system starting with the extraction of the Region of Interest (ROI) to preprocessing, feature extraction and comparison. The input data are the images of the different individual perspectives acquired from the 3D capture device, the output is a comparison score that can be used to determine whether the provided finger belongs to a certain (enrolled) data subject or not.

ROI Extraction

Prior to the ROI extraction, the finger is aligned and normalised. The alignment should place the finger always in the same position in the image, independent of the relative position of the finger during the acquisition. To achieve this, the finger lines (edge between finger and the background of the image) are detected and the centre

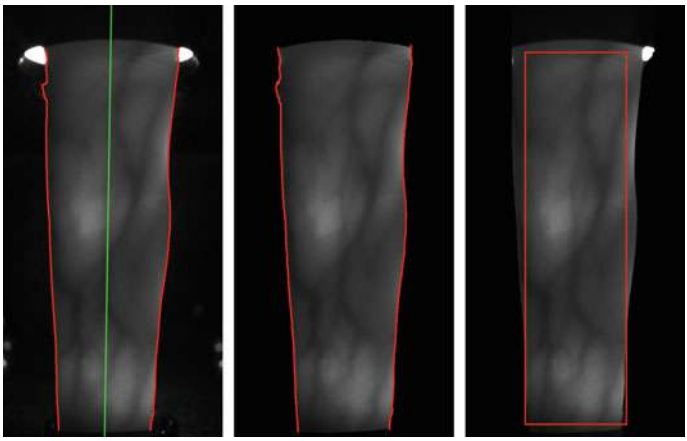


Fig. 10.7 ROI extraction process (images originally published in [2], ©2018 IEEE)

line (in the middle of the two finger lines) is determined. Afterwards, the centre line of the finger is rotated and translated in a way that it is placed in the middle of the image and the image region outside of the finger is masked by setting the pixels to black. The final step is to extract a rectangular ROI of a fixed size (1100 × 300 pixel) from a fixed position. The three steps are visualised in Fig. 10.7. The implementation used is based on the method proposed in [53].

Preprocessing

Preprocessing tries to enhance the low contrast and improve the image quality. In the following the preprocessing methods, we employed in our finger vein recognition tool chain are explained.

Simple **CLAHE** [54] or other local histogram equalisation techniques are most prevalent according to the literature for this purpose. A localised contrast enhancement technique like CLAHE is a suitable baseline tool to enhance the vein images as they exhibit unevenly distributed contrast. CLAHE has an integrated contrast limitation (clip limit) which should avoid the amplification of noise.

High-Frequency Emphasis Filtering (**HFEF**) [55], originally proposed for hand vein image enhancement tries to enhance the vein images in the frequency domain. At first, the discrete Fourier transform of the image is computed, followed by the application of a Butterworth high-pass filter of order n. The authors originally proposed to use a global histogram equalisation but we decided to apply CLAHE instead.

Circular Gabor Filter (**CGF**) as proposed by Zhang and Yang [56] is another finger vein image enhancement technique which is rotation invariant and achieves an optimal joint localisation in both, the spatial and the frequency domain. The authors originally suggested to use grey level grouping for contrast enhancement but we again apply CLAHE instead.

Furthermore, the images were resized to half of their original size, which not only speeded up the comparison process but also improved the results. For more details on

the preprocessing methods, the interested reader is referred to the authors' original publications.

Feature Extraction

We used five different feature extraction methods. The first three techniques discussed aim to extract the vein pattern from the background resulting in a binary image (vein pattern based methods) followed by a comparison of these binary images using a correlation measure. All algorithms are well-established finger vein recognition algorithms. We used the publicly available implementations published in [5].

Maximum Curvature (MC [34]) aims to emphasise only the centre lines of the veins and is therefore insensitive to varying vein widths. The first step is the extraction of the centre positions of the veins by determining the local maximum curvature in cross-sectional profiles obtained in four directions: horizontal, vertical and the two oblique directions. The cross-sectional profile is determined based on the first and second derivates. Then each profile is classified as either being concave or convex, where only the local maxima belonging to a concave profile indicate a vein line. Afterwards, a score according to the width and curvature of the vein region is assigned to each centre position and recorded in a matrix called locus space. Due to noise or other distortions, some pixels may not have been classified correctly at the first step, thus the centre positions of the veins are connected using a filtering operation in all four directions taking the 8-neighbourhood of pixels into account. The final binary output image is obtained by thresholding of the locus space using the median as a threshold.

Principal Curvature (PC [57]): At first the gradient field of the image is calculated. In order to prevent the unwanted amplification of small noise components, a hard thresholding which filters out small gradients by setting their values to zero is done. Then the gradient at each pixel is normalised to a magnitude of 1 to get a normalised gradient field. This normalised gradient field is smoothed by applying a Gaussian filter. The next step is the actual principal curvature calculation. The curvatures are obtained from the Eigenvalues of the Hessian matrix at each pixel. The two Eigenvectors of the Hessian matrix represent the directions of the maximum and minimum curvature and the corresponding Eigenvalues are the principal curvatures. Only the bigger Eigenvalue which corresponds to the maximum curvature among all directions is used. The last step is a threshold based binarisation of the principal curvature values to arrive at the binary vein output image.

Gabor Filter (GF [47]): Gabor filters are inspired by the human visual system's multichannel processing of visual information and have been widely used in biometrics. A Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave. Kumar and Zhou [47] proposed a Gabor filter based finger vein extraction approach. Therefore, a filter bank consisting of several 2D even symmetric Gabor filters with different orientations (in $\frac{\pi}{k}$ steps where k is the number of orientations) is created. k feature images are extracted by filtering the vein image using the different filter kernels contained in the Gabor filter bank. The final feature image is obtained by summing all the single feature images from the previous step and thresholding

the resulting feature image. This image is then post-processed using morphological operations to remove noise to get the final binary vein output image.

In contrast to the vein pattern based techniques described above, two key-point based techniques were used. Key-point based techniques try to use information from the most discriminative points as well as considering the neighbourhood and context information around these points by extracting key-point locations and assigning a descriptor to each detected key-point location.

The first one is a **Scale-Invariant Feature Transform** (SIFT [58]) based technique with additional key-point filtering along the finger boundaries to suppress information originating from the finger shape instead of the vascular pattern. This technique was originally proposed by Kauba et al. [33].

Deformation-Tolerant Feature Point Matching (DTFPM [13]): The second key-point based technique replaces the conventional SIFT descriptor and key-point detector by vascular pattern tailored ones. This method is robust against irregular shading and vein deformations due to posture changes. At first, the authors apply a technique originally proposed by Yang and Yang [59] for enhancing the vein images. Then a minimum-curvature map is calculated from the enhanced vein images based on Eigenvalue analysis. The feature point locations are determined from this curvature image (smaller Eigenvalue) at any point where the vein shape is non-linear. The feature descriptor takes the vein shape around the key-point location into account and is extracted from the so-called vein pattern map (larger Eigenvalue). The feature vector contains a quantification of the different vein directions inside a variable-sized window around the key-point location. The descriptor is normalised with the help of a finger shape model in a way that the descriptor area becomes smaller the closer the key-point location is to the finger boundaries. The authors claim that their proposed method is tolerant against several different types of finger posture changes, e.g. longitudinal finger rotation, translations and bending of the finger.

Comparison

For the comparison of the binary feature images we extended the approach in [25] and [34]. As the input images are neither registered to each other nor aligned, the correlation between the input image and in x- and y-direction shifted versions of the reference image is calculated. The maximum of these correlation values is normalised and then used as the final comparison score.

The SIFT features are compared by finding their nearest neighbours/best correspondences and calculating a score based on the distances between the corresponding key-points.

DTFPM employs a deformation tolerant comparison strategy by using non-rigid registration. At first, the correspondences between the key-points in the two images for comparison are found. These correspondences are filtered using a local and global histogram technique based on the relative distances between the corresponding key-points. After this filtering step, the key-point coordinates of one of the involved feature vectors are transformed by applying a non-rigid transformation based on an outlier-robust thin-plate spline model as proposed in [60]. Afterwards, the correspondences between the adjusted key-points are determined again. These updated

correspondences are filtered by a comparison of the descriptor distances with fixed thresholds. The final comparison score is determined as the ratio of the matched points and the sum of the number of detected key-points in both images.

10.6.3 Score-Level Fusion Strategy and Toolkit

We applied three different fusion strategies. The first strategy involves the fusion of all possible combinations of pairs of distinct views (which are $\binom{N}{k} = \binom{73}{2} = 2628$ combinations, 73 different views are considered) as well as all possible three tuples of distinct views (which are $\binom{73}{3} = 62196$ combinations) for each of the five-feature extraction methods. As motivated in the introduction, it is beneficial if the number of involved views is as little as possible to reduce the complexity and the production costs of the biometric capture device and to be able to build such a device without any moving parts. Thus, only pairs and three tuples are considered here. The second strategy employs the fusion of all possible combinations of feature extraction methods per view. There are $\binom{5}{2} + \binom{5}{3} + \binom{5}{4} + \binom{5}{5} = 26$ combinations per perspective, resulting in a total of 10,830 different fusion combinations. Here, our aim is to identify the best combination of features for each individual view which does not necessarily have to be the same across all the different views. The third strategy is a combination (fusion) of the best results obtained during the first and second one.

All three fusion strategies are applied at score-level. The second strategy could be applied at feature-level too, but not for all the involved feature extraction types as they are not compatible with each other. The feature-level fusion of MC, PC and GF is possible while the fusion of DTFPM and SIFT with any of the other feature extraction types is not possible. Feature-level fusion is not possible for the first strategy at all, as there is no meaningful way to combine the features of different perspectives, e.g. by merging the extracted vein lines or using majority voting as the visible vein lines differ for each view. Score-level fusion usually performs better than decision-level fusion, as there is more information available at the score level and there are more variants to fuse the individual scores. Hence, we decided to apply score-level fusion in all three fusion strategies.

In our previous work [2], a simple sum based fusion rule, without any weights for the input scores, was applied. In this work, a more advanced score-level fusion approach, namely the BOSARIS toolkit [14] is utilised. BOSARIS provides a MATLAB based framework for calibrating, fusing and evaluating scores from binary classifiers and has originally been developed for automatic speaker recognition. It can be applied to any biometric trait where two alternate classes are distinguished (genuine/impostor). The toolkit provides several functionalities, e.g. a normalised Bayes error rate plot, ROC and DET plots, including efficient algorithms to generate these plots for large score files, logistic regression solutions for the fusion of several subsystems, solutions for calibration (mapping scores to likelihood ratios), a logistic regression optimiser and an efficient binary score file format. During this work, we only harness the fusion capabilities of BOSARIS though. BOSARIS needs

a supervised training phase where combination weights are trained based on logistic regression in order to fuse multiple input systems into a single output one providing well-calibrated log-likelihood-ratios. This is achieved by employing a general purpose, unconstrained convex optimisation algorithm, which is used to train the logistic regression fusion and calibration methods. Hence, BOSARIS needs a training set of data to find the optimal combination of weights for the actual fusion in order to minimise the classification error and thus to maximise the recognition performance based on the fused output scores. BOSARIS has the option to set a target prior according to the costs of a miss and a false alarm for the training phase of the fusion. We set this target prior to 0.5 assuming that the costs of a miss and a false alarm are both weighted equally.

10.6.4 Evaluation Protocol

The experiments are split into four parts: in the first part, we analyse the recognition performance of all single perspectives. Every perspective is considered as a separate dataset. Here, we do not perform any cross-projection comparison. The images are processed as described in Sect. 10.6.2 and 73 projections all around the finger in 5° steps are extracted. The recognition performance is quantified in terms of the EER as well as the FMR1000 (the lowest FNMR for FMR = 0.1%) and the ZeroFMR (the lowest FNMR for FMR = 0%). The performance values are calculated for each single perspective. For the parameter optimisation, the data set is divided into two roughly equal-sized subsets. The division is based on the contained subjects, i.e. all fingers of the same person are in one subset. Each subset is used to determine the parameters which are then applied to the other subset. This ensures a 100% separation of the data used for determining the optimal parameters and the actual test set. The necessary comparison scores for the FAR/FRR calculation, which is the basis for the EER/FMR1000/ZeroFMR calculation, are determined according to the test protocol of the FVC2004 [61]: to compute the genuine scores, all possible genuine comparisons are done. Instead of computing all possible impostor scores only the first image of a finger is compared against the first image of all other fingers. The final results are evaluated based on the combined scores (genuine and impostor) of both test runs. The parameter optimisation is executed only for the palmar dataset. The same parameter settings are also applied for the experiments on the other perspectives. The resulting number of comparisons for both subsets are listed in Table 10.3. All performance-related result values are given in percentage terms, e.g. 0.04 means 0.04%.

In the second part of our experiments, we fuse different features originating from the same feature extraction method but extracted from different perspectives as described in Sect. 10.6.3. The third part of the experiments is dedicated to a multi-algorithm fusion. We fuse all possible combinations of the five employed feature extraction methods at score level based on the scores obtained during the first part of the experiments, resulting in 2-, 3-, 4- and 5-tuples. In the last part, we com-

Table 10.3 Number of comparisons for each subset

Name	Subjects	Genuine	Impostor	Total
Subset 1	32	1280	8128	9408
Subset 2	31	1240	7626	8866
Total	63	2520	15,754	18,274

bine the two strategies of multi-perspective and multi-algorithm fusion. Based on the results from the two individual fusion strategies we determine the best possible combinations/fusion of perspectives and feature extraction methods. All four parts are evaluated using the same protocol to determine the performance figures. For all fusion experiments, the input data are the comparison scores generated during the single perspective experiments. We apply a fivefold cross-validations procedure, where we use every fold once for the training of the fusion module. The determined fusion parameters are applied to the test data consisting of the four remaining folds. The final results are evaluated based on the combined scores (genuine and impostor) of all five test runs.

We provide the scores files for each individual perspective and feature extraction methods as well as a script to run BOSARIS and generate all the fused scores files and performance figures we used during our experiments. These files and the scripts can be downloaded at http://www.wavelab.at/sources/Prommegger19b/.

10.6.5 Single Perspective Performance Results

The single perspective analysis for MC, PC, GF and SIFT have already been carried out in our previous work [2]. We added DTFPM as an additional key-point based recognition scheme. We had to change our ROI extraction to make the ROIs compatible with DTFPM. Our previous ROI approach selected a fixed size rectangle placed at the centre of the finger, independent of the finger's width. DTFPM is sensitive to parts of the finger outline and background areas that are contained in the input images and expects the finger width normalised to the ROI height. Thus, we updated our ROI extraction scheme as described in Sect. 10.6.2 and recalculated the results for the already evaluated algorithms based on the new ROIs. Note that due to the new ROIs these updated results are different from our previous work. Figure 10.8 top shows the results in terms of EER. There are two lines for every method: the thin line shows the actual EER value, the thicker line is a smoothed version calculated based on the EER using a moving average filter of size 5, which should highlight the trend of the recognition performance. The images captured of neighbouring views contain quite a similar vein structures (note that our step-width is 5°), thus the recognition performance is similar too. The best results are obtained around the palmar (0°, 360°) and dorsal (180°) region. The results of the perspectives in-between are inferior. This

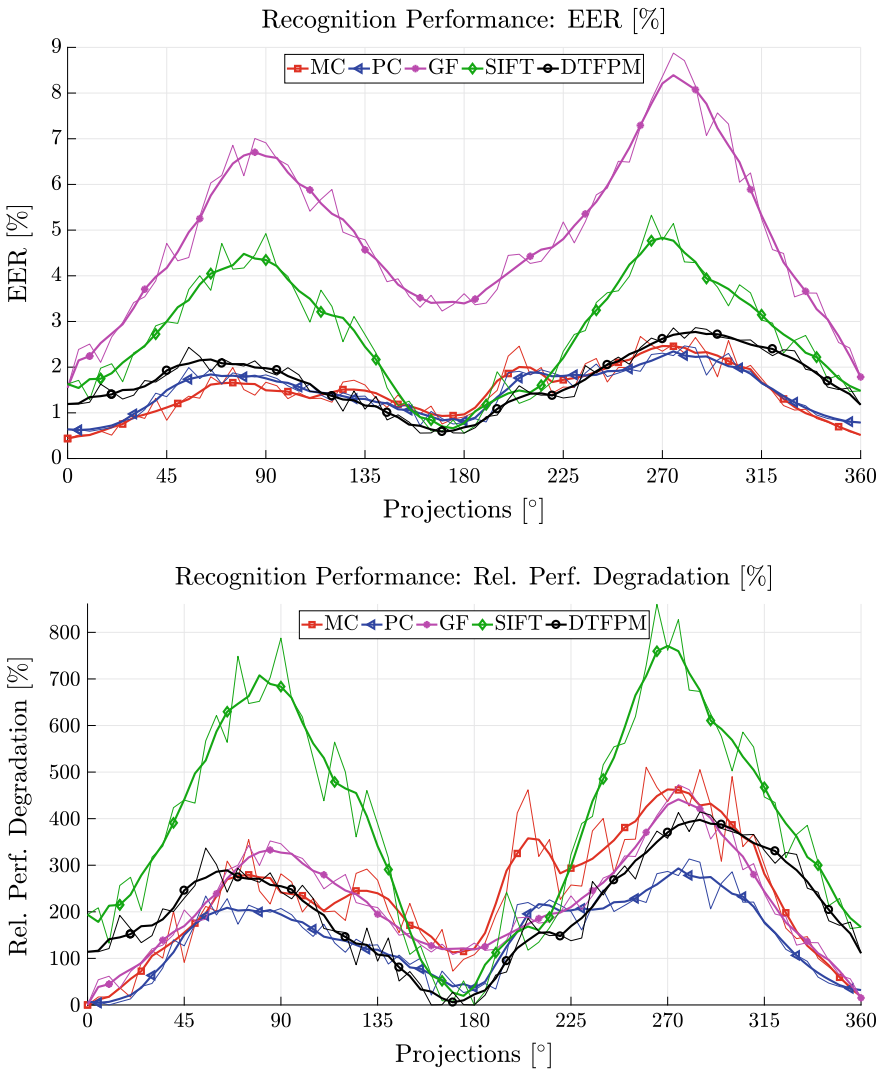


Fig. 10.8 Recognition performance for different projections: EER (top) and relative performance degradation in relation to the best performing view (bottom)

is due to the fact, that they contain fewer visible vein lines and thus fewer vein information than the palmar and dorsal view. Figure 10.9 shows the original ROI, the ROI after preprocessing and the extracted features (using MC) for the views 0°, 90°, 180° and 270°. It reveals that the 90° and 270° views contain less vein information than the palmar and dorsal view. Moreover, the vein extraction algorithms include some features related with the texture of the finger. This is especially visible at 180° where

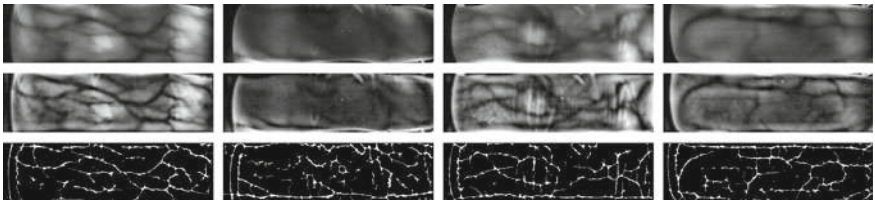


Fig. 10.9 ROI (first row), enhanced images (second row) and extracted MC features (third row) for different projections (originally published in [2], ©2018 IEEE). Note that there are less vein lines visible for 90° and 270° compared to 0° and 180°

some of the features are related with the finger knuckles instead of veins. These features are visible as horizontal lines in the feature image.

For the key-point based algorithms, especially SIFT, the palmar region exhibits a better performance than the other perspectives as well, but the best performance is achieved around the dorsal region. For SIFT this can be explained based on the employed preprocessing: only image (vein) enhancement and no vein extraction (binarisation) ahead of the SIFT key-point calculation is applied. Hence, the non-vein finger texture information is not suppressed in the input images of SIFT. Especially, the structure of finger knuckles seem to contain a lot of additional information which SIFT is able to exploit during feature extraction. Finger knuckles have been introduced by Zhang et al. [62] as an independent biometric characteristic. Yang et al. [63] experienced a similar behaviour. They fused the finger texture of the dorsal view with the vein structure of the palmar view which leads to an improvement in the recognition performance. Consequently, the additional information originating from the finger knuckles and the finger texture present at the dorsal view leads to the superior performance of SIFT for the dorsal view compared to the palmar one.

Table 10.4 lists the information regarding the best and worst perspective for each feature extraction method. MC, PC and GF perform best around the palmar view (note that 360° = 0°), while SIFT and DTFPM perform best around the dorsal view. The overall best result was achieved for MC at 0° with an EER of 0.44% (±0.15) where the number in brackets is the confidence interval. For all feature extraction methods, the worst results can be reported around 270°. The Relative Performance Degradation (RPD) of the different perspectives is visualised in Fig. 10.8 bottom. The RPD, stated in Eq. (10.1), is calculated with respect to the minimum EER (EER^{FT}_{min}) reached for a certain feature extraction method, where $EER^{FT}_{perspective}$ is the EER of the current perspective. The maximum performance degradation across the different algorithms is between 200 and 800%.

$$RPD^{FT}_{perspective} = \frac{EER^{FT}_{perspective} - EER^{FT}_{\text{min}}}{EER^{FT}_{\text{min}}} \tag{10.1}$$

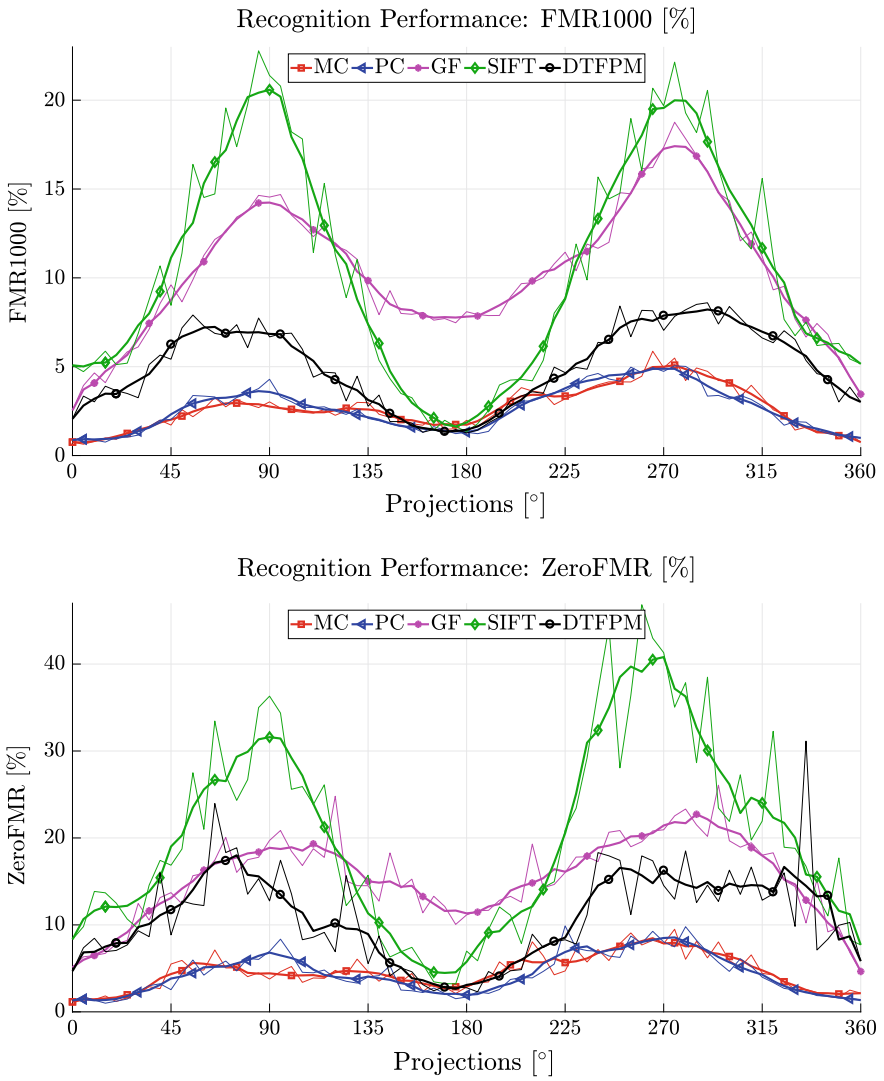


Fig. 10.10 Recognition performance among the different projections: FMR1000 (top), ZeroFMR (bottom)

The FMR1000 and ZeroFMR are visualised in Fig. 10.10 top and bottom, respectively. They follow the same trend as the EER: a good performance around the palmar and dorsal region and an inferior one for the views in between.

Table 10.4 Best/worst single perspective results per feature extraction method and single perspective

Feature type	Best perspective				Worst perspective			
	View	EER	FMR1000	ZeroFMR	View	EER	FMR1000	ZeroFMR
MC	0°	**0.44 (±0.15)**	0.76	1.15	260°	2.67 (±0.37)	4.46	7.69
PC	10°	0.60 (±0.18)	0.87	1.35	280°	2.47 (±0.36)	5.02	9.79
GF	0°	1.55 (±0.28)	2.54	5.13	275°	8.87 (±0.65)	18.76	22.54
SIFT	180°	0.55 (±0.17)	1.35	6.98	265°	5.33 (±0.53)	20.67	42.98
DTFPM	160°	0.56 (±0.17)	1.31	3.13	285°	2.87 (±0.38)	8.51	12.56

10.6.6 Multi-perspective Fusion Results

In the second part of our experiments, we analyse the impact of fusing the extracted features of the same feature extraction method from multiple perspectives (MPF). In detail, we evaluate the fusion of all possible pairs and three tuples.

The first part of this section deals with the fusion of all possible pairs. Figure 10.11 shows heat maps of the EER for all combinations per feature extraction method (top row: MC, PC, bottom row: GF, SIFT and DTFPM). The perspectives involved in the fusion are plotted on x- and y-axis, whereas the performance in terms of EER is visualised using a colour scheme from light/white which corresponds to a low EER (good performance) to dark/red which corresponds to a high EER (bad performance). The actual logarithmic scale is given in the colour bar on the right side of the plots. Note that the results are symmetric with regard to the main diagonal (45°). This diagonal corresponds to the single perspective performance results and is visible as dark line (high EER) in all five plots.

According to the performance analysis of the single perspectives (Sect. 10.6.5), the palmar and dorsal region perform best. Although, there are slight variations among the different feature extraction methods, the results obtained from the single perspectives are confirmed by the two-perspective fusion: a combination of two perspectives including the palmar (close to 0°, 360°) or dorsal (close to 180°) region always results in a good recognition performance. A fusion of two views in-between those two regions result in an inferior performance. For MC, PC and GF the EER for all fusion combinations including the palmar (area along the outer edges of the plot) and dorsal view (cross lines in the centre) perform better (light, white to yellow colours) than fusion combinations without these views (dark, orange to red colours), achieving the best results when both regions are fused (light, white colour).

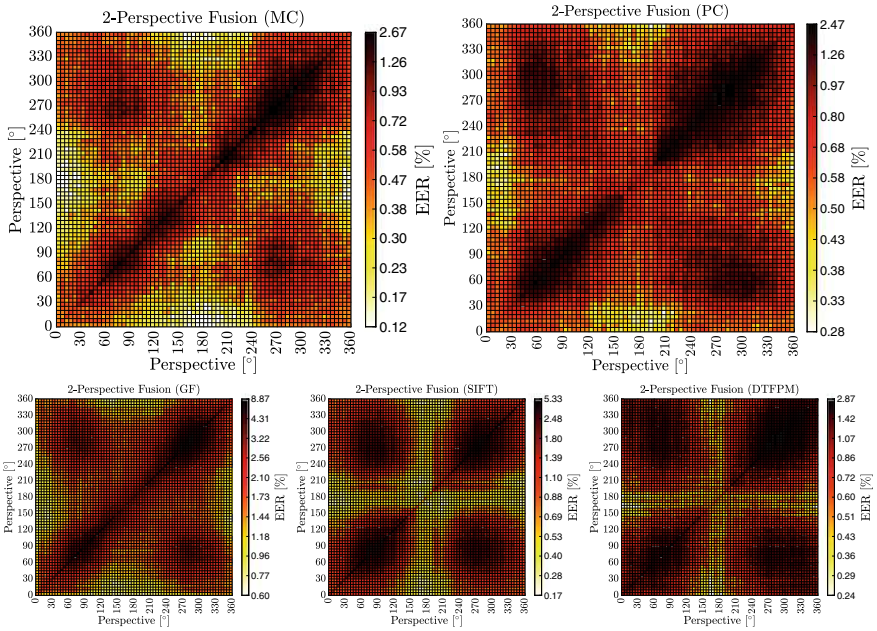


Fig. 10.11 Recognition performance for two-view fusion. Top row: MC (left), PC (right), bottom row: GF (left), SIFT (middle) and DTFPM (right)

Both key-point based methods show a different behaviour. The fusion of the palmar and dorsal region is still superior to all other fusion combinations, but SIFT and DTFPM perform well if the dorsal perspective is included in the fusion in general. This can also be seen in the plots as the 180° cross shows light, white to yellow colours which indicates a good performance. For SIFT, this is even more pronounced than for DTFPM.

Table 10.5 lists the best results in terms of EER, FMR1000 and ZeroFMR for each feature extraction method in detail. MC when fusing 0° and 180° achieves the overall best performance with an EER of 0.12%. For the evaluation of the results, the single perspective baseline EER and the relative performance increase (RPI) with respect to the baseline EER, as calculated in Eq. (10.2), are stated. The performance increase compared to the best single view result is between 110% (PC) and 270% (MC), which corresponds to a 2–3.5 times lower EER than the single perspective performance, respectively.

$$RPI = \frac{EER_{Baseline} - EER_{Fusion}}{EER_{Fusion}} \tag{10.2}$$

In addition to all pairs, all possible triples are evaluated. Table 10.6 shows the five best performing combinations per feature extraction method. Again the single perspective baseline EER and the relative performance increase is included. The highest

Table 10.5 Best two-perspective fusion results per feature extraction method. Best result is highlighted **bold font**. For comparability also the single perspective baseline EER and the relative performance improvement (based on the single perspective performance) is included

Feature type	2 Perspective fusion					Single perspective		Rel. Perf.
	View 1	View 2	EER	FMR1000	ZeroFMR	View	EER	Incr. [%]
MC	0°	180°	**0.12 (±0.08)**	0.12	0.16	0°	0.44	**264.90**
PC	10°	190°	0.28 (±0.12)	0.36	0.56	10°	0.60	113.14
GF	140°	360°	0.60 (±0.18)	0.80	1.56	0°	1.55	156.48
SIFT	165°	205°	0.17 (±0.09)	0.36	1.63	180°	0.55	229.72
DTFPM	0°	160°	0.24 (±0.11)	0.32	1.55	160°	0.56	132.27

recognition performance improvement is between 150% for PC and 1100% for MC which is in any case better than the best two-perspective fusion (see Table 10.5). The overall best result with an EER of 0.036% is achieved using MC when fusing the 5°, 170° and 235° view.

Table 10.6 also includes the perspectives of interest. It is striking, that once again a lot of combinations include perspectives close to the palmar (0°, 360°) and dorsal (180°) regions. Thus, we additionally analysed the occurrence of the palmar and dorsal view in the top 25 results for each feature extraction method. All angles within a certain range around 0° and 180° are mapped to the palmar and dorsal region, respectively. Three different mapping ranges are evaluated: ±15° (345° −15°, 165° −195°), ± 20° (340°−20°, 160° −200°) and ± 25° (335°−25°, 155° −205°). The results are presented in Table 10.7. It turns out that the best performing individual region (palmar for MC, PC, GF and dorsal for SIFT and DTFPM) is present in most of the top 25 fusion combinations. At a mapping range of ±25° it is even included in at least 96% of the top 25 results. For this mapping range also the opposite region is part of at least 80% of the combinations, except for GF (only 24%). For GF, this can be explained by the big performance difference of palmar (~1.5%) and dorsal region (~3.6%).

In order to be able to decide whether a three-perspective fusion is beneficial compared to a two-perspective approach, one way is to calculate the significance of the recognition performance improvement. We use the method proposed in [64] to calculate a boundary for the significance from the achieved EERs. Table 10.8 lists the χ^2 values in detail. The following translations of χ^2 values into p_v values can be used to interpret the values stated in the table: $\chi^2 = 6.6$ corresponds to $p_v = 0.01(\equiv 1\%)$, $\chi^2 = 7.9$ to $p_v = 0.005(\equiv 0.5\%)$ and $\chi^2 = 10.8$ to $p_v = 0.001(\equiv 0.1\%)$. Thus, all performance improvements exhibiting $\chi^2 > 6.6$ are regarded as significant. The resulting χ^2 values indicate that a fusion of two and three perspectives lead to

Table 10.6 Recognition performance for three-view fusion: five best results per feature extraction method. Best result per feature extraction method is highlighted **bold font**. For comparability also the single perspective baseline EER and the relative performance improvement (based on the single perspective performance) is included

Feature type	3 Perspective fusion						Single perspective		Rel. Perf.
	View 1	View 2	View 3	EER	FMR1000	ZeroFMR	View	EER	Impr. [%]
MC	5°	170°	235°	**0.036 (±0.04)**	0.040	0.240	0°	0.44	**1111.78**
	0°	210°	235°	0.036 (±0.04)	0.040	0.120			1107.27
	10°	165°	215°	0.039 (±0.05)	0.040	0.159			1019.25
	20°	160°	235°	0.039 (±0.05)	0.040	0.040			1014.94
	165°	235°	355°	0.039 (±0.05)	0.040	0.159			1014.94
PC	10°	175°	200°	**0.238 (±0.11)**	0.401	0.602	10°	0.60	**150.21**
	20°	205°	235°	0.239 (±0.11)	0.319	0.638			149.65
	175°	235°	360°	0.239 (±0.11)	0.399	0.518			149.65
	140°	190°	360°	0.239 (±0.11)	0.282	0.524			149.59
	155°	210°	360°	0.239 (±0.11)	0.399	0.839			149.45
GF	125°	225°	360°	**0.284 (±0.12)**	0.401	1.325	0°	1.55	**446.48**
	90°	205°	360°	0.313 (±0.13)	0.638	1.794			394.98
	75°	140°	360°	0.321 (±0.13)	0.442	1.165			383.32
	120°	220°	355°	0.321 (±0.13)	0.758	1.475			383.09
	120°	200°	360°	0.321 (±0.13)	0.481	0.882			382.82
SIFT	165°	205°	350°	**0.058 (±0.05)**	0.040	0.635	180°	0.55	**857.58**
	20°	170°	210°	0.075 (±0.06)	0.040	0.714			643.62
	170°	205°	350°	0.081 (±0.06)	0.079	0.476			585.30
	170°	205°	335°	0.081 (±0.06)	0.079	0.635			585.30
	140°	205°	350°	0.081 (±0.06)	0.079	0.714			585.30

(continued)

Table 10.6 (continued)

Feature type	3 Perspective fusion						Single perspective		Rel. Perf.
	View 1	View 2	View 3	EER	FMR1000	ZeroFMR	View	EER	Impr. [%]
DTFPM	5°	160°	280°	**0.159 (±0.09)**	0.559	1.837	160°	0.56	**249.88**
	0°	180°	295°	0.162 (±0.09)	0.439	1.276			243.31
	15°	160°	295°	0.162 (±0.09)	0.439	1.637			243.04
	0°	180°	185°	0.165 (±0.09)	0.437	1.033			237.24
	0°	180°	245°	0.169 (±0.09)	0.439	2.396			228.78

Table 10.7 Analysis of the occurrence of palmar and dorsal views per feature extraction method in the 25 best three-perspective fusions. Both means that palmar and dorsal are present at the same combination.

Feature type (%)	Max distance ±15°			Max distance ±20°			Max distance ±25°		
	Palmar	Dorsal	Both	Palmar	Dorsal	Both	Palmar	Dorsal	Both
MC	84.0	52.0	40.0	92.0	76.0	68.0	**100.0**	84.0	84.0
PC	92.0	68.0	68.0	**100.0**	68.0	68.0	**100.0**	80.0	80.0
GF	**100.0**	8.0	8.0	**100.0**	16.0	16.0	**100.0**	24.0	24.0
SIFT	80.0	88.0	68.0	84.0	88.0	72.0	92.0	96.0	88.0
DTFPM	92.0	60.0	56.0	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**

a significant improvement compared to the single view performance, whereas the improvement for a three perspective fusion compared to fusing two views is lower but still significant for MC, GF and SIFT.

10.6.7 Multi-algorithm Fusion Results

This time different feature extraction methods per perspective are fused (MAF) instead of perspectives per feature extraction method. We evaluate all possible pairs, triples, quadruples and the combination of all five- feature extraction methods, resulting in 26 different combinations per perspective. Figure 10.12 shows the best fusion result per number of fused feature extraction methods. The best result, for example, two-feature extraction methods included in the fusion at 0° means that the best performing pair of features in terms of EER of all pairs calculated at 0° is depicted. It

Table 10.8 Estimated χ^2 from the EER for multi-perspective fusion. Best results per number of involved views is highlighted **bold font**

Feature extraction method	Best EER for [n] involved views			Significance $n_1 \rightarrow n_2$ (χ^2 value)		
	n = 1	n = 2	n = 3	n = 1→n = 2	n = 1→n = 3	n = 2→n = 3
MC	**0.44 (±0.15)**	**0.12 (±0.08)**	**0.036 (±0.04)**	33.415	62.660	8.265
PC	0.60 (±0.18)	0.28 (±0.12)	0.238 (±0.11)	21.264	28.576	0.622
GF	1.55 (±0.28)	0.60 (±0.18)	0.284 (±0.12)	**76.708**	**159.698**	**20.642**
SIFT	0.55 (±0.17)	0.17 (±0.09)	0.058 (±0.05)	36.650	72.755	10.054
DTFPM	0.56 (±0.17)	0.24 (±0.11)	0.159 (±0.09)	23.391	140.869	3.005

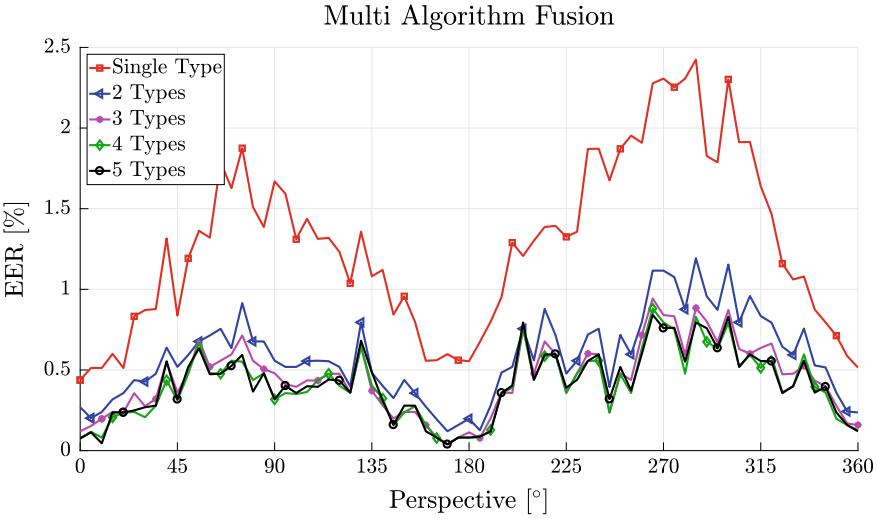


Fig. 10.12 Recognition performance for multi-algorithm fusion: best result in terms of EER per number of feature extraction methods fused is depicted for each perspective

can be seen that even the fusion of two-feature extraction methods increases the performance remarkably. Adding the third feature extraction method further improves the result, whereas fusing four- or five-feature extraction methods does not further improve the recognition performance significantly.

Table 10.9 lists the results of the MAF in more detail. The column occurrence states how often in terms of perspectives a feature extraction method combination performs superior to all other combinations of the same number of included feature

Table 10.9 Multi-algorithm fusion results per number of included features. Occurrence indicates the numbers of perspectives for which the specified combination achieves the best score, the given EER values are calculated over all perspectives. The two view columns state at which view the best and worst performance has been achieved. The best result per number of included feature extraction methods is highlighted **bold face**

# Features included	Feature types	Occurrences	Best		Avg	Worst	
			EER	View	EER	EER	View
1	MC	34 (46.58%)	**0.44 (±0.15)**	0°	1.46	2.67 (±0.37)	260°
	PC	19 (26.03%)	0.60 (±0.18)	10°	1.47	2.47 (±0.36)	280°
	DTFPM	16 (21.92%)	0.56 (±0.17)	160°	1.71	2.87 (±0.38)	285°
	SIFT	4 (5.48%)	0.55 (±0.17)	180°	2.75	5.33 (±0.53)	265°
	GF	–	1.55 (±0.28)	0°	4.89	8.87 (±0.65)	275°
2	PC, DTFPM	31 (42.47%)	0.20 (±0.10)	180°	0.66	1.32 (±0.26)	205°
	MC, DTFPM	22 (30.14%)	0.13 (±0.08)	185°	0.68	1.47 (±0.28)	285°
	MC, SIFT	11 (15.07%)	**0.12 (±0.08)**	170°	0.78	1.83 (±0.31)	265°
	SIFT, DTFPM	8 (10.96%)	0.16 (±0.09)	175°	1.04	2.08 (±0.33)	265°
	MC, PC	1 (1.37%)	0.32 (±0.13)	10°	0.95	1.95 (±0.32)	285°
	PC, SIFT	–	0.24 (±0.11)	180°	0.92	1.88 (±0.31)	265°
	GF, DTFPM	–	0.32 (±0.13)	180°	1.17	2.32 (±0.35)	265°
	GF, SIFT	–	0.40 (±0.14)	170°	1.63	3.56 (±0.43)	265°
	MC, GF	–	0.44 (±0.15)	0°	1.39	2.54 (±0.36)	300°
	PC, GF	–	0.51 (±0.16)	360°	1.28	2.32 (±0.35)	265°

(continued)

Table 10.9 (continued)

# Features included	Feature types	Occurrences	Best		Avg	Worst	
			EER	View	EER	EER	View
3	MC, SIFT, DTFPM	33 (45.21%)	**0.04 (±0.05)**	170°	0.50	0.99 (±0.23)	285°
	MC, PC, DTFPM	23 (31.51%)	0.12 (±0.08)	185°	0.52	1.23 (±0.25)	205°
	PC, SIFT, DTFPM	11 (15.07%)	0.12 (±0.08)	165°	0.53	0.96 (±0.22)	270°
	PC, GF, DTFPM	3 (4.11%)	0.23 (±0.11)	245°	0.62	1.31 (±0.26)	205°
	MC, GF, DTFPM	2 (2.74%)	0.16 (±0.09)	185°	0.66	1.47 (±0.28)	285°
	MC, PC, SIFT	1 (1.37%)	0.12 (±0.08)	170°	0.64	1.31 (±0.26)	265°
	MC, GF, SIFT	–	0.12 (±0.08)	170°	0.77	1.76 (±0.30)	265°
	GF, SIFT, DTFPM	–	0.12 (±0.08)	175°	0.82	1.68 (±0.30)	265°
	PC, GF, SIFT	–	0.25 (±0.11)	170°	0.82	1.71 (±0.30)	265°
	MC, PC, GF	–	0.32 (±0.13)	0°	0.94	1.91 (±0.31)	285°
4	MC, PC, SIFT, DTFPM	51 (69.86%)	**0.04 (±0.05)**	170°	0.42	0.88 (±0.21)	265°
	MC, PC, GF, DTFPM	10 (13.70%)	0.12 (±0.08)	185°	0.51	1.23 (±0.25)	205°
	MC, GF, SIFT, DTFPM	9 (12.33%)	0.04 (±0.05)	170°	0.50	1.07 (±0.24)	275°
	PC, GF, SIFT, DTFPM	3 (4.11%)	0.11 (±0.08)	185°	0.50	1.00 (±0.23)	265°
	MC, PC, GF, SIFT	–	0.09 (±0.07)	170°	0.63	1.32 (±0.26)	265°
5	MC, PC, GF, SIFT, DTFPM	73 (100.00%)	**0.04 (±0.05)**	170°	0.41	0.84 (±0.21)	265°

extraction methods. The minimum, average and maximum EER are determined based on the results for all perspectives of the given feature extraction method combination. Considering single feature extraction methods, MC or PC are included in more than 70% of the best results. GF is not included in any combination that performs best for any perspective. The results of fusing feature extraction method pairs clearly show that it is beneficial to fuse a vein pattern based algorithm (MC, PC, GF) to a key-point based one (SIFT, DTFPM). The combinations of either MC/PC and SIFT/DTFPM are leading to 98% of the best results in two-feature extraction methods fusion.

Table 10.10 Estimated χ^2 from the EER for multi-algorithm fusion

Nr of features EER	n = 1 0.44 (±0.15)	n = 2 0.12 (±0.08)	n = 3 0.04 (±0.05)	n = 4 0.04 (±0.05)	n = 5 0.04 (±0.05)
n = 1 0.44 (±0.15)	–	33.42	60.91	60.91	60.91
n = 2 0.12 (±0.08)	33.42	–	7.31	7.31	7.31
n = 3 0.04 (±0.05)	60.91	7.31	–	0	0
n = 4 0.04 (±0.05)	60.91	7.31	0	–	0
n = 5 0.04 (±0.05)	60.91	7.31	0	0	–

DTFPM (83%) is involved more often than SIFT (26%). Again, GF is not present in any of the best combinations. The overall best result with an EER of 0.04% is achieved when fusing MC, PC, SIFT and DTFPM. Once again, the analysis of the perspective, at which the best result is achieved, confirms, that views from the palmar (0°, 360°) and dorsal (180°) region perform best.

Same as for the two-perspective fusion, we also check the performance increase of three-perspective fusion on its significance. Table 10.10 lists the results in detail. The resulting χ^2 values indicate, that a fusion of two or more feature extraction methods is always beneficial compared to a single feature extraction method. The same holds true when comparing a two-feature extraction method fusion to a three, four or five one. However, applying a four or five feature-type fusion instead of a three feature-type one leads to no significant improvements anymore.

10.6.8 Combined Multi-perspective and Multi-algorithm Fusion

In this section, we combine multiple perspectives and multiple feature extraction methods into one combined fusion method (CMPMAF). For the selection of the relevant perspectives and feature extraction methods we considered the results for multi-perspective fusion (Sect. 10.6.6) and feature extraction method fusion (Sect. 10.6.7). Although the χ^2 values for the multi-perspective fusion in Table 10.8 are only boundaries, they still indicate that the performance increase from two to three perspectives is significant for MC, GF and SIFT. The drawback of adding additional perspectives is the added cost/complexity to the system (additional camera and illumination module, higher computational costs). Therefore, we decided that the significance of the improvement is not high enough to justify the extra effort. As

Table 10.11 Performance results: Fusion of vein pattern based with key-point based features for both, palmar and dorsal view. The best result is highlighted **bold face**

Feature types	Perspectives	EER	FMR1000	ZeroFMR
MC, SIFT	0°, 180°	**0.04 (±0.05)**	0.04	0.64
MC, DTFPM	0°, 180°	0.08 (±0.07)	0.08	0.12
PC, SIFT	0°, 180°	0.16 (±0.09)	0.16	0.32
PC, DTFPM	0°, 180°	0.16 (±0.09)	0.16	0.24
GF, SIFT	0°, 180°	0.20 (±0.10)	0.20	0.60
GF, DTFPM	0°, 180°	0.20 (±0.10)	0.20	0.28

a result of this, we only consider the two perspective fusion. The results presented in Fig. 10.11 and Table 10.5 show that the best results are achieved when fusing palmar and dorsal view. This behaviour can be confirmed when analysing the occurrence of certain perspectives of the three-perspective fusion: Table 10.7 states that the palmar and dorsal region is part of most of the top 25 results. Therefore, we selected 0° and 180° for our combined fusion.

For MAF, the significance analysis (see Table 10.10) indicates that the performance increase from a two to a three feature extraction method fusion is significant but would lead to additional computational costs (for score-level fusion, every feature extraction method needs to be processed by the whole processing chain up to the comparison). Thus, we decided to include the two-feature extraction method MAF into our combined fusion strategy only. Furthermore, the results listed in 10.9 state that 88% of the best two-feature extraction method fusion combinations include one vein pattern based (MC, PC, GF) and one key-point based (SIFT, DTFPM) feature. Therefore, we analysed all possible combinations of those feature extraction methods using both, palmar and dorsal view. Table 10.11 lists the results of the CMPMAF. We evaluated all six possible combinations and arrived at a best EER of 0.04% with a confidence interval of 0.05% for the combined fusion of MC and SIFT for palmar and dorsal view. This result is 11 times better than the best single perspective result (MC at 0° with an EER of 0.44%). All other combinations also perform well. The worst result with an EER of 0.20% is achieved when fusing GF with either SIFT or DTFPM. This is still more than two times better than the best single perspective result. For the sake of completeness, we also calculated the results of the best 3-, 4- and 5-MAF combinations with the palmar and dorsal view. These results, listed in Table 10.12, show that the EER can be further improved. The best result with an EER of 0 is achieved when fusing the scores of all five feature types.

Table 10.13 compares the performance of the best combined two-perspective two-algorithm fusion with the best results of all other fusion strategies. One can see that the calculated χ^2 indicates a significant performance improvement with respect to the single perspective, the 2-MPF and the 2-MAF strategy. All other fusion strategies achieved about the same EER.

Table 10.12 Performance results: Fusion of vein pattern based with key-point based features for both, palmar and dorsal view. The best result is highlighted **bold face**

Feature types	Perspectives	EER	FMR1000	ZeroFMR
MC, SIFT, DTFPM	000°, 180°	0.04 (±0.04)	0.00	0.36
MC, PC, SIFT, DTFPM	000°, 180°	0.01 (±0.01)	0.00	0.12
MC, PC, GF, SIFT, DTFPM	000°, 180°	**0.00 (±0.00)**	0.00	0.00

Table 10.13 Comparison of the best two-perspective two-algorithm fusion combination to the best result of the other fusion strategies including the relative performance improvement, the factor, by which the EER decreased and the boundary χ^2 for significance

Fusion strategy	EER	EER CMPMAF	Rel. Perf. Impr. [%]	Factor	χ^2
Single perspective	0.44 (±0.15)	0.04 (±0.05)	1000	11	60.91
2-MPF	0.12 (±0.08)		200	3	7.31
3-MPF	0.04 (±0.04)		0	1	0.00
2-MAF	0.12 (±0.08)		200	3	7.31
3-MAF	0.04 (±0.05)		0	1	0.00
4-MAF	0.04 (±0.05)		0	1	0.00
5-MAF	0.04 (±0.05)		0	1	0.00

10.6.9 Results Discussion

The evaluation of the independent recognition performances for different projections revealed, that indeed the widely used palmar perspective performed best, followed by the dorsal one performing second best. The views in-between exhibit a slightly worse performance, which is still acceptable. Our results indicate that the presence of finger texture and finger knuckles has a positive influence on the recognition performance. Figure 10.9 shows, that the well-established feature extraction algorithms not only extract features resulting from the finger veins but also from the skin texture of the finger and therefore inherently fuse texture and vein structure. The best single view result was achieved using MC features at the palmar view with an EER of 0.44%.

However, the main objective of this work was to find a suitable trade-off between the number of involved views and feature extraction methods and the recognition performance. In order to arrive at a design decision for a multi-perspective finger vein capture device, several aspects have to be considered: first of all, the gain in recognition accuracy, followed by the production costs and complexity of the biometric capture device which is directly related to the number of involved views and finally the computational complexity of the finger vein recognition system including

the capturing time, i.e. the total processing time, which is related to both, the number of different views and the number of different feature extraction methods involved. Adding more perspectives or feature extraction methods increases the complexity of the finger vein sensor and the recognition tool chain. For every feature extraction method, all steps of the recognition tool chain from preprocessing to comparison need to be executed. Adding further perspectives additionally increases the cost and complexity of the capture device's hardware by the need of either adding more camera/illumination modules (one per perspective) or a rotator that moves camera and illumination module into position. Ideally, the number of perspectives and feature extraction methods are kept to a minimum. Furthermore, additional aspects like an improved resistance against presentation attacks and an increased robustness against environmental influences should be included too. Therefore, the decision on how many perspectives and feature extraction methods are used has to be a trade-off between added cost/complexity and improvement of the recognition performance. Our proposed design is based on the findings during the fusion evaluations.

The multi-perspective fusion results showed that by fusing two independent views, in particular, the palmar and dorsal view, a significant performance gain can be achieved. Adding a second perspective improved the recognition performance between a factor 2–3.5, depending on the feature extraction method. The best result with an EER of 0.12% was achieved using MC features fusing the palmar and dorsal view. Adding a third view still improves the performance compared to two perspectives, but not to the same extent (significance) as from a single perspective to the 2-MPF. In this case, the best result of 0.036% EER was achieved using MC when fusing 5°, 170° and 235°. A biometric capture device able to capture the palmar and the dorsal view simultaneously can be built without any moving parts. Two cameras and two illumination modules are sufficient. Each additional view poses noticeable extra costs in terms of hardware (camera and illumination modules) and complexity of the capture device construction. Therefore, one must decide whether the improvement in accuracy justifies the extra effort. As our results show, the performance improvement from a 2-MPF to a 3-MPF is not as significant as from a single perspective to a 2-MPF, a two-perspective capture device, capturing the vein structure from the palmar and dorsal region is the best choice.

For MAF, a single perspective capturing device is sufficient. Such a biometric capture device can be built in a more compact and less expensive manner than a multi-perspective one. Moreover, existing finger vein capture devices acquiring images of the palmar view, can be utilised to apply multi-algorithm fusion too. However, adding an additional feature type to the MAF increases the computational cost. The MAF results showed, that the fusion of different feature extraction methods per single view improves the overall performance remarkably as well. The best results were obtained when fusing vein pattern based algorithms (especially MC and PC) with key-point based methods (SIFT, DTFPM). The best MAF result with an EER of 0.04% was achieved when fusing MC, SIFT and DTFPM in the dorsal region. Including more feature types does not improve the performance compared to the 3-MAF. As the computational complexity for the calculation and comparison of DTFPM features are higher than for the other features types, and the performance increase compared

to the best 2-MAF utilising MC and SIFT (EER = 0.12%) features is not as significant as from a single perspective to the 2-MAF, the best MAF option is a 2-MAF including MC and SIFT features.

In a third step, we combined MPF and MAF. By using the best performing perspectives of the two-perspective approach (palmar and dorsal) and combining them with a vein pattern based (MC, PC or GF) and a key-point based method (SIFT or DTFPM), we were able to achieve an EER of 0.04% utilising MC and SIFT. This corresponds to an improvement by a factor of 11 compared to the best single perspective performance, while achieving similar results as for the best MPF and MAF strategies. Adding more feature types to the combined fusion strategy further improved the result. Combining palmar and dorsal view together with all five feature types resulted in a perfect result with EER, FMR1000 and ZeroFMR of 0%.

A multi-perspective finger vein capture device is more resistant against presentation attacks, especially against simple paper printout based attacks. Depending on the actual construction of the multi-perspective capture device, it might also be more robust against contamination (e.g. dust and dirt, sun protection lotion or hand cream on the finger surface) of the finger due to the fact that more than one perspective is captured. Hence, the two-perspective capture device is the preferred option over the single perspective, multi-algorithm fusion one regarding these additional aspects.

Taking all the above-mentioned considerations into account, especially the additional advantages provided by a multi-perspective capture device in terms of resistance against presentation attack and robustness against external influences, the most preferable option is to design a two-perspective capture device capturing the palmar and the dorsal view applying a two-algorithm fusion including MC and SIFT features, whereas by including only one view the advantages of multi-perspective recognition can not be retained. The second feature extraction method can be included without involving additional hardware costs just by extending the recognition tool chain and putting up with the extended processing time, which makes the two-feature version beneficial in any case. This proposed finger vein capture device set-up arrives at an EER of 0.04%, which is a performance gain by a factor of 11 compared to the best single-view, single feature performance. Hence, this option provides an optimal trade-off between recognition accuracy, construction costs and processing time.

10.7 Conclusion and Future Work

In this chapter, we introduced multi-perspective finger vein recognition. For most work reported in the literature, only the palmar view is used in finger vein recognition. However, as the finger is an elliptically shaped cylinder, there are several other views available all around the finger's longitudinal axis. In order to be able to exploit these additional views, a suitable biometric capture device able to capture these different views is necessary. This chapter is based on our previous work [2], where we constructed a rotating, multi-perspective finger vein capture device which was then utilised to capture a multi-perspective finger vein data set. Based on this dataset, the

recognition performance of each view was evaluated individually. Then we applied three different score-level fusion strategies, the first one fusing all possible pairs and triples of distinct views, the second one fusing all different feature combinations per each single view and the third one combining the first two approaches. The first strategy was employed to find out the best performing pairs and three tuples of views in terms of recognition performance. The more views are desired to be captured, the higher the complexity and production costs of a suitable biometric capture device. At some point (a certain number of desired views), only a rotating device is able to capture the desired views. A rotating capture device bears several disadvantages, e.g. it is more prone to failures and has an increased capturing time. If only a limited number of views is involved, the production costs and the complexity of the biometric capture device are kept low. The second strategy was applied to investigate the best feature extraction method combination per view. The third strategy, which combines the first two approaches, was applied to find out if the recognition results can be further improved.

The single view evaluation results confirmed that the widely used palmar perspective, followed by the dorsal one (not taking views which are only a few degrees off from the palmar and dorsal view into account), achieves the best performance in finger vein recognition. All the perspectives in-between the palmar and dorsal one exhibit an inferior recognition performance to the palmar and dorsal one. Regarding the multi-perspective score-level fusion it turned out that a fusion of only two perspectives increases the recognition performance significantly, where a fusion of the palmar and the dorsal view performed best. Adding a third perspective still improves the results over the two perspective ones, but not to the same extent as the two perspective ones. The multi-algorithm fusion achieves similar results to the multi-perspective one, arriving at an EER of 0.04% for the combination of three-feature extraction methods. A pure multi-algorithm fusion is preferable in terms of hardware costs and capture device's complexity but does not exhibit the advantages of a multi-perspective recognition in regards to resistance against presentation attacks and increased robustness against external influences. By applying both fusion approaches at the same time for the best performing two perspectives (palmar and dorsal) and the best performing two distinct feature extraction methods (MC, a vein pattern based one and SIFT, a key-point based one), we were able to improve the recognition performance by a factor of 11 compared to the best single view result, achieving an EER of 0.04%.

Regarding recognition performance, hardware costs, processing time and robustness against presentation attacks and external influences the overall best option is to go for the combined multi-perspective and multi-algorithm fusion. In particular, a finger vein capture device capturing the palmar and the dorsal view including MC and SIFT features in a combined fusion provides the best trade-off between the above mentioned considerations and is, therefore, our preferred design decision.

Future Work

The first step will be the construction of a combined multi-perspective and multi-algorithm type fusion finger vein capture device to prove its applicability in real-life

applications of finger vein recognition. We plan to do extended tests with this device, regarding presentation attacks, robustness against external influences like changing ambient conditions as well as subject-related influences.

Besides the capture device construction, our future work will include further analysis using our multi-perspective finger vein dataset. There are several other aspects besides the single perspective performance and the fusion of multiple perspectives which can be evaluated based on this dataset. One example is the robustness evaluation of different finger vein recognition algorithms against longitudinal finger rotation, which we already performed in a separate work [65]. We showed that this kind of rotation poses a severe problem for most algorithms. Since for our dataset the longitudinal rotation angle is known, we will test different techniques to compensate the finger rotation, either by estimating the rotation angle based on the captured images only or by using the known rotation angle and then applying a rotation compensating transform.

Another interesting question is if the best performing view is consistent across different subjects/fingers. To perform this analysis we will extend our dataset to contain at least 100+ subjects and then conduct a subject/finger based analysis to find out if the palmar perspective is the best one for all or at least a majority of the subjects/fingers or if there are significant differences.

Another field of interest is finger vein recognition in the 3D space. Therefore, we want to reconstruct a 3D model of the finger vein structure based on multiple images captured in different perspectives and apply different feature extraction and comparison strategies.

Acknowledgements This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 700259, project PROTECT—Pervasive and UseR Focused BiomeTrics BordEr ProjeCT.

References

1. Kono M (2000) A new method for the identification of individuals by using of vein pattern matching of a finger. In: Proceedings of fifth symposium on pattern measurement, Yamaguchi, Japan, pp 9–12
2. Prommegger B, Kauba C, Uhl A (2018) Multi-perspective finger-vein biometrics. In: Proceedings of the IEEE 9th international conference on biometrics: theory, applications, and systems (BTAS2018), Los Angeles, California, USA, pp 1–9
3. Raghavendra R, Busch C (2015) Exploring dorsal finger vein pattern for robust person recognition. In: 2015 international conference on biometrics (ICB), pp 341–348
4. University of Reading. PROTECT multimodal DB dataset, June 2017. http://projectprotect.eu/dataset/
5. Kauba C, Prommegger B, Uhl A (2018) Focussing the beam—a new laser illumination based data set providing insights to finger-vein recognition. In: Proceedings of the IEEE 9th international conference on biometrics: theory, applications, and systems (BTAS2018), Los Angeles, California, USA, pp 1–9

6. Kauba C, Prommegger B, Uhl A (2019) Openvein—an open-source modular multi-purpose finger-vein scanner design. In: Uhl A, Busch C, Marcel S, Veldhuis R (eds) Handbook of vascular biometrics. Springer Science+Business Media, Boston, MA, USA, pp 77–112
7. Gray H, Goss CM (1974) Anatomy of the human body. Am J Phys Med Rehabil 53(6):293
8. Tome P, Marcel S (2015) On the vulnerability of palm vein recognition to spoofing attacks. In: The 8th IAPR international conference on biometrics (ICB)
9. Tome P, Vanoni M, Marcel S (2014) On the vulnerability of finger vein recognition to spoofing
10. Lu Y, Yoon S, Park DS (2014) Finger vein identification system using two cameras. Electron Lett 50(22):1591–1593
11. Zhang Q, Zhou Y, Wang D, Hu X (2013) Personal authentication using hand vein and knuckle shape point cloud matching. In: 2013 IEEE Sixth international conference on biometrics: theory, applications and systems (BTAS). IEEE, pp 1–6
12. Qi Y, Zhou Y, Zhou C, Hu X, Hu X (2016) Vein point cloud registration algorithm for multi-pose hand vein authentication. In: 2016 IEEE international conference on identity, security and behavior analysis (ISBA). IEEE, pp 1–6
13. Matsuda Y, Miura N, Nagasaka A, Kiyomizu H, Miyatake T (2016) Finger-vein authentication based on deformation-tolerant feature-point matching. Mach Vis Appl 27(2):237–250
14. Brümmer N, De Villiers E (2013) The bosaris toolkit: theory, algorithms and code for surviving the new dcf.arXiv:1304.2865
15. Uhl A (2019) State-of-the-art in vascular biometrics. In: Uhl A, Busch C, Marcel S, Veldhuis R (eds). In: Handbook of vascular biometrics. Springer Science+Business Media, Boston, MA, USA, 3–62
16. Changzhou Songyang Machinery & Electronics New Technic Institute (2018) SY42STH47-1684A high torque hybrid stepping motor data sheet. https://www.pololu.com/file/0J714/SY42STH38-1684A.pdf. Accessed 20 June 2018
17. IDS Imaging Development Systems GmbH (2018) UI-ML1240-NIR NIR-enhanced industrial camera data sheet. https://de.ids-imaging.com/IDS/datasheet_pdf.php?sku=AB00184. Accessed 20 June 2018
18. Fujifilm Corporation (2018) Fujifilm HF9HA-1B product page. http://www.fujifilmusa.com/products/optical_devices/machine-vision/2-3-15/hf9ha-1b/index.html. Accessed 20 June 2018
19. Midopt Corporation (2018) MIDOPT LP780 NIR pass-through filter product page. http://midopt.com/filters/lp780/. Accessed 20 June 2018
20. International Electrotechnical Commission et al (2015) Multimodal and other multibiometric fusion. ISO/IEC TR 24722
21. Yang J, Jia Y (2012) A method of multispectral finger-vein image fusion. In: 2012 IEEE 11th international conference on signal processing (ICSP), vol 1. IEEE, pp 753–756
22. Guan F, Wang K, Mo H, Ma H, Liu J (2009) Research of finger vein recognition based on fusion of wavelet moment and horizontal and vertical 2dpca. In: 2nd international congress on image and signal processing, 2009 CISP'09. IEEE, pp 1–5
23. Yang J, Zhang X (2010) Feature-level fusion of global and local features for finger-vein recognition. In: 2010 IEEE 10th international conference on signal processing (ICSP). IEEE, pp 1702–1705
24. Gupta P, Gupta P (2015) An accurate finger vein based verification system. Digital Signal Process 38:43–52
25. Miura N, Nagasaka A, Miyatake T (2004) Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach Vis Appl 15(4):194–203
26. Kauba C, Piciucco E, Maiorana E, Campisi P, Uhl A (2016) Advanced variants of feature level fusion for finger vein recognition. In: Proceedings of the international conference of the biometrics special interest group (BIOSIG'16), Darmstadt, Germany, pp 1–12
27. Zhou Y, Kumar A (2010) Contactless palm vein identification using multiple representations. In: 2010 fourth IEEE international conference on biometrics: theory applications and systems (BTAS). IEEE, pp 1–6

28. Yang Y, Yang G, Wang S (2012) Finger vein recognition based on multi-instance. Int J Digital Content Technol Appl 6(11)
29. Park KR (2012) Finger vein recognition by combining global and local features based on svm. Comput Inf 30(2):295–309
30. Liu F, Yang G, Yin Y, Xi X (2013) Finger-vein recognition based on fusion of pixel level feature and super-pixel level feature. In: Biometric recognition. Springer, pp 274–281
31. Qin H, Qin L, Xue L, He X, Chengbo Y, Liang X (2013) Finger-vein verification based on multi-features fusion. Sensors 13(11):15048–15067
32. Lu Y, Yoon S, Park DS (2013) Finger vein recognition based on matching score-level fusion of gabor features. J Korean Inst Commun Inf Sci 38(2):174–182
33. Kauba C, Reissig J, Uhl A (2014) Pre-processing cascades and fusion in finger vein recognition. In: Proceedings of the international conference of the biometrics special interest group (BIOSIG'14), Darmstadt, Germany, Sept 2014
34. Miura N, Nagasaka A, Miyatake T (2007) Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans Inf Syst 90(8):1185–1194
35. Yang J, Shi Y, Yang J, Jiang L (2009) A novel finger-vein recognition method with feature combination. In: 2009 16th IEEE international conference on image processing (ICIP). IEEE, pp 2709–2712
36. Razzak MI, Yusof R, Khalid M (2010) Multimodal face and finger veins biometric authentication. Sci Res Essays 5(17):2529–2534
37. He M, Horng SJ, Fan P, Run RS, Chen RJ, Lai JL, Khan MK, Sentosa KO (2010) Performance evaluation of score level fusion in multimodal biometric systems. Pattern Recogn 43(5):1789–1800
38. Kang W, Chen X, Qiuxia W (2015) The biometric recognition on contactless multi-spectrum finger images. Infrared Phys Technol 68:19–27
39. Khellat-Kihel S, Abrishambaf R, Monteiro JL, Benyettou M (2016) Multimodal fusion of the finger vein, fingerprint and the finger-knuckle-print using kernel fisher analysis. Appl Soft Comput 42:439–447
40. Lin K, Han F, Yang Y, Zhang Z (2011) Feature level fusion of fingerprint and finger vein biometrics. In: International conference in swarm intelligence. Springer, pp 348–355
41. Park YH, Tien DN, Lee HC, Park KR, Lee EC, Kim SM, Kim HC (2011) A multimodal biometric recognition of touched fingerprint and finger-vein. In: 2011 international conference on multimedia and signal processing (CMSP), vol 1. IEEE, pp 247–250
42. Peng J, El-Latif AAA, Li Q, Niu X (2014) Multimodal biometric authentication based on score level fusion of finger biometrics. Optik-Int J Light Electron Opt 125(23):6891–6897
43. Peng J, Li Q, El-Latif AAA, Niu X (2015) Linear discriminant multi-set canonical correlations analysis (ldmcca): an efficient approach for feature fusion of finger biometrics. Multimed Tools Appl 74(13):4469–4486
44. Yang J, Zhang X (2012) Feature-level fusion of fingerprint and finger-vein for personal identification. Pattern Recogn Lett 33(5):623–628
45. Yang J, Zhong Z, Jia G, Li Y (2016) Spatial circular granulation method based on multimodal finger feature. J Electr Comput Eng
46. Yang Y, Lin K, Han F, Zhang Z (2012) Dynamic weighting for effective fusion of fingerprint and finger vein. PICA: Prog Intell Comput Appl 1(1):50–61
47. Kumar A, Zhou Y (2012) Human identification using finger images. IEEE Trans Image Proc 21(4):2228–2244
48. Yang W, Huang X, Liao Q (2012) Fusion of finger vein and finger dorsal texture for personal identification based on comparative competitive coding. In: 2012 19th IEEE international conference on image processing (ICIP). IEEE, pp 1141–1144
49. Yang W, Huang X, Zhou F, Liao Q (2014) Comparative competitive coding for personal identification by using finger vein and finger dorsal texture fusion. Inf Sci 268:20–32
50. Asaari MAM, Suandi SA, Rosdi BA (2014) Fusion of band limited phase only correlation and width centroid contour distance for finger based biometrics. Expert Syst Appl 41(7):3367–3382

51. Kang BJ, Park KR (2010) Multimodal biometric method based on vein and geometry of a single finger. IET Comput Vis 4(3):209–217
52. Xiaoming X, Yilong Y, Gongping Y, Xianjing M (2013) Personalized fusion method based on finger vein and finger contour. J Comput Res Dev 9:015
53. Lu Y, Xie SJ, Yoon S, Yang J, Park DS (2013) Robust finger vein ROI localization based on flexible segmentation. Sensors 13(11):14339–14366
54. Zuiderveld K (1994) Contrast limited adaptive histogram equalization. In: Heckbert PS (ed) Graphics gems IV. Morgan Kaufmann, pp 474–485
55. Zhao J, Tian H, Xu W, Li X (2009) A new approach to hand vein image enhancement. In: Second international conference on intelligent computation technology and automation, 2009. ICICTA'09, vol 1. IEEE, pp 499–501
56. Zhang J, Yang J (2009) Finger-vein image enhancement based on combination of gray-level grouping and circular gabor filter. In: International conference on information engineering and computer science, 2009. ICIECS. IEEE, pp 1–4
57. Choi JH, Song W, Kim T, Lee SR, Kim HC (2009) Finger vein extraction using gradient normalization and principal curvature. Proc SPIE 7251:1–9
58. Lowe DG (1999) Object recognition from local scale-invariant features. In: Proceedings of the seventh IEEE international conference on computer vision (CVPR'99), vol 2. IEEE, pp 1150–1157
59. Yang J, Yang J (2009) Multi-channel gabor filter design for finger-vein image enhancement. In: Fifth international conference on image and graphics, 2009. ICIG'09. IEEE, pp 87–91
60. Rohr K, Fornefett M, Stiehl HS (1999) Approximating thin-plate splines for elastic registration: Integration of landmark errors and orientation attributes. In: Biennial international conference on information processing in medical imaging. Springer, pp 252–265
61. Maio D, Maltoni D, Cappelli R, Wayman JL, Jain AK (2004) FVC2004: third fingerprint verification competition. In: ICBA, vol 3072. Springer, pp 1–7
62. Zhang L, Zhang L, Zhang D (2009) Finger-knuckle-print: a new biometric identifier. In: 2009 16th IEEE international conference on image processing (ICIP). IEEE, pp 1981–1984
63. Yang W, Yu X, Liao Q (2009) Personal authentication using finger vein pattern and finger-dorsa texture fusion. In: Proceedings of the 17th ACM international conference on multimedia. ACM, pp 905–908
64. Hofbauer H, Uhl A (2016) Calculating a boundary for the significance from the equal-error rate. In: Proceedings of the 9th IAPR/IEEE international conference on biometrics (ICB'16), pp 1–4
65. Prommegger B, Kauba C, Uhl A (2018) Longitudinal finger rotation—problems and effects in finger-vein recognition. In: Proceedings of the international conference of the biometrics special interest group (BIOSIG'18), Darmstadt, Germany

Part III
Sclera and Retina Biometrics

Chapter 11
Retinal Vascular Characteristics

Lukáš Semerád and Martin Drahanský

Abstract This chapter begins with a description of eye anatomy followed by the anatomy of retinas as well as the acquisition methods for obtaining retinal images. Our own device for capturing the vascular pattern of the retina is introduced in the following text. This chapter presents our aim to estimate the information present in human retina images. The next section describes the search for diseases found in retinal images, and the last section is devoted to our method for generating synthetic retinal images.

Keywords Synthetic retinal images · Vascular bed · Diabetic retinopathy · Hard exudates · Age-related macular degeneration · Druses · Exudates · Bloodstream mask · Information amounts · Bifurcations and crossings · Neural network · Human eye · Retina · Fundus camera · Slit lamp · Blind spot · Fovea · Device EYRINA · Retina recognition

11.1 Introduction

Just like several other biometric characteristics, our eyes are completely unique and, thus, can be used for biometric purposes. There are two core parts in our eyes that even show high biometric entropy. The first is the *iris* and the second is the *retina*, which is located at the backside of the eyeball and not observable by the naked eye. Recognition based on these two biometric characteristics is a relatively new method and little effort has been invested by industries.

The iris and the retina as elements inside the eye are very well protected against damage. The iris and retina patterns are unique to every individual (this also applies to monozygotic twins) and the structure is as follows (see Fig. 11.1) [1, 2]. The *cornea* is located at the front of the eye. It is a transparent connective tissue that, along with

L. Semerád (✉) · M. Drahanský
Faculty of Information Technology, Centre of Excellence IT4Innovations,
Brno University of Technology, Brno, Czechia
e-mail: isemerad@fit.vutbr.cz

M. Drahanský
e-mail: drahan@fit.vutbr.cz

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_11

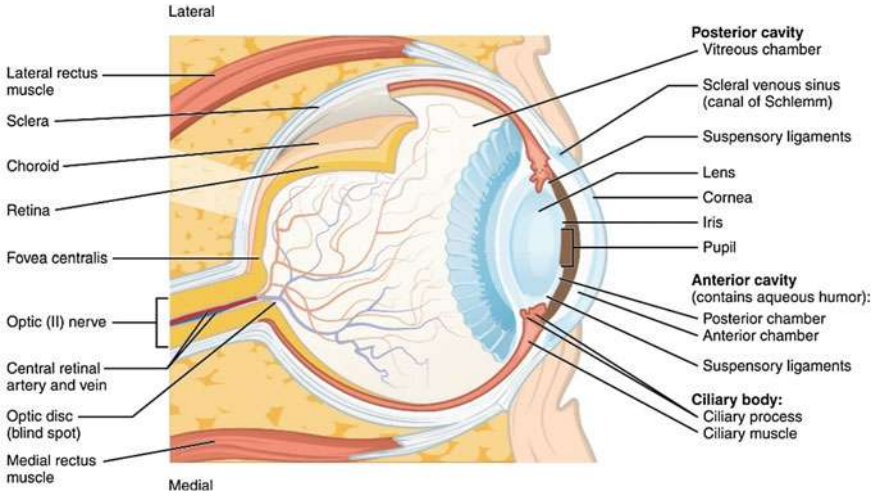


Fig. 11.1 Anatomy of the human eye [42]

the lens, allows the light to break into the eye. The *iris* has the shape of an annulus; it is a circularly arranged musculature that narrows/enlarges the pupil. The *pupil* is an opening in the middle of the iris, regulating the amount of light coming into the eye. The *sclera* is a white visible layer covering the entire eyeball, which passes into the cornea in the front. The retina is the inner part containing cells sensitive to light. It shows the image, much like a camera. The *optic nerve* carries many nerve fibres that enter the central nervous system.

There are two scientific disciplines that deal with eye characteristics—those are ophthalmology and biometrics. *Ophthalmology* is a medical discipline aimed at analysing and treating the health of the eye and its associated areas. In the field of *biometrics* (recognising an individual based on the unique biometric characteristics of the human body), the unique properties of the eye are not subject to change in time, and they are also so unique that it is possible to unequivocally identify two distinct individuals apart from each other in order to verify the identity of that person.

11.1.1 Anatomy of the Retina

The retina is considered to be a part of the *Central Nervous System* (CNS) [1, 2]. This is the only part of the CNS that can be observed noninvasively. It is a light-sensitive layer of cells located in the back of the eye with a thickness of 0.2–0.4 mm. It is responsible for sensing the light rays that hit it through the pupil, and a lens that turns and inverts the image. The only neurons that react directly to light are *photoreceptors*. These are divided into two main types: *cones* and *rods*. For adults, the retina covers approximately 72% of the inner eye. The entire surface of the retina

contains about 7 million cones and 75–150 million rods. This would compare the eye to a 157-megapixel camera. Rods are used to detect light and are capable of responding to the impact of one to two photons by providing black-and-white vision. Cones are used to detect colours and are divided into three types depending on which base colour they are sensitive to (red, green, blue), but these are less sensitive to light intensity [1, 2].

We can observe the two most distinctive points on an eye's retina—see Fig. 11.2. It is a *blind spot* (or an optical disc) and a *macula* (*yellow spot*) [1, 2]. A blind spot is the point where the optic nerve enters the eye; it has a size of about 3 mm^2 and lacks all receptors. So if the image falls into the blind spot, it will not be visible to a person. The brain often "guesses" how the image should look in order to fill in this place. On the other hand, the *macula* (*yellow spot*) [1, 2] is referred to as the sharpest vision area; it has a diameter of about 5 mm and the cones predominate it (it is less sensitive to light). This area has the highest concentration of light-sensitive cells, whose density decreases towards the edges. The centre of the macula is *fovea*, which is the term describing receptor concentration and visual acuity. Our direct view is reflected in this area. Interestingly enough, the macula (yellow spot) is not really yellow, but slightly redder than the surrounding area. This attribute, however, was given by the fact that yellow appears after the death of an individual.

The retina vessel's apparatus is similar to the brain, where the structure and venous tangle remain unchanged throughout life. The retina has two main sources of blood: the *retinal artery* and the *vessels*. Larger blood flow to the retina is through the blood vessel that nourishes its outer layer with photoreceptors. Another blood supply is provided by the retinal artery, which primarily nourishes the inside of the retina. This artery usually has four major branches.

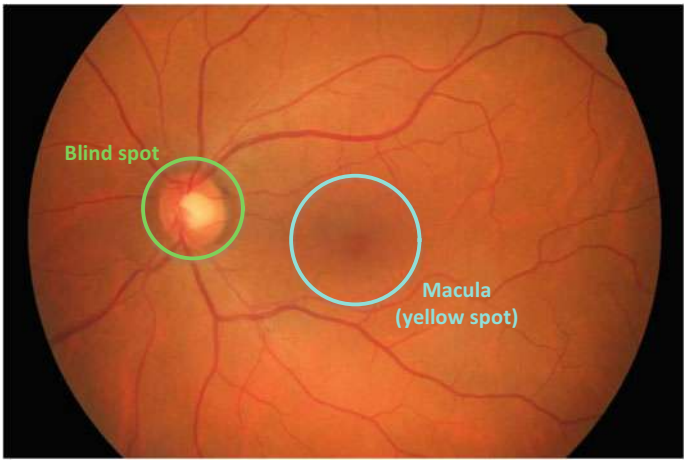


Fig. 11.2 A snapshot of the retina taken by the fundus camera

The retina located inside the eye is well protected from external influences. During life, the vessel pattern does not change and is therefore suitable for biometric purposes.

The retina acquires an image similar to how a camera does. The beam passing through the pupil appears in the focus of the lens on the retina, much like the film. In the medical field, specialised optical devices are used for the visual examination of the retina.

The iris is beyond the scope of this chapter, however, some interesting works include [3–5].

11.1.2 History of Retinal Recognition

In 1935, ophthalmologists *Carleton Simon* and *Isidore Goldstein* discovered eye diseases where the image of the bloodstream in two individuals in the retina was unique for each individual. Subsequently, they published a journal article on the use of vein imaging in the retina as a unique pattern for identification [6]. Their research was supported by Dr. Paul Tower, who in 1955 published an article on studying monozygotic twins [7]. He discovered that retinal vessel patterns show the least resemblance to all the other patterns examined. At that time, the identification of the vessel's retina was a timeless thought.

With the concept of a simple, fully automated device capable of retrieving a snapshot of the retina and verifying the identity of the user, Robert Hill, who established EyeDentify in 1975, devoted almost all of his time and effort to this development. However, functional devices did not appear on the market for several years after [8, 9].

Several other companies attempted to use the available fundus cameras and modify them to retrieve the image of the retina for identification purposes. However, these fundus cameras had several significant disadvantages, such as the relatively complicated alignment of the optical axis, visible light spectra, making the identification quite uncomfortable for the users, and last but not least, the cost of these cameras was very high.

Further experiments led to the use of Infrared (IR) illumination, as these beams are almost transparent to the choroid that reflect this radiation to create an image of the eye's blood vessels. IR illumination is invisible to humans, so there is also no reduction in the pupil diameter when the eye is irradiated.

The first working prototype of the device was built in 1981. The device with an eye-optic camera used to illuminate the IR radiation was connected to an ordinary personal computer for image capture analysis. After extensive testing, a simple correlation comparison algorithm was chosen to be the most appropriate.

After another four years of hard work, EyeDentify Inc. launched *EyeDentification System 7.5*, where verification is performed based on the retina image and the PIN entered by the user with the data is stored in the database [8, 9].

The last known retinal scanning device to be manufactured by EyeDentify Inc. was the *ICAM 2001*. This device might be able to store up to 3,000 subjects, having a storage capacity of up to 3,300 history transactions [8]. Regrettably, this product was withdrawn from the market because of user acceptance and its high price. Some other companies like Retica Systems Inc. were working on a prototype of retinal acquisition devices for biometric purposes that might be much easier to implement into commercial applications and might be much more user friendly. However, even this was a failure and the device did not succeed in the market.

11.1.3 Medical and Biometric Examination and Acquisition Tools

First of all, we will start with the description of existing medical devices for retinal examination and acquisition, followed by biometric devices. The medical devices provide high-quality scans of the retina, however, the two major disadvantages are predetermining these devices to fail within the biometric market—first, because of their very high price, which ranges from the thousands (used devices) to the tens of thousands of EUR; second, because of their manual or semi-automatic mode, where medical staff is required. So far, there is no device on the market that can scan the retina without user intervention, i.e. something that is fully automatic. We are working on this automatic device, but its price is not yet acceptable for the biometric market.

11.1.3.1 Medical Devices

The most commonly used device for examining the retina is a *direct ophthalmoscope*. When using an ophthalmoscope, the patient's eye is examined from a distance of several centimetres through the pupil. Several types of ophthalmoscopes are currently known, but the principle is essentially the same: the eye of the investigated data subject and the investigator is in one axis, and the retina is illuminated by a light source from a semipermeable mirror, or a mirror with a hole located in the observation axis at an angle of 45° [10]. The disadvantage of a direct ophthalmoscope is a relatively small area of investigation, the need for skill when handling, and patient cooperation.

For a more thorough examination of the eye background, the so-called *fundus camera* is used (as shown in Fig. 11.3), which is currently most likely to have the greatest importance in retina examinations. It allows colour photography to capture almost the entire surface of the retina, as can be seen in Fig. 11.2. The optical principle of this device is based on so-called indirect ophthalmoscopy [10]. Fundus cameras are equipped with a white light source (i.e. a laser) to illuminate the retina and then scan it with a CCD sensor. Some types can also find the centre of the retina and automatically focus it, using a frequency analysis of the scanned image.

Fig. 11.3 (Left) Slit lamp example [43]; (right) example of a non-mydriatic fundus camera [44]

The main ophthalmoscopic examination methods of the anterior and posterior parts of the eye include direct and indirect ophthalmoscopy as well as the most widely used examination, a *slit lamp* (see Fig. 11.3 on the left), which makes it possible to examine the anterior segment of the eye using so-called *biomicroscopy*. A *fundus camera*, sometimes referred to as a *retinal camera*, is a special device for displaying the posterior segment of the optic nerve, the yellow spots and the peripheral part of the retina (see Fig. 11.3 on the right). It works on the principle of indirect ophthalmoscopy where a source of primary white light is built inside the instrument. The light can be modified by different types of filters, and the optical system is focused on the data subject's eye, where it is reflected from the retina and points back to the fundus camera lens. There are mydriatic and non-mydriatic types that differ in whether or not the subject's eye must be taken into mydriasis. The purpose of mydriasis is to extend the human eye's pupil so that the "inlet opening" is larger, allowing one to be able to read a larger portion of the retina. Of course, non-mydriatic fundus cameras are preferred because the data subject can immediately leave after the examination and can drive a motor vehicle, which is not possible in the case of mydriasis. However, mydriasis is necessary for some subjects. The price of these medical devices is in the order of tens of thousands of EUR, which is determined only by medically specialised workplaces.

The mechanical construction of the optical device is a rather complex matter. It is clear that the scanning device operates on the principle of medical eye-optic devices. These so-called retinoscopes, or fundus cameras, are relatively complicated devices and the price for them is quite high as well.

The principle is still the same as it is for a retinoscope, where a beam of light is focused on the retina and the CCD camera scans the reflected light. The beam of light from the retinoscope is adjusted so that the eye lens focuses on the surface of the retina. This reflects a portion of the transmitted light beam back to the ophthalmic

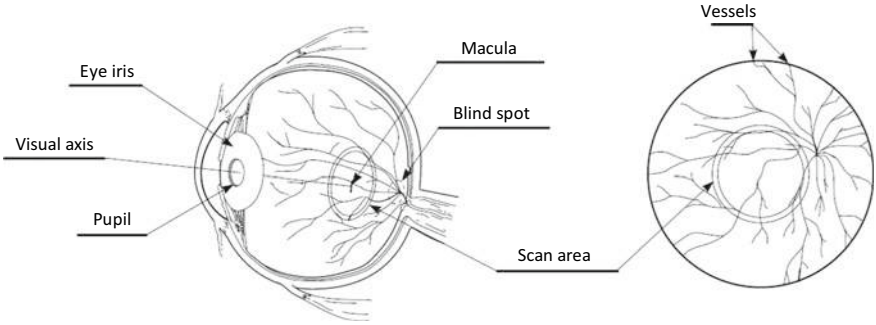


Fig. 11.4 The functional principle for obtaining a retinal image of the eye background

lens that then readjusts it, the beam leaving the eye at the same angle below which the eye enters (return reflection). In this way, an image of the surface of the eye can be obtained at about 10° around the visual axis, as shown in Fig. 11.4. The device performed a circular snapshot of the retina, mainly due to the reflection of light from the cornea, which would be unusable during raster scanning.

11.1.3.2 Biometric Devices

The first products from EyeDentify Inc. used a relatively complicated optical system with rotating mirrors to cover the area of the retina—this system is described in U.S. Pat. No. 4,620,318 [11]. To align the scan axis and the visual axis, the so-called UV-IR cut filters (*Hot Mirrors*—reflect infrared light and passes through the visible light) are used in the design. A schematic drawing of the patent is in Fig. 11.5. The distance between the eye and the lens was about 2–3 cm from the camera. The alignment system on the optical axis of the instrument is an important issue, and it is described in more detail in U.S. Pat. No. 4,923,297 [12].

Newer optical systems from EyeDentify Inc. were much easier and had the benefits of repairing optical axes with less user effort than the previous systems. The key part

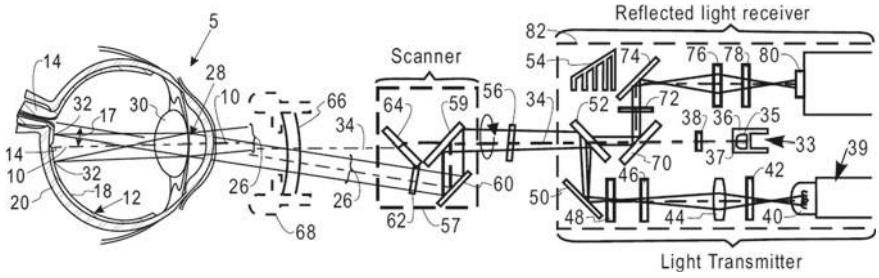


Fig. 11.5 The first version of the EyeDentification System 7.5 optical system [12]

Fig. 11.6 (Left) EyeDentify [9]; (right) EyeDentificationSystem [45]

was a rotating scanning disc that carried multifocal Fresnel lenses. This construction is described in U.S. Pat. No. 5,532,771 [13].

A pioneer in developing these identification systems is primarily EyeDentify Inc., who designed and manufactured the EyeDentification System 7.5 (see Fig. 11.6) and its latest ICAM 2001 model, which was designed in 2001. Other companies are Retinal Technologies, known since 2004 as Retica Systems, but details of their system are not known. The company TPI (Trans Pacific Int.) has recently offered an ICAM 2001-like sensor, but there is no longer any information about it available.

11.1.3.3 Device EYRINA

At the end of this subsection, we will devote our attention to our own construction of an interesting and nonexistent device that can be used in both the field of biometric systems and in the field of ophthalmology—we call it EYRINA. This device is a fully automatic non-mydriatic fundus camera. Many years ago, we started with a simple device (see Fig. 11.7 on the left), but over time, we came to the third generation of the device (see Fig. 11.7 on the right). We are now working on the fourth generation

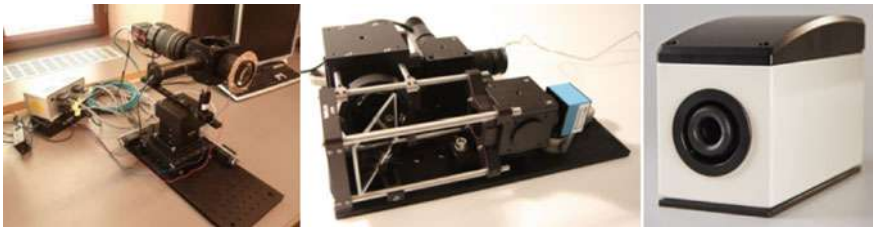


Fig. 11.7 A non-mydriatic fundus camera—first generation left, second generation middle and third generation right

of this device that will be completely automatic. The original concept was focused only on the retina (a direct view in the optical axis of the eye), then we arrived (second generation) to retrieve the retina and the iris of the eye in one device, while the third and fourth generation is again focused solely on the retina of the eye. The third generation can already find the eye in the camera, move the optical system to the centre of the image (alignment of the optical axis of the eye and the camera) and take pictures of the eye retina (in the visible spectrum) to shoot a short video (in the infrared spectrum). The fourth generation will be able to capture almost the entire ocular background (not just a direct view in the optical axis of the eye) and combine the image into one file. This will, of course, be associated with software that can already find the macula and blind spot, arteries and vessels, detect and extract bifurcations and crossings and find areas with potential pathological findings while we can detect exudates/druses and haemorrhages, including the calculation of their overall area. In the future, we will focus on the reliability and accuracy of detectors and extractors, including other types of illnesses that will be in the interest of ophthalmologists.

The central part of the third generation built two tubes with optics that can compensate the diopter distortion approx. ± 10 D. The left tube is connected to the motion screw and the NEMA motor, i.e. we were able to move the frontal (left) tube. The eye is very close to the eyebrow holder. Between these two tubes, we have a semipermeable mirror. Under this mirror is an LED for making the look of the patient to be fixed on a concrete position. The illumination unit is placed behind the mirror on the covering unit. Behind the background (right) tube is a high-resolution camera. The mainboard and PCBs are placed in the back of the fundus camera, where the connectors and cables are placed as well. The connection is done using a USB cable to the computer.

The image of a real eye from the second version of EYRINA could be found in Fig. 11.8. Now, we just used an ophthalmologic eye phantom for version 3.

Version 3 was able to automatically capture a direct view to the eye, i.e. pupil detection, focusing and taking pictures automatically; however, it is not possible to

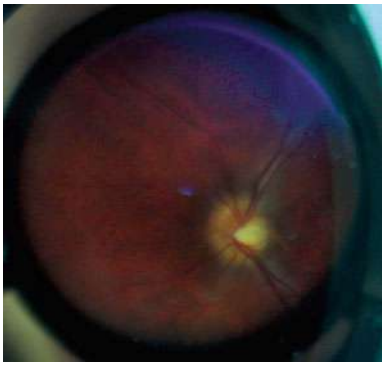


Fig. 11.8 Retinal image of a real retina from the second version of EYRINA

Fig. 11.9 Model of the construction of a fourth-generation device

capture images for retinal images stitching, and if the user has not centred the optical axis of his/her eye with the optical axis of the camera system, the view to the eye is not correct. The new version 4 has a 5-axes manipulator, which is able to find the centred position of both optical axes (eye and camera) automatically. The other new parts are the compensation of diopter distortion ±12 D (with additional rings for up to ±30 D), automatic composition of scanned images, automatic recognition of the optic disc, macula and selected pathologies, and a Wi-Fi/USB connection. The model of the fourth version of this fundus camera is visible in Fig. 11.9. This camera should be ready for laboratory installation in Autumn 2019.

11.1.4 Recognition Schemes

In the introductory chapter is an overview about the existing work on retina recognition. There are several schemes that could be used for the recognition of retinal images. For example, there are different approaches for retina image biometric recognition. Farzin [8] and Hill [9] segment the blood vessels, from which it generates features and stores up to 256 12-bit samples reduced to a reference record of 40 bytes for each eye. Contrast information is stored in the time domain. Fuhrmann and Uhl [14] extract vessels, from which the retina code is obtained. This is a binary code that describes the vessels around the optical disc.

The first idea for recognition (described in Chap. 3.1) is based on the work of Arakala et al. [15], where the biometric entropy of retina and recognition based on area around the optical disc is calculated. We have extended this area and started using it for identification. Our idea of localisation points to the retinal vascular bed and is based on the similarity of the structure with the papillary lines in the fingerprints. There, bifurcation, termination, position and direction of the minutiae are detected. In retinas, blood vessels are not as severely terminated as in fingerprints, gradually diminishing until lost. Therefore, we do not detect termination. On the contrary, the

bifurcation here is terminated. In addition, the complicated structure of the several layers of blood vessels over one another is virtually crossing the vessels in the image. It is not easy to know what is crossing and what is bifurcation, so we detect these features together. We then base biometric recognition on these points.

We are also looking for the centre of the blind spot and the fovea. We created a coordinate system with the centre in the middle of abscissa between the centre of the blind spot and the centre of the fovea. The individual points are then represented by the angle and distance in these units, i.e. the results are a set of vectors showing the concrete place in the retinal image. Thus, we are invariant to the different way of acquiring the retina, since the optical axes of the eye and the sensing device may not always be unified.

In the retina, the situation is relatively simple because the algorithms are searching the image for *bifurcations* and *crossings* of the retinal vascular system, whose positions clearly define the biometric instance (i.e. the retina pattern). An example is shown in Fig. 11.10. Recognition becomes problematic when a stronger pathological phenomenon (e.g. a haemorrhage) occurs in the retina that affects the detection and extraction of bifurcations and crossings. For biometric systems, it should be noted that their use also includes the disclosure of information about their own health status since, as mentioned above, a relatively large amount of information on human health can be read from the image of an iris, and that is, especially, the case for a retina as well. It is therefore up to each of us in regard to how much we will protect this private information and whether or not we will use the systems. However, if the manufacturer guarantees that the health information does not get stored, and only the unique features are stored (not the image), then the system may be used based on data protection legislation (e.g. GDPR).

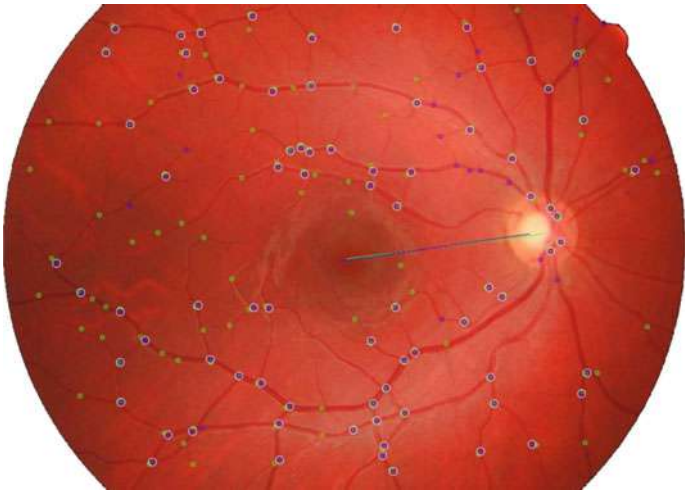


Fig. 11.10 Extracted features (bifurcations and crossings, incl. the connection of macula and blind spot) in the retina [37]

11.1.5 Achieved Results Using Our Scheme

The aim of this work was to compare manually marked and automatically found bifurcations/crossings using our application, *RetinaFeatureExtractor*, and find out the success of the automatic search. First, we created a Python *extract_features.py* script that reads retina images from the selected folder and uses *RetinaFeatureExtractor* to find bifurcations/crossings for each image and save them into text files in the same hierarchy as the source images. After obtaining a set of automatically found bifurcations/crossings, we designed an algorithm for comparing them to manually selected bifurcations/crossings (ground truth). We then created a Python *comparison.py* script that compares the found bifurcations.

The algorithm automatically finds bifurcations/crossings that are paired with the manually found bifurcations/crossings. The algorithm works as follows:

- Converts the found bifurcations/crossings to the same coordinate system.
- For each manually found bifurcation/crossing, it locates around the size t candidates for pairing and remembers their distance.
- If the number of manually found bifurcations/crossings and candidates for pairing is not the same, the smaller of the sets is completed with placeholders.
- Builds a complete bipartite graph where one disjunctive set of vertices is created by the manually found bifurcations/crossings, and the second by the candidates. It also set the price of edges between the manually found bifurcations/crossings and their corresponding candidates and computes the distance. For other edges, it sets the value from the interval $<t+1, \infty)$.
- Finds the minimum matching in the bipartite graph.
- From paired pairs, it removes those where one of the bifurcations/crossings is a placeholder, or those pairs of them where the distance is greater than t.
- Calculates the percentage of the manually marked points that have been paired.

In both sets, the positions of the blind and yellow spot are given. It is in files with manually marked bifurcations/crossings and the blind spot is marked with a rectangle, and in automatically found bifurcations/crossings it is a circle. The yellow spot is in both file types marked with a circle. Bifurcations/crossings are expressed by r and ψ. The r is the distance from centre of the blind spot, but it is recalculated so that the distance from the centre of blind spot to the centre of the yellow spot is 1. The ψ stands for the angle from the blind spot with zero value to the centre of yellow spot.

We decided to convert the found bifurcations/crossings into a Cartesian coordinate system. We needed to calculate the distance between the centre of the blind spot (hereafter C_{BS}) and yellow spot (hereafter C_{YS}). In the file with manually marked bifurcations/crossings, only the centre of the rectangle indicating the blind spot had to be calculated; in the expression of the circles, their centre was already contained. We then calculated their Euclidean distance (hereinafter d). Afterwards, we calculated the angle between the centres of both spots (hereafter α) according to Eq. (1.1).

$$\alpha = arctg2((y.C_{YS} - y.C_{BS}), (x.C_{YS} - x.C_{BS})). \tag{1.1}$$

Using Eq. (1.2), we calculated the bifurcation/crossing distance from the blind spot:

$$v = r \cdot d \tag{1.2}$$

Then, using Eqs. (1.3) and (1.4), we calculated the coordinates dx and dy:

$$dx = d \cdot \cos(\psi + \alpha), \tag{1.3}$$

$$dy = d \cdot \sin(\psi + \alpha). \tag{1.4}$$

The resulting point of bifurcation/crossing in the Cartesian system is obtained as $\left[dx + x.C_{BS}; dy + y.C_{BS}\right]$.

We saved the converted points to the list and used their position in the list that we could use as ID to compile disjunctive sets. We assigned a placeholder ID with a value of -1. To calculate the minimum pairing we used the fact that this problem can be converted to the problem of integer programming [16]. After the calculation, we obtained the edges between the individual vertices of the graphs and we could calculate how many manually found bifurcations/crossings were paired. The resulting image for the comparison is shown in Fig. 11.11.

We used three publicly available databases: Drions [17], Messidor [18] and HRF (High-Resolution Fundus Image Database) [19].

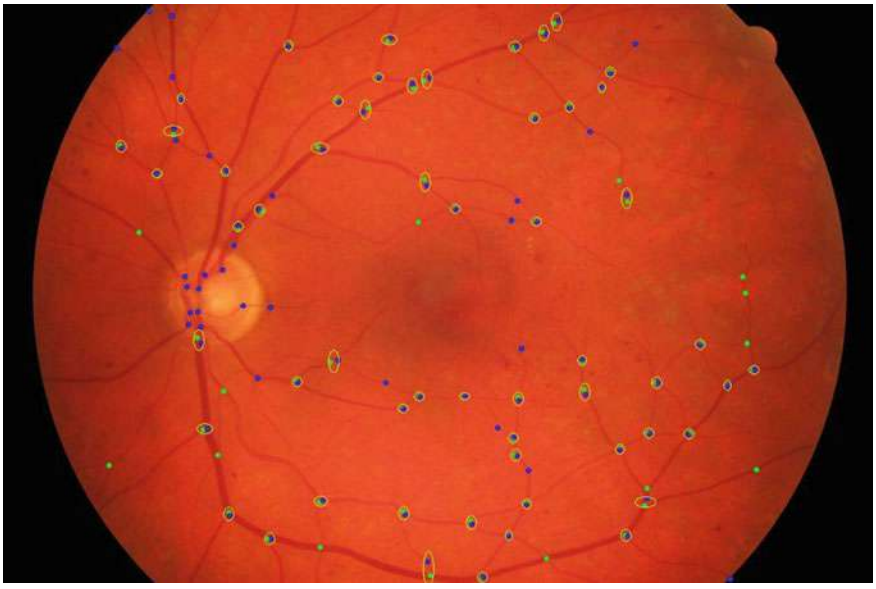


Fig. 11.11 The resulting image for the comparison of manually and automatically found bifurcations/crossings

Table 11.1 The summarised results for manual and automatic bifurcation/crossing detection

Database	Average success rate [%]	Average marking error [%]	Average point spacing [px]
Drions images	62	47	5.25
HRF	66	31	4.55
Messidor (Base12)	74	65	4.25
Messidor (Base13)	79	63	5.01
Messidor (Base22)	61	45	4.65
Messidor (Base3)	82	72	4.95
Average	70.67	53.83	4.78

The *Drions* database consists of 110 colourised digital retinal images from the Ophthalmology Service at Miguel Servet Hospital, Saragossa (Spain). Images are in RGB JPG format, and the resolution is 600 × 400 with 8 bits/pixel [17]. The *Messidor* database originally contains 1,200 eye fundus colour numerical images of the posterior pole. Images were acquired by 3 ophthalmologic departments. The images were captured using 8 bits per colour plane at 440 × 960, 240 × 488 or 304 × 536 pixels. The *HRF* database contains 15 images of healthy patients, 15 images of patients with diabetic retinopathy and 15 images of glaucomatous patients.

We used images from these databases to compare our manually selected and automatically marked bifurcations and crossings in them.

The results are summarised in Table 11.1.

At the same time, we have modified and improved our algorithm that we tested on the VARIA database [20], which contains 233 images from 139 individuals. We conducted a classic comparison of found bifurcations/crossings that correspond to the fingerprint method. The DET curve is shown in Fig. 11.12.

ALG-1 is an elementary algorithm that only shrinks images to one-fifth, smoothes them, and equalises the histogram.

ALG-3 processes images as follows: after processing ALG-1, it detects an optical disc and fovea and then aligns the images to a uniform plane. Next, it highlights the vessels in the image and crops the compared area around the optical disc.

ALG-2 compared to ALG-3 does not cut the image, only on the optical disc area. Moreover, the resulting image is applied to edge detection.

Source code of algorithms is available on [21].

11.1.6 Limitations

There are some limitations in retinal biometrics that discourage greater use in biometric systems. There is currently no system that can remove these shortcomings to a greater extent [9]:

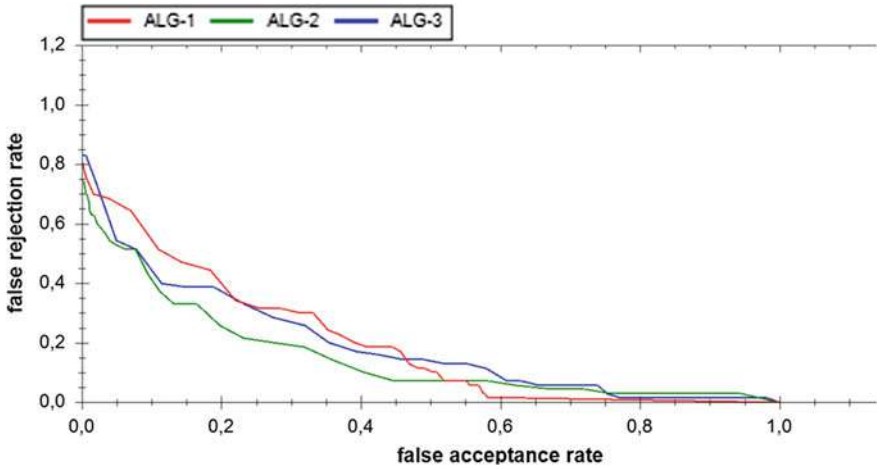


Fig. 11.12 The DET curve for our three versions of the algorithm *RetinaFeatureExtractor*

- *Fear of eye damage*—The low level of infrared illumination used in this type of device is completely harmless to the eye, but there is a myth among the lay public that these devices can damage the retina. All users need to be familiar with the system in order to gain confidence in it.
- *Outdoor and indoor use*—Small pupils can increase the false rejection rate. Since the light has to pass through the pupil twice (once in the eye, then outward), the return beam can be significantly weakened if the user's pupil is too small.
- *Ergonomics*—The need to come close to the sensor may reduce the comfort of using the device.
- *Severe astigmatism*—Data subjects with visual impairments (astigmatism) are unable to focus the eye onto the point (a function comparable to measuring the focusing ability of the eye for an ophthalmologist), thus avoiding the correct generation of the template.
- *High price*—It can be assumed that the price of the device, especially the retroviral optical device itself, will always be greater than, for example, the price of fingerprint or voice recognition capture devices.

The use of retinal recognition is appropriate in areas with *high-security requirements,* such as nuclear development, arms development, as well as manufacturing, government and military facilities and other critical infrastructure.

11.2 Eye Diseases

The main focus of this chapter is on ophthalmology in regard to examining the retina of the eye, taking into account, of course, the overall health of the eye (e.g. cataracts or

increased intraocular pressure). Within the retina is a relatively large line of diseases and damages that interest medical doctors, but they are detailed in an encyclopaedia of ophthalmology consisting of hundreds of pages (e.g. [22] (1,638 pages) or [23] (2,731 pages)). The largest group is diabetes and Age-related Macular Degeneration (ARMD). Occasionally exudates/druses or haemorrhages (bleeding or blood clots) appear in the retina; however, as mentioned above, potential damage (e.g. perforation or retinal detachment) or retinal disease is such a matter.

In comparison with other biometric characteristics (e.g. fingerprints, the vascular patterns of the hand or finger), the role of diseases connected to a concrete biometric information career (e.g. finger, hand) plays a very important role. It is not only the *ageing* factor, which can bring some changes into the retinal image sample, but the pathologies on the retina can disable the subject, making them unable to use the biometric system. The most common disease manifestations are related to diabetes mellitus and ARMD, whereas these pathologies (e.g. haemorrhages and aneurisms) can change the quality of the image so much that the vascular pattern is partially covered or completely invisible. Therefore, a short description of the most important and the most widespread retinal diseases are mentioned and shortly described to get the feeling of how much they can decrease the biometric performance of the recognition algorithms. These diseases are expected to influence recognition scheme described in the Sect. 11.1.4. The impact on biometric recognition is based on our observations and has no empirical evidence.

Diabetes mellitus (DM, diabetes) [24] is a disease characterised by elevated blood glucose (hyperglycemia) due to the relative or absolute lack of insulin. Chronic hyperglycemia is associated with long-lasting damage, dysfunction and failure of various organs in the human body—especially, the eyes, kidneys, heart and blood vessels. Most types of diabetes [24] fall into two broader categories: type 1 and type 2.

While diabetes mellitus (diabetes) has been described in ancient times, *diabetic retinopathy* [25, 26] is a disease discovered relatively late. Diabetic Retinopathy (DR) is the most common vascular disease of the retina. It is a very common late complication of diabetes and usually occurs after more than 10 years of having diabetes.

Diabetic retinopathy occurs in several stages. The first stage can only be detected by fluorophotometry. The next stage is called *simple*, *incipient* or *Non-proliferative Diabetic Retinopathy* (NPDR). This is characterised by the formation of small *micro-aneurysms* (vessel bulging), which often crack and result in another typical symptom—the formation of small *intrarethral* or *pre-renal haemorrhages*. Because the micro-aneurysms and haemorrhages include blood, their colour is very similar to the vessel pattern colour, i.e. if larger areas in the eye are affected by these diseases, it is expected to the biometric recognition performance drops down, because the recognition of retinal images is based on the comparison of vessel structures for both images. Microinfarcts have a white colour, a fibrous structure, and are referred to as “cotton stains”. If the capillary obliteration is repeated at the same site, heavy exudates arise. These are a sign of chronic oxygen deficiency. They are yellow, sharply bounded,

and formed by fat-filled cells. This stage is called *Proliferative Diabetic Retinopathy* (PDR) [25, 26].

Micro-aneurysms (MA) [25, 26] are considered to be basic manifestations of diabetic retinopathy. Although micro-aneurysms are characteristic of diabetic retinopathy, they cannot be considered a pathologic finding for this disease. They can, however, manifest in many other diseases. MAs are the first lesions of the DR that are proven by biomicroscopic examination. The flowing MA leads to the formation of edema and annularly deposited exudates. Their size is between 12 μm and 100 μm. These are round dark red dots, which are very difficult to distinguish from a micro-haemorrhage. Unlike these, they should have more bordered edges. If their size is greater than 125 μm, it must be taken into account that they may be micro-haemorrhages. As mentioned above, their colour is similar to the vascular pattern and it is expected that they influence biometric recognition performance.

Depending on the location within the retina, we can distinguish *haemorrhage* intraretinally and sub-retinally [25, 26]. Haemorrhages occur secondarily as a result of the rupture of micro-aneurysms, veins and capillaries. Spotted haemorrhages are tiny, round red dots kept at the level of capillaries and only exceptionally deeper (see Fig. 11.13 right). Their shape is dependent on their location, but also on the origin of the bleeding. Spontaneous haemorrhages have the characteristic appearance of stains and their colour is light red to dark. As mentioned above, their colour is similar to a vascular pattern and it is expected that they influence the biometric recognition performance.

Hard exudates (Fig. 11.13 left) [25, 26] are not only characteristic of diabetic retinopathy. They are also found in many other diseases. Hard-dotted exudates are round, clear yellow dots. They create different clusters with a pronounced tendency to migrate. Stubborn hard exudates are predominantly surface-shaped and have the shape of a hump. The colour of this pathology is different from the vascular structure, so it does not affect biometric recognition performance, but it can affect the ability of preprocessing algorithms to prepare the image for venous structure extraction.

Soft exudates (Fig. 11.13 left) [25, 26] are considered to be a typical manifestation of diabetic retinopathy, but it can also be found in other diseases. They result from

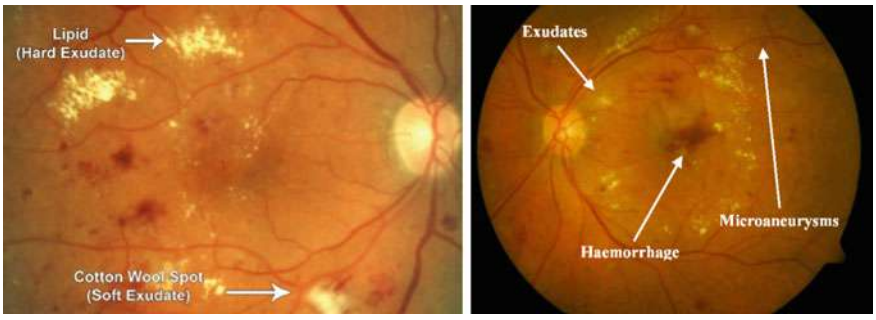


Fig. 11.13 (Left) Hard and soft exudates [46] and (right) haemorrhage and micro-aneurysms [47]

arteriolar occlusions (closures) in the nervous retinal layer. They are often accompanied by a plague-like haemorrhage. There are often extended capillaries along the edges. The colour of this pathology is different from the venous structure, so it does not affect biometric recognition performance, but it can affect the ability of preprocessing algorithms to prepare the image for venous structure extraction.

Age-related Macular Degeneration (ARMD) [27–29] is a multifactorial disease. The only reliably proven cause of ARMD development is age. ARMD is characterised by a group of lesions, among which we classically include the accumulation of deposits in the depth of the retina—drunia, neovascularisation, fluid bleeding, fluid accumulation and geographic atrophy.

Based on clinical manifestations, we can distinguish between dry (atrophic, non-exudative) and wet (exudative, neovascular) disease [27–29]. The dry form affects less than 90% of patients and is about 10% moist.

Dry form—This is caused by the extinction of the capillaries. Clinical findings found that in the dry form of ARMD druses, there are changes in pigmentation and some degree of atrophy. The terminal stage is called *geographic atrophy*. The *druses* are directly visible yellowish deposits at the depth of the retina, corresponding to the accumulation of pathological material in the inner retinal layers. The druses vary in size, shape, appearance. Depending on the type, we can distinguish between soft and hard druses. Soft druses are larger and have a "soft look". They also have a distinct thickness and a tendency to collapse. Druses that are less than half the diameter of the vein at the edge of the target, and they are referred to as small (up to 63 μm) and respond to hard druses. Druses ≥125 μm are large and respond to soft druses. Hard druses are not ophthalmoscopically trapped up to 30–50 μm [30]. *Geographic atrophy* is the final stage of the dry, atrophic form of ARMD—see Figs. 11.14 and 11.15. It appears as a sharp, borderline oval or a circular hypopigmentation to depigmentation or direct absence of retinal pigment epithelium. Initially, the atrophy is only light, localised, and gradually spreading often in the horseshoe shape around the fovea. The development of atrophy is related to the presence of druses and, in particular, their collapse or disappearance [27–29].

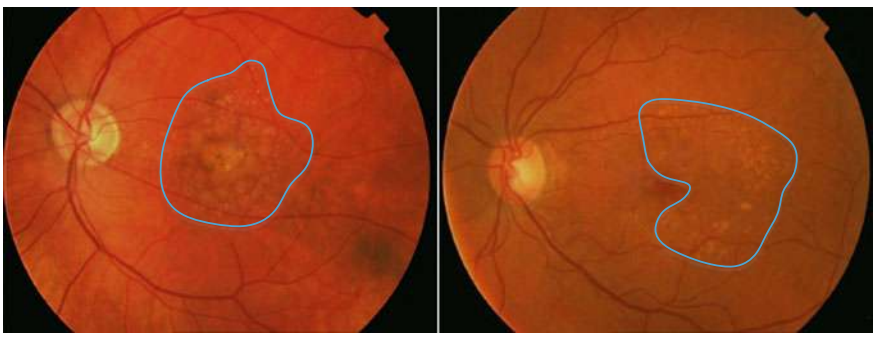


Fig. 11.14 (Left) ARMD—soft druses [48]; (right) ARMD—hard druses [28]

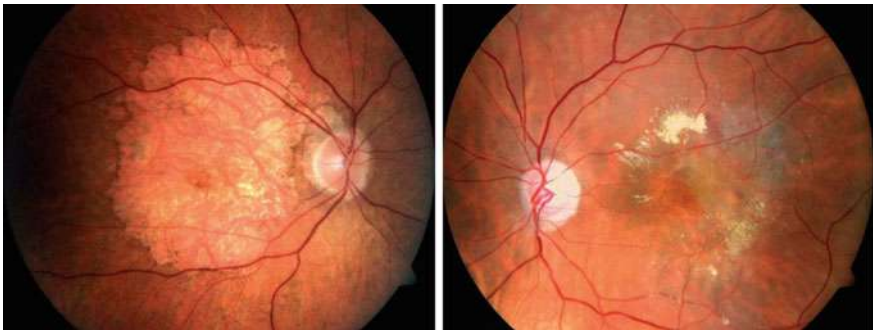


Fig. 11.15 (Left) Geographic atrophy [28]; (right) wet form with edema [49]

Moist form—This is caused by the growth of newly formed vessels from the vasculature that spread below the Bruch membrane. Within the Bruch membrane, cracks are created by which the newly created vessels penetrate under the pigment tissue and later under the retina. The newly created vessels are fragile and often bleed into the sub-retinal space [27–29].

In this case, soft and hard druses are not comparable in colour and shape with the vascular pattern in retinal images; however, they can influence the image preprocessing algorithms, which are preparing the image for extraction of the vascular pattern. Herewith the biometric recognition performance can dropdown. However, this is not a big change. All of the algorithms for retinal image preprocessing should be adopted to treat such diseases to be able to reliably extract the vascular pattern.

The *retinal detachment* (see Fig. 11.16 left) of the eye occurs when a variety of cracks appear in the retina, causing the vitreous fluid to get under the retina and lift it up. Oftentimes, this detachment occurs at the edge of the retina, but from there it slowly moves to the centre of vision when untreated. The ageng process can result in small deposits within the retina, which can create a new connection between the

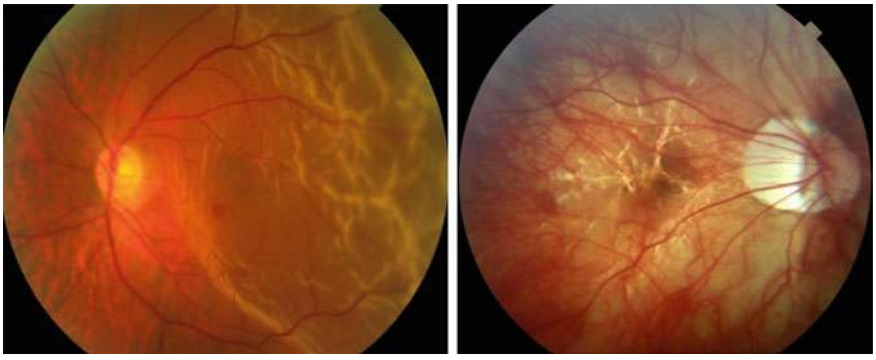


Fig. 11.16 (Left) Retinal detachment [48]; (right) retinal (lacquer) crack [50]

vitreous and the retina [29, 31]. This disease completely destroys the concrete (up the complete) parts of the retina, whereas the vascular pattern is lifted and moved in space, i.e. the original structure before and after this disease is so different that the subject is not be recognised when using a biometric system based on retinal images.

The *retina* can *crack* (see Fig. 11.16 right) in the eye of a person for various reasons. This may be due to the complications of another eye disease, a degenerative form of eye disease, or it can also occur when eye or brain injury occurs. This cracking usually occurs if the retina is not properly perfused for a long time [29, 31]. This means that the venous system beneath the top layer of the retina begins to intermingle, i.e. a new venous structure appears in the retinal image that is difficult to distinguish from the top layer, disabling recognition from the originally stored biometric template. However, it is possible to create a new biometric template in the actual status of the disease that is adapted to the current status after every successful biometric verification.

Retinal inflammation is also known as *retinitis*. Inflammation of the retina of the eye can cause viruses and parasites, but the most common cause is bacteria. In many cases, inflammation of the retina is not isolated and is accompanied by the inflammation of the blood vessel, which holds the retina with blood [29, 31]. Retinitis creates new and distinctive patterns, mostly dark in colour, which greatly complicate the extraction of the venous structure. It is expected to thus have a very strong influence on biometric recognition performance.

Swelling of the *retina*, or diabetic macular edema, affects diabetics as the name suggests. This swelling occurs after leakage of the macula by the fluid. This swelling may occur for data subjects who suffer from long-term diabetes, or if they have too high glucose levels during treatment. Swelling is caused by damage to the retina and its surroundings. These catheters then release the fluid into the retina, where it accumulates, causing swelling [29, 31]. The influence to biometric recognition performance is comparable with the manifestation of retinal detachment—the structure is changed within the space, thus having an impact on the position of vascular system in the retinal layer.

Relatively frequent diseases of the retina are circulatory disorders, where the retinal vessel closes. These closures arise mostly as a result of arteriosclerosis, which is a degenerative vascular disease where it is narrowing and a lower blood supply to tissues [29, 31].

Central vision artery occlusion causes a sudden deterioration in vision. On the ocular background there is a narrowed artery, retinal dyspnea and swelling. Drugs for vascular enlargement, thrombus dissolving medicines and blood clotting drugs are applied [29, 31].

The *closure of the central retinal vein* is manifested by the rapid deterioration of vision; the thrombus causes vein overpressure, vein enlargement is irregular and retinal bleeding occurs. Drugs are used to enlarge the blood vessels and after a time, the thrombi are absorbed, or the circulatory conditions in the retina are improved via laser [29, 31].

Circulatory disorders always have a very significant effect on the colour of the cardiovascular system, making the veins and arteries very difficult to detect, especially

when the vessel is combined with its haemorrhage. In this case, it is not possible to reliably detect and extract the venous system, thereby dramatically reducing biometric recognition performance. Even image preprocessing algorithms will not cope with this problem.

11.2.1 Automatic Detection of Druses and Exudates

The disease occurring in the retina may occasionally prevent the proper evaluation of biometric features. Retinal disease can significantly affect the quality and performance of the recognition. The subject can be warned that the quality of his/her retina is changing and artefacts (warn to go to an ophthalmologist) appear, i.e. they are making recognition difficult. Large areas of the retina image impacted by disease or any disorder will lower the recognition performance, and thus retina image quality counts by rating the concepts of ISO/IEC 29794-1. At the present time, we are focusing on detecting and delimiting the exudates/druses and haemorrhages in the image, automatically detecting the position of the macula and blind spot. These are the reference points by which we determine the location of pathological findings. We associate the centre of gravity of the blind spot with the centre of gravity of the macula (yellow spot). Afterwards, we locate the centre of a given point on this abscissa, which is the reference point for comparing and positioning not only the biometric features in the image, but also the diseases and disorders. The greatest negative consequence of vision is spread to the part called the fovea centralis, where the sharpest vision is located. Once this area is damaged, it has a very significant impact on sight. It is also relevant to detect the quality of blood flow within the retina. There is still a lot to do in all areas of imaging and video processing for medical purposes, as input data is very different.

Due to the lack of images with ARMD in the creation of this work, the images with exudates will be used as well. Druses arising from ARMD are very similar to those exudates that occur in diabetic retinopathy. For this reason, it is possible to detect these findings with the same algorithm. In both cases, there are fatty substances deposited in the retina, which have a high-intensity yellow colour (see Fig. 11.20). Their number, shape, size and position on the retina differ from patient to patient.

The detection of droplets and exudates works with the green channel of the default image (Fig. 11.17 left). A normalised blur with a mask of 7×7 pixels is used. This is due to the exclusion of small, unmarked areas that are sometimes difficult to classify by an experienced ophthalmologist. This Gaussian adaptive threshold is then superimposed on this fuzzy image, which is very effective in defining suspicious areas. The threshold for Gauss's adaptive threshold is calculated individually for each pixel where this calculation is obtained by the weighted sum of the adjacent pixels of a given pixel from which a certain constant is subtracted. In this case, the surrounding area is 5 pixels, and the reading constant is 0, so nothing is deducted. The result of this threshold can be seen in Fig. 11.17 middle. Only now a mask containing the areas of the bloodstream and optical disc that have already been detected earlier

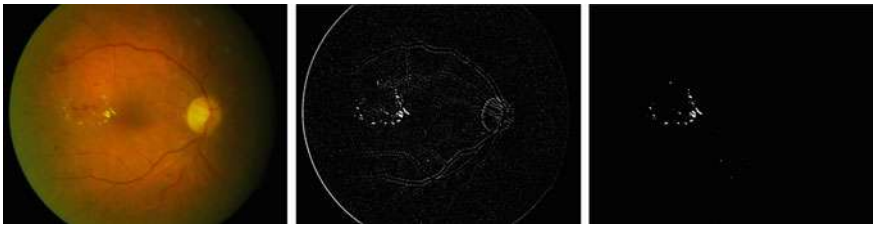


Fig. 11.17 (Left) Original image; (middle) thresholding; (right) obtained suspicious areas

can be applied. If this mask was used at the beginning, it would adversely affect this threshold because it would create too much contrast in the image between the excluded areas and the rest of the retina. This would cause the contours of the blood vessels and the optical disc to be included in suspicious areas, which is undesirable. After the mask is applied, the image is then subjected to a median smoothing with a 5×5 matrix size to remove the noise. The resulting suspicious areas are captured in Fig. 11.17 right.

Retinal images, whose bloodstream contrasts very well with the retina, cause the contours of these vessels to be included in suspicious areas. To prevent this, it is necessary to adjust the bloodstream mask before it is used. Editing is a dilation of this mask in order to enlarge the blood vessels. The difference between the original and the dilated mask is shown in Fig. 11.18 left and right. As soon as this mask is applied, unwanted contours are excluded from the image being processed. A comparison between suspicious areas using an untreated and modified mask can be seen in Fig. 11.19 left and right.

The final step is to determine which of the suspected areas are druses or exudates and which not. For this purpose, the HSV colour model is used, to which the input image is converted. The HSV colour model consists of three components: hue, saturation and value, or the amount of white light in the image.

First, the contours of the suspicious areas are determined in order to calculate their contents. If the content of a given area is greater than 3 pixels, the corresponding

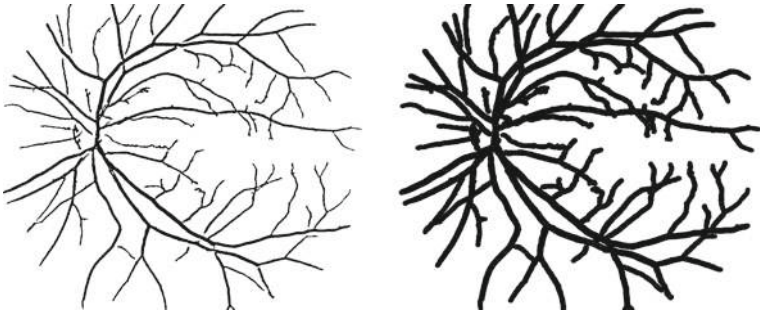


Fig. 11.18 (Left) Original mask; (right) mask after dilatation

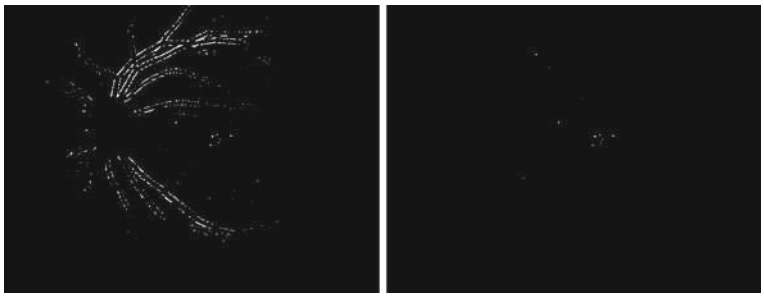


Fig. 11.19 (Left) Suspicious areas with untreated mask; (right) suspicious areas with a modified mask

Table 11.2 Overview of HSVs for classification of suspicious areas

Value	Limit 1	Limit 2	Limit 3
H	30–12	30–15	30–19
S	255–170	255–120	255–187
V	255–120	255–84	255–75

area in the HSV image is located. From this, the average colour tone, saturation and brightness of this area can be calculated. Experimenting on the different images set out the limits set out in Table 11.2. If one of the areas falls within one of these limits, it is a druse or exudate.

Once a region has been classified as a finding, its centre of gravity is calculated using the mathematical moments, which represents the centre from which a circle is created to indicate the finding. Labelling is first performed on a blank image, from which external contours are selected after checking all areas. These are plotted in the resulting image so that individual circles do not overlap the detected findings. The result of the detection can be seen in Fig. 11.20 (see Fig. 11.21).

11.2.2 Testing

The algorithm has been primarily designed to detect findings in Diaret databases, but we also use images from the HRFIDB, DRIVE, and four frames from the bottom of a camera located in the biometric laboratory at the Faculty of Information Technology, Brno University of Technology, to test the robustness. These databases differ in image quality, which greatly affects the accuracy of detection. Table 11.3 shows their basic characteristics. In the initial testing of other databases, the algorithm seemed entirely unusable. After analysng the problem of incorrect detection, the parameters were modified and the algorithm achieved better results.

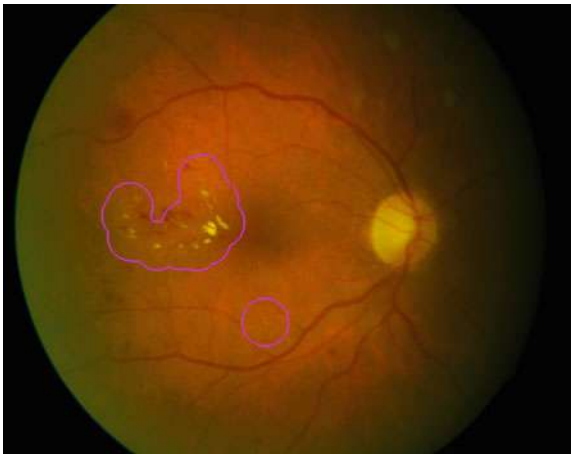


Fig. 11.20 Detection result

Fig. 11.21 Haemorrhage (left), detection of suspected areas (centre) and haemorrhage (right)

Table 11.3 Database characteristics

Database	Number of frames	Format	Size	Camera	FOV
DIARETDB 0	89	PNG	1,500 × 1,152	–	50°
DIARETDB 1	130	PNG	1,500 × 1,152	–	50°
HRFIDB	16	JPG	3,504 × 2,336	Canon CR-1	45°
DRIVE	20	TIF	565 × 584	Canon CR5	45°
BUT retinal database	4	PNG	3,888 × 2,592	Canon CR-1	–

To evaluate the success of detecting the background mask, optical disc and fovea, an ophthalmologist is not required. These parts of the retina may also be determined by a layman after initial training on the basic anatomy of the retina. However, to evaluate the accuracy of detection, it is necessary to compare these results with the actual results, where detection was performed by a manual physician, optimally an ophthalmologist. These findings are relatively difficult to identify and detection requires practice. Evaluating images is also time consuming. Determination of the findings was carried out manually on the basis of a test program in the presence of a student at the Faculty of Medicine at the Masaryk University in Brno. In addition, the DIARETBD0 and DIARETDB1 databases are attached to *diaretdb0_groundtruths* and *diaretdb1_groundtruths*, where there is information about what symptoms are found in the image (red small dots, haemorrhages, hard exudates, soft exudates, neovascularisation).

In order to detect micro-aneurysms, haemorrhages, exudates and druses, a test program has been developed to speed up and automatically evaluate this process. The test program will display two windows to the user. The first window will display an original image with automatically marked holes through which the matrix is placed. On this matrix, you can click through the cursor to pixels (30×30) that we want to mark as finds. In the second window there is an original image from the database—see Fig. 11.22.

The output from the test program provides four types of data: true positive, false positive, true negative, false negative. We obtain these values by comparing ground truth and automatically evaluated areas for each frame. The resulting values are averaged from all images in order to determine overall sensitivity and specificity. Sensitivity for us, in this case, represents the percentage of the actually affected parts of the retina classified by automatic detection as affected. The true positive rate is obtained using the formula:

$$TPR = \frac{TP}{TP + FN}. \tag{2.1}$$

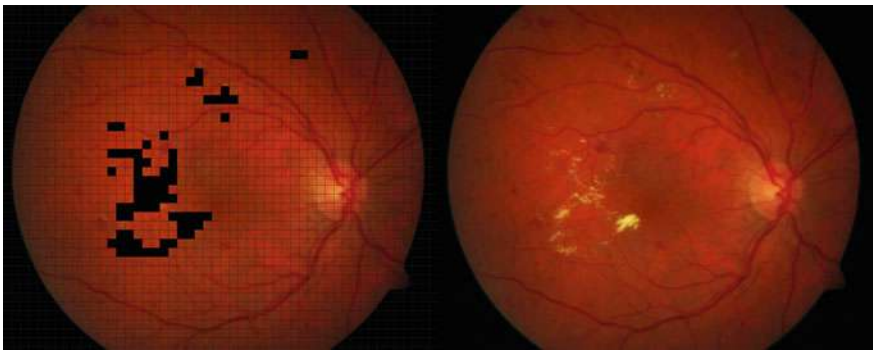


Fig. 11.22 Making ground truths of diseases

Specificity, or true negative rate in our case, means the percentage of healthy parts classified by automatic detection as a healthy retina. We will calculate it according to this relationship:

$$TNR = \frac{TN}{TN + FP}. \tag{2.2}$$

As we can see in Table 11.4, the optical disc was misidentified in eight cases. Incorrect optical disc detection is caused by poor image quality; these shots contain shadows or light reflections from the bottom of the camera. In one case, incorrect detection causes an exudate of the same size and intensity as the optical disc.

The following two tables show the results of individual flaw detection tests (Tables 11.5 and 11.6).

To test the possibility of using the algorithm for other fundus cameras, we use images from the HRFIDB [19] and DRIVE [32] databases, along with four frames from the BUT retinal database. In the first test, the algorithm over these databases showed zero usability. This result causes a different image quality. Table 11.7 shows the success of optical disc detection. The best results were obtained over the HRFIDB database and on the pictures from the BUT database. These pictures are of good quality and do not contain significant disease manifestations.

The following tables show the success of detecting findings: exudates, druses, micro-aneurysms, haemorrhages (Tables 11.8 and 11.9).

There were no signs in the pictures taken from the school camera (Table 11.10).

Table 11.4 Optical disc

Database	True positive	False positive	Success rate [%]
DIARETDB0	85	4	95.29
DIARETDB1	126	4	96.82

Table 11.5 Results of DIARETDB0

Diaretdb0	Sensitivity [%]	Specificity [%]	Success rate [%]
Exudates and druses	94.26	99.41	99.65
Microanalysis and haemorrhage	92.66	99.24	99.24

Table 11.6 Results of DIARETDB1

Diaretdb1	Sensitivity [%]	Specificity [%]	Success rate [%]
Exudates and druses	90.28	99.32	99.65
Microanalysis and haemorrhage	91.46	99.35	99.42

Table 11.7 Results of OD detection

Database	True positive	False positive	Success rate [%]
HRFIDB	16	0	100.00
DRIVE	19	1	94.73
BUT retinal database	4	0	100.00

Table 11.8 Results of HRFIDB

HRFIDB	Sensitivity [%]	Specificity [%]	Success rate [%]
Exudates and druses	69.81	98.76	98.36
Micro-aneurysm and haemorrhage	18.30	99.87	99.51

Table 11.9 Results—DRIVE

Drive	Sensitivity [%]	Specificity [%]	Success rate [%]
Exudates and druses	63.63	99.70	99.70
Micro-aneurysm and haemorrhage	NA	98.63	98.53

Table 11.10 Results—BUT retinal database

BUT retinal database	Sensitivity [%]	Specificity [%]	Success rate [%]
Exudates and druses	NA	99.93	99.93
Micro-aneurysm and haemorrhage	NA	99.97	99.95

11.3 Biometric Information Amounts in the Retina

The third part of this chapter summarises our research in computing the amount of information in retinal images. We analysed the available databases on the Internet and on our own, we computed the amount of bifurcations and crossings there are, and made a first model of the occurrence of these points in the retina. Based on this result we are working on computing a theoretical model for estimating the amount of information (the maximum amount of embedded information in the retina). The grid with occurrence probability distribution is shown in the figures as the end of this section.

In the future, we want to start determining entropy in retina images. Entropy is sometimes also referred to as a system disorder. It is one of the basic concepts in many scientific fields. Information entropy is also called Shannon entropy. In the following lines, the entropy term will always mean information entropy. We will count entropy as a combination of possible variants. For example, fingerprinting methods can be used to calculate retinal biometric entropy. The entropy counting of the biological properties of the eye itself is limited by the sensing device. The resulting entropy is then related to the available resolution. The reason why we want

to estimate the maximum, average and minimal entropy is to get the idea of how precise the recognition could be and how many people we can use this technology for. It is believed that the retinal biometric entropy is corresponding to 10 times more then our population has, however, this has not been proven until today.

Estimations for eye biometric entropy were done by several researchers. Daugman [33] analysed binary iris features, on which the Hamming distance is used for comparing all subjects of a database to each other. He related the score distribution to a Bernoulli Experiment having $N = \frac{\mu(1-\mu)}{\sigma^2}$ degrees of freedom, where μ is the observed Hamming distance mean value and σ^2 is the variance, respectively.

Adler et al. [34] referred to the biometric information as biometric uniqueness measurement. The approaches are based on a brute force estimate of collision, estimating the number of independent bits on binarised feature vectors and the relative entropy between genuine and impostor subspaces.

Nauch et al. [35] analysed the entropy of *i*-vector feature spaces in speaker recognition. They compared the duration-variable p subspaces (Gaussian distribution $p(x) \sim \mathrm{N}(\overrightarrow{\mu_p}, \Sigma_p)$) with the full-duration q spaces (Gaussian distribution $q(x) \sim \mathrm{N}(\overrightarrow{\mu_q}, \Sigma_q)$), simulating the automatic recognition case for the analytic purposes of estimating the biometric information of state-of-the-art speaker recognition in a duration-sensitive manner.

Arakala et al. [15] used an enrollment scheme based on individual vessels around the blind spot. Each vein line is represented by a triple position thickness angle, where the position is the angle in degrees to the centre of the blind spot, the thickness of the vessel is again in degrees and the angle is the slope of the vessel against the thought line passing through the centre of the blind spot. It was found that the position attribute corresponds to a uniform distribution of probability, the distribution of the angles corresponded to a normal distribution with a centre at 90° and a mean deviation of 7.5°. Two peaks appeared in thickness, so the description of the probability distribution was divided into peak and normal distributions. The study resulted in an approximate entropy value of 17 bits.

11.3.1 Theoretical Determination of Biometric Information in Retina

Based on the previously mentioned work [15], we try to count biometric entropy in a wider area around the blind spot. First, we mark the ring area with a radius of distance between the blind spot and fovea and cut off the blind spot. Then we mark crossings and bifurcations. The resulting region we unfold from polar coordinates to Cartesian ones. The resulting rectangle is then used for easier indexing of the place.

Using this principle, we expect deployment at any point of area. Then, using the combinatorial Eq. (3.1), we calculate the maximum (theoretical) number of feature points. We simulate all combinations of points in area. In this equation, we are particularly interested in the position of the points, then the angle at which the

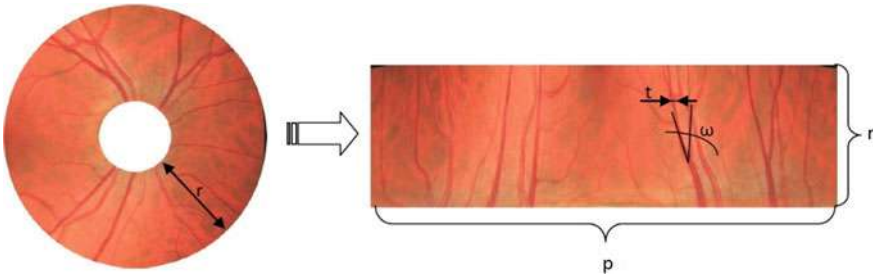


Fig. 11.23 Unfolding interest area

individual vessels are at the centre of the blind spot, and finally their thickness.

$$\Psi = \binom{p \cdot r}{n} \cdot \binom{\omega + 2}{3} \cdot \binom{t + 1}{2}, \tag{3.1}$$

where r is the width of the ring in pixels, p is the width in pixels of the expanded ring around the blind spot, n is the average number of features (crossings and bifurcations) in the image, ω is the number of possible angles that the vessels enclose with each other and t (in the Fig. 11.23) is the maximum vessel thickness. The first part of the formula expresses the possible location of features. It is a combination without repetition—two features cannot occur in the same place. The angles ω usually have a value of about 120°, as their sum will always be 360°. Angles can be repeated, so a repeat combination is used in the formula. Likewise for the last part. The vessel thicknesses of two out of three will be used for their resolution. The third thickness is usually the same as one of the two previous ones.

When adding derived parameters from several retina samples, we can approximately calculate how many combinations of all parameters are within their limits.

$$\Psi = \binom{p \cdot r}{n} \cdot \binom{\omega + 2}{3} \cdot \binom{t + 1}{2} = \binom{360 \cdot 120}{20} \cdot \binom{60 + 2}{3} \cdot \binom{12 + 1}{2} = 6.2 \times 10^{80}. \tag{3.2}$$

11.3.2 Used Databases and Applications

For the purpose described at the beginning of this section, we used three publicly available databases: Messidor [18], e-ophtha [36] and High-Resolution Fundus (HRF) [19]. The *Messidor* database contains 1,200 eye fundus colour numerical images of the posterior pole. Images were acquired by three ophthalmologic departments using a colour video 3CCD camera on a Topcon TRC NW6 non-mydriatic retinograph with a 45° field of view. The images were captured using 8 bits per colour plane, at 440 × 960, 240 × 488 or 304 × 536 pixels. 800 images were captured with

pupil dilation (one drop of Tropicamide at 0.5%) and 400 without dilation. The *e-ophtha* database contains 47 images with exudates and 35 images with no lesions. The *HRF* database contains 15 images of healthy patients, 15 images of patients with diabetic retinopathy and 15 images of glaucomatous patients. Binary gold standard vessel segmentation images are available for each image. Additionally, the masks determining Field of View (FOV) are provided for particular datasets. The gold standard data is generated by a group of experts working in the field of retinal image analysis and medical staff from the cooperating ophthalmology clinics.

We randomly selected 460 images from Messidor, 160 images from e-ophtha and 50 images from HRF. In the selected retinal images, both left and right eye images were available. Images were reduced to a resolution of about 1 Mpx in order to fit images on screen.

We developed three application software modules (marked as SW_1, SW_2 and SW_3). SW_1 was developed for manually marking blind spots, yellow spots and features as well as determining their polar coordinates. We marked all retinal images via SW_1 one by one. At first, we marked the boundary of the blind spot and then the centre of the yellow spot. SW_1 considered the blind spot as the pole and the line between the blind spot to the yellow spot as the polar axis. Therefore, the angle between the two spots was 0°. SW_1 considered the distance between two spots as the unit distance. Usually, the distance in pixels was not equal for two different retinal images. However, SW_1 considered distance as one unit for each image. Therefore, the position of the yellow spot in every image was (1, 0°) in polar coordinates. After marking two spots, we marked each feature by a single click. SW_1 estimated the polar coordinates of each feature by increasing clockwise and scaling distance.

SW_2 was developed to conduct the marking process automatically and to compare its detection accuracy with the manually marked-up results. The details of this software were summarised in one master thesis [37].

SW_3 was developed to estimate the number of features in different regions as shown in Fig. 11.23. SW_3 loaded all marked retinal images one by one and mapped the polar coordinates of features to Cartesian coordinates. After that, SW_3 presented the intensity of occurring features in the area of 5×5 pixels by a range of varying shades of grey. The darker shade represented the higher occurrence of features, whereas the lighter shade represented a lower occurrence. Then SW_3 drew two circles in order to show the boundary of the location of features, where the inner circle covered a 90% area of the outer circle. Two circles were split up into four sectors by a horizontal line and a vertical line. Radiuses were drawn every 18°, which split each sector into five regions. The percentage of the occurrence of features in each region was written outside of the outer circle. SW_3 also drew two ellipses, E_{blind} and E_{yellow}, in order to show the region surrounding the blind spot and the yellow spot, respectively. The sizes of the ellipses were dependent on a threshold value δ_1. That means the size of a single ellipse was increased until the number of features inside that ellipse did not cross the δ_1 value. SW_3 also drew an arc along the x-axis. The width of the arc was decided by a threshold value of δ_2. We set δ_1 to 10 and δ_2 to 500, based on the number of labelled points in all retinae.

11.3.3 Results

On average, we found 48 features on each image. The success rates of locating blind spots and the yellow spot automatically were 92.97% and 94.05%, respectively. The wrong localisation of spots was caused primarily because of spots that were too bright or too dark. The average deviation of a feature marked by SW_1 and SW_2 was about 5 pixels [37]. E_{blind} occupied 2.040% of the retina area, whereas E_{yellow} occupied 2.728% of the retina area, as shown in Fig. 11.24. The number of features is very low inside E_{blind} and E_{yellow}, especially inside E_{yellow}. Therefore, E_{yellow} was bigger than E_{blind}. On the real retinal image, near the yellow spot, the branches were so small and the blood vessels were so thin that they were not captured by the fundus camera. Therefore, a wide empty space can be seen near E_{yellow} in Fig. 11.24. We also noticed that the major blood vessels often directed to four main directions from the blind spot.

By creating a bifurcation and crossings scheme, we can now start generating formulas for calculating the biometric entropy of retinal images using our biometric recognition method. In the Fig. 11.24, there are areas around the blind spot and the fovea where almost no markers are present. The area between the maximum edge (grey in the picture) of the points and the (green) inner circle is eliminated from the calculation. It's a part that did not have to be seen in most of the pictures.

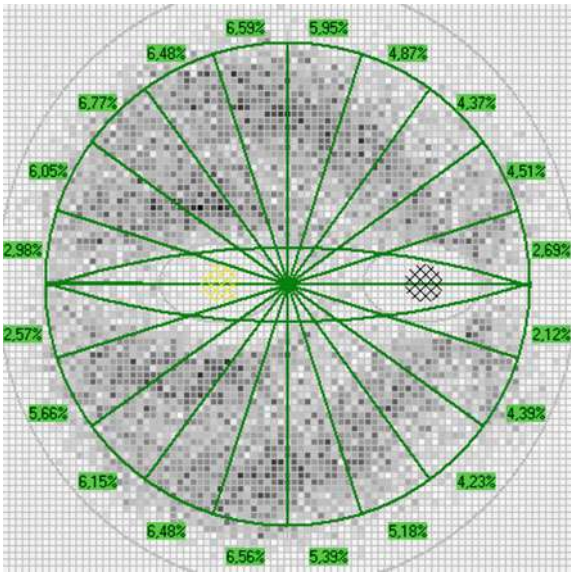


Fig. 11.24 Merged all bifurcations and crossings from the marked images

11.4 Synthetic Retinal Images

The last section of this chapter will be devoted to our generator of synthetic retinal images. We are able to generate a synthetic retinal image, including the blind spot, macula and vascular patterns with randomly generated or predefined features (crossings and bifurcations). Now we are working on the additional features that will decrease the quality of such images, e.g. reflections, diseases. We are also working on supplementing that with something that will generate diseases and damage on the image of retina, so we can create a unique database for deep learning.

The main reason for a such generator is that it is very difficult to get a large-scale database(s) with thousands of retinal images. To collect retinal images from subjects, you need the appropriate equipment (minimally digital ophthalmoscope or even better a fundus camera) and you need to find the volunteers who will be willing to let their retinas get acquired. The best way, comparably with fingerprint areas in biometric systems (synthetic image generators SFinGe, Anguli and SyFDaS), is to use a generator of synthetic images. With that it is possible to generate any large-scale database, where you can predefine (in a configuration file) the setting, i.e. how many images with which background, distortions and features should be generated. Therefore, this part is very important for biometric systems, because with this way the training and testing of algorithms for biometric retinal recognition could be done on large-scale databases. It is important that the quality of the images correspond to the real images, i.e. some work is still ahead of us.

First, a basic idea of how the generator will work and how its main parts are identified is described. Furthermore, the designs of the individual parts of the generator are described in greater detail and are intended to create partial sections of the resulting image. The aim is to design the generator so that it generates images as close as possible to real images of the retina. Real images often have a very different look in terms of colour distribution or detail. One of the test options which we compare the reality of created images is using the bifurcation and crossing searching described in Sect. 11.1.4.

The generator is able to create the desired number of randomly generated synthetic retinal images at the selected resolution and the selected general properties, such as the image angle or the zoom rate according to the specified parameters.

The generator can then generate a large number of images of the retina, where it is possible to train and test various algorithms. If we add a disease creation module to the generator, we can also test algorithms for further detection.

11.4.1 Vascular Bed Layer

The retinal vasculature of the retina consists of the arterial and venous channels. Both of these beds can be divided into upper and lower branches, which are further divided into nasal and temporal branches.

When generating the texture of this layer, the generator uses pre-generated branching positions for the arterial and vein branches. The method for generating these positions is described in Sect. 11.4.4. Generally, the generator first creates separate textures of the arterial and venous channels, which then merge into one final texture (see Fig. 11.25). This division is necessary due to the way the vascular bed is rendered. It counts that blood vessels do not cross each other.

Partial textures are merged so that when the artery and vein are in the same position, a new value of the colour and transparency of the texture is calculated at that position. In this calculation, both original colours are used with respect to transparency, with the unified textured vein being drawn above the artery. If only the artery or vein is at the given position, it will be redrawn into the resulting texture unchanged. If there is no vessel in the position, this position remains transparent on the resulting texture.

Partial textures then arise through the gradual plotting of the individual branches of the arterial or venous passages.

In order for a natural resulting vessel shape, it is necessary that the connectors between the individual branches of the branch take the form of a curve without significant sharp breaks at the branching points. Because the curve is the link between the sequences of points, it cannot be divided into several parts at one point. Therefore, the branched tree of the given branch is plotted sequentially, as shown in Fig. 11.26. A description of this plotting is given in Chap. 4.4.

Gradual rendering takes place by gradually forming a curve from the initial point of the branch of the vascular stream, which passes through the following branches of branching, where it continues with a wider vessel at any one of the endpoints of the vascular bed. As soon as the vessel is drawn from the beginning to the end, a

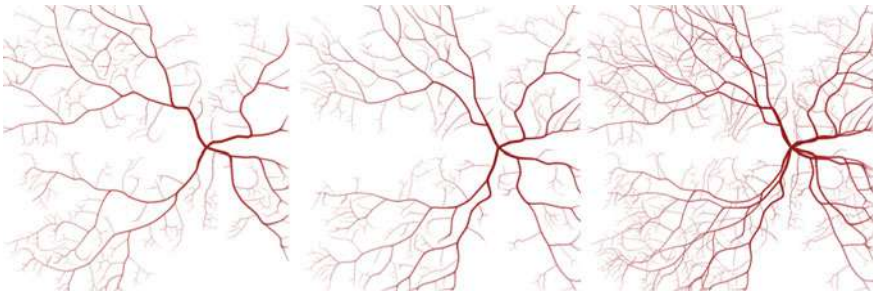


Fig. 11.25 (Left) Arterial fluid texture; (middle) vein texture; (right) resulting vascular fluid texture

Fig. 11.26 Gradual rendering of the upper temporal branch

Fig. 11.27 Connecting the new vessel to the already depicted vessel at the branch point

new starting point is chosen as one of the already drawn branch points, in which the beginning has the widest still unrefined vessel. The vessel with this starting point will be drawn in the same way as the first vessel. This procedure is repeated until all the blood vessels of the branch are drawn. To plot the vessel the cubic Bézier curve is used: see [38].

The vessel is plotted sequentially from the starting point to the endpoint following the pair of branching points running consecutively. For each point's pair and the relevant control points that affect the shape of the curve between them, the partial points of the curve are then calculated.

Calculated Bézier curve points are then linked by lines whose points are calculated using the Bresenham algorithm. A texture of the blood vessel is drawn around this curve, consisting of partial segments. For each point of the curve, a semicircle is drawn in the direction of the line below which the point belongs. The Bresenham algorithm is also used to draw this semicircle, with the radius of the circle (line length) equal to half the width of the vessel at that point. In this rendering process, all points belonging to the texture of the vessel are rendered, but for one point its colour is calculated several times with different parameters. The resulting colour is selected as the colour whose individual components have the highest value. The lightest and least transparent colour corresponds to the smallest distance from the centre of the vessel.

This method of selecting the resulting point colour is the reason why arteries and veins have to be plotted separately and then combined into one texture in another way. However, it is used when plotting a new vessel to connect this vessel to the already drawn vessel at the branch point: see Fig. 11.27.

The basic RGB colour of the texture is in the artery (160, 15, 15) and in the vein (150, 5, 15). The individual colour components are adjusted for each frame by multiplying by *rand* (0.99, 1.01).

11.4.2 Layers

When looking at the real images of retinas, it is possible to easily identify four different parts of the image that can be generated separately and then be combined into a final image. These subparts are represented as image layers in the generator, where the lowermost layer contains the background texture of the retina. Here, the layer containing the texture of the optic nerve target overlaps. Both of these layers

are covered by another layer containing the texture of the vascular bed. All layers then overlay the textured frame layer. Figure 11.28 shows the plot of the individual layers in the given order.

The layer has the shape of a square surface on which the texture is applied. The side size of this area is equal to the shorter side of the rendering window, which is multiplied by two scaling parameters. The centre of the layer is aligned to the centre of the rendering window, with only the parts of the generated text within the rendering window being included in the resulting image.

Because of the layer size and texture variable applied to it, the generator uses a custom coordinate system to create textures, where it then maps the individual pixels of the texture.

Scaling, shifting and rotating the layer and the texture are designed to be independent of texture generation. While scaling modifies the layer size and does not manipulate the coordinate system, rotation and displacement do not change the position of the layer but are applied to the coordinate system.

As can be seen in the real frames shown in the earlier sections of this work, the images of the retina do not always occupy the whole area of the image, or sometimes they are partially cut-off. Therefore, we resize the layer so that the size of the rendering window does not change, as well as the resolution of the resulting image.

As with the first case, but this time without changing the frame texture layer size, it is possible to choose how much of the retina is presented in the image, so be sure to choose the pixel size of the fundus camera that would capture such a frame. Different settings for this parameter are shown in Fig. 11.29.

Real motion capture is not always ideal. The image is more or less rotated and possibly slightly shifted. The displacement may also be deliberate if another part of the retina is being captured. For this reason, these transformations also allow the proposed generator. Both transformations are applied to the coordinate system, not to the layer itself. First, a shift is made followed by rotation. For each layer, it is possible to set the own rotation and displacement size with both layers transforming over layers. Thus, when the background is rotated and shifted, the target of the optic nerve and the vascular bed is shifted. Further transformation at the optic nerve target layer can then change its position relative to the background. Likewise, the position of the vascular bed can be changed to the lower two layers. Since these transformations

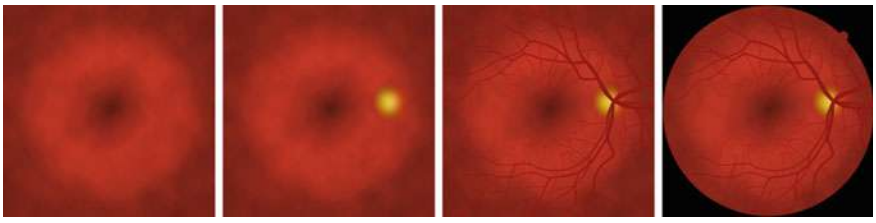


Fig. 11.28 A gradual render of layers. (left) Background layer; (left middle) adding a layer of the optic nerve target; (right middle) adding a vascular bed layer; (right) adding a layer of frame

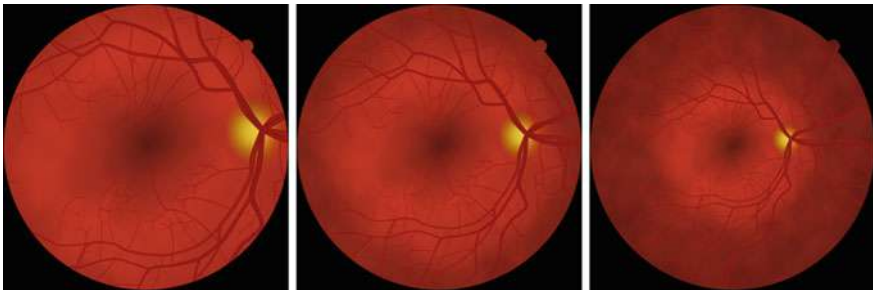


Fig. 11.29 The different sizes of the retrieved part of the retina: (left) maximal zoom; (middle) central zoom; (right) no zoom

are intended to simulate a different eye position when capturing the retina, they are not applied to the frame layer.

11.4.3 Background Layers

The retina background is mostly reddish; the fovea and ex-macular periphery are darker. The area between the fovea and the border of the macular area is then lighter. In a more detailed view, smaller objects of different colours and intensities are visible throughout the area, creating a dense vascular network of the cavity.

The generated background texture is opaque to basic RGB colour (200, 60, 40). Figure 11.30 shows the resulting background texture.

This function describes the randomness of the background texture and is generated by the shadowing choroid. It uses Perlin noise, which has three octaves, frequency and amplitude set to 1, and returning values from interval <−1;1>. Perlin noise is also initialised by a random number, making it different for each frame.

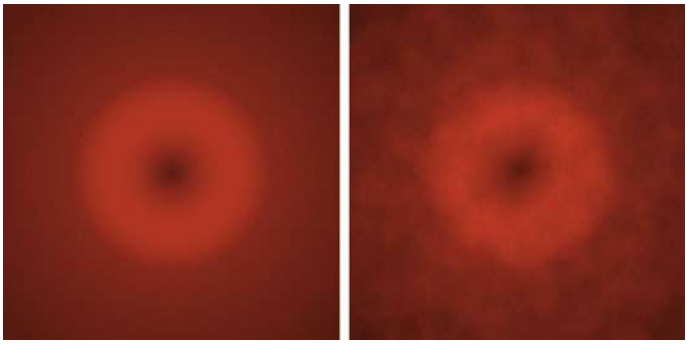


Fig. 11.30 (Left) The resulting background texture without a noise function; (right) with a noise function

Graphically, the function is depicted in Fig. 11.31 where the dark areas indicate the positive values of the noise and the light areas of the negative values. When the dark area is getting lighter, the closer value of the function is to 1 and when the light area is getting lighter, the function's value is closer to −1. At the transition of dark and light areas, the function has a value of 0.

The texture of the Optic Disc (OD) target is largely transparent except the ellipse-shaped area that contains the texture of the OD target itself. When generating a texture inside this ellipse, the base colour of the RGB value is again returned (250, 250, 150). Each folder is multiplied by the function *rand* (0.98, 1.02), as well as background textures to ensure the variability of the base colour for different images.

Figure 11.32 shows the resultant texture of the OD target (cut from the overall layer texture) together with the individual colour components from which it was composed. However, the colour of the texture still changes in the final rendering, and because of its partial transparency, its colour also affects the colour of the background texture beneath it.

For each image, the final position of the OD is slightly different due to accidental slight rotation and displacement. When the left-eye image is generated, the rotation is 180°.

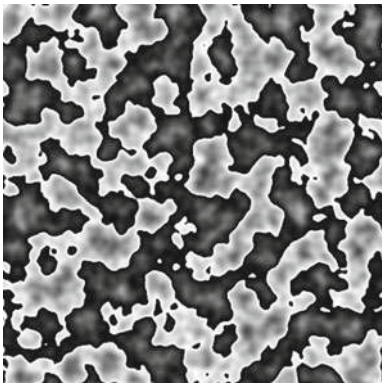


Fig. 11.31 Noise function

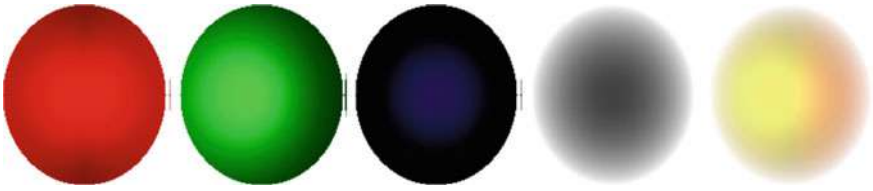


Fig. 11.32 The texture of the optic nerve target and its parts: (left) red texture colour component; (left middle) green; (middle) blue; (right middle) texture transparency; (right) resulting texture

11.4.4 Generating a Vascular Bed

Before drawing a vascular bed, it is first necessary to generate the branch positions of the blood vessels and properties of these points needed for plotting. These points are generated separately for each of the major branches of the artery and vein. Branching points are generated for all branches by the same algorithm with different values of some parameters. Their generation is divided into two parts. First, a tree of branch points is generated, and then the positions of individual points are gradually calculated with respect to the already calculated positions of the other points in the tree.

Each branch point has several properties that need to be generated:

- **Point position** (counted later),
- **Distance from previous point**—length of line between these two points,
- **Vessel width**—value from interval <0;1> , where 1 has a vessel at the starting point of a given branch, and a value of 0 has the endpoints of a given branch.
- Point type
  - Y-branching—the vessel is divided into two approximately equally wide vessels,
  - T-branching—the vessel is divided into a wide and narrow vessel,
  - no branching—the vessel is not split, just passing through the point,
  - end of vessel.
- **Types of vessel** (see Fig. 11.33)

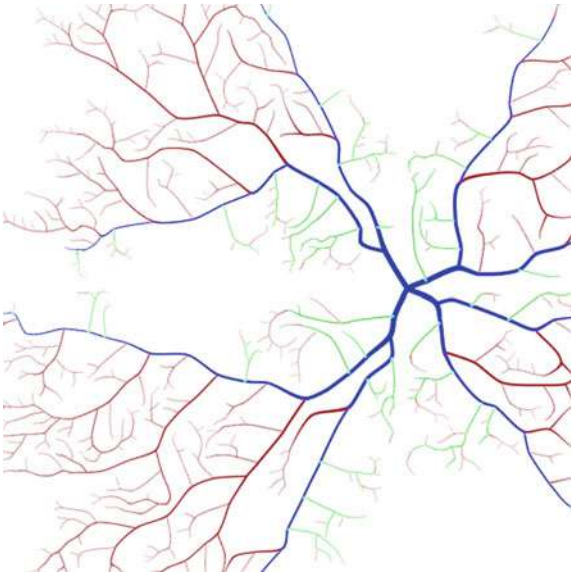


Fig. 11.33 Colour illustration of different types of vessels

– left and right strong blood vessels (blue),
– left/right wider weak blood vessel emerging from the left/right strong blood vessel (green),
– other blood vessels (red).

The root of the branch tree is the point located at the centre of the optic nerve target. It creates one of the following branching points, and then generates the tree recursively so that each new branching point generates the following two branch points. Generation ends when the vessel's width at the newly created point is ≤ 0. The properties of the following branch points are also calculated, and the design of the method of calculating some of them was based on the information published in [39].

The distance from the previous point d is calculated for the following two points according to the vessel width w_a at the current point as follows:

$$d = \begin{cases} rand\,(0.15,\ 0.05) \text{ for } w_a > 0.15 \\ rand\,(0.05,\ 0.02) \quad \text{else} \end{cases}. \tag{4.1}$$

This has the consequence of the narrow blood vessels having more branches.

First, depending on the type of branch of the current point, the ratio is calculated to which the right and left successor are divided. If it is the current point Y-branch, the ratio of right and left successors is calculated as r: $(1 - r)$, where $r = rand$ (0.45, 0.55). In case of T-branching, it is 50% probability $r = rand$ (0.95, 0.99), otherwise $r = rand$ (0.01, 0.05).

If the current point is a part of the leftmost or rightmost strong blood vessel, this probability is altered in the T-branch, such that the weaker T-branch branches are generated towards the boundary of the quadrant. In the beginning, there is a 70% probability that the weaker vessel is generated towards the boundary of the quadrant. If this happens, this probability will decrease by 10% for the type of vessel (left or right); if not, the probability will increase by 10%.

The value of the vessel's width is then calculated for both of the following branch points using their distance from the actual point, the vessel width at the current point, and the division ratio as follows:

$$w_r = \left(w_a \times \sqrt{r}\right) - \left(w_a \times \frac{d_r}{10}\right) - \frac{d_r}{20}, \tag{4.2}$$

$$w_l = \left(w_a \times \sqrt{1 - r}\right) - \left(w_a \times \frac{d_l}{10}\right) - \frac{d_l}{20}. \tag{4.3}$$

If the width of the calculated vessel at the next point is not positive, this point is marked as the vessel endpoint. If the calculated width is negative, the distance of that point from the previous point is adjusted to the width of the vessel, which at that point is equal to zero.

In other cases, it is decided whether the following point will be a Y-branch or a T-branch. One of the auxiliary features of a point is the probability of selecting the

Y-Branch for its following branching points, that is, at the starting point, set to 20%. If the selected branch type of the next branch is the Y-branch, then this probability is set to 0% at this next point. If the T-branch is selected and the next point is the weaker T-branch of the current point, the probability for this next point is set to 40%. Otherwise, the probability is increased by 25%.

First, the position of the leftmost and rightmost points of thick blood vessels (type 1) is calculated, then it points the position of the left/right wider weak blood vessels resulting from vascular type 1 (type 2) and finally, the position of the other vessel (type 3). Within these types of vessels, the order of points in the calculation of the positions is given by the width of the vessel at a given point, with the positions of the wider vessels being counted first. Point positions are counted in this order because not all tree branch points generated will eventually be used.

When calculating the position of a particular branch point, the set of positions on which this point may be located is first determined. From the beginning, these are the positions around the previous branch point at the distance that this particular point has generated as a property. Then, depending on the direction of the vessel at the previous point, this set is limited by the interval of angles in which the position of the point may be. For each of the remaining positions, the weight of the position is calculated based on the deviation from the centre of the interval.

On the real images, the observed part of the retina is circular and the rest of square image is black. A majority of the right-hand portion of the image tends to see a smaller part of the retina in the shape of a semicircle or rectangle. This is to know where the picture is; for example, if it is not turned.

The generator allows you to choose which quadrant the mark will be in, and also whether the mark will have the shape of a semicircle or rectangle. The generated texture has a black colour and, depending on the coordinates, only the transparency of the texture changes.

11.4.5 Testing

We are now comparing the created synthetic retinal images with our ground truth. We use manually marked, real retinal images to create a density map, where there are the most bifurcation and crossing points. Using the same procedure, we want to automatically create a density map for synthetic retinal images and compare both results.

We developed the applications SW_1 and SW_2 (see Sect. 11.3.2). SW_1 was developed for manually marking blind spots, yellow spots and features, as well as determining their polar coordinates. We marked all retinal images via SW_1 one by one.

SW_2 was developed to estimate the number of feature points in different regions. SW_2 loaded all marked retinal images one by one and mapped polar coordinates of feature points to Cartesian coordinates. After that, SW_2 presented the intensity of occurring features in 5×5 pixels by a range of shades of grey. The darker shade

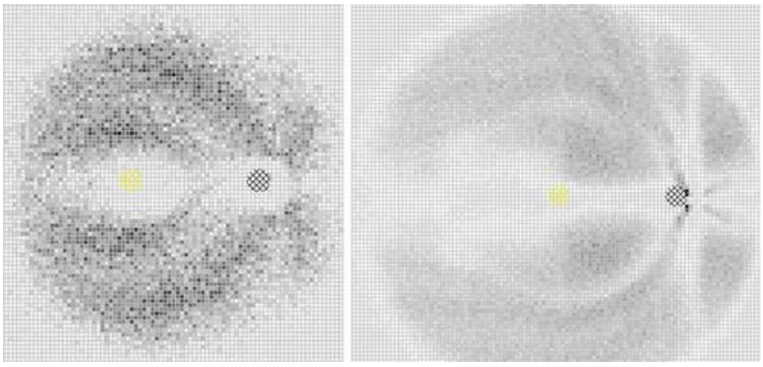


Fig. 11.34 (Left) Density occurrence of summarised real retinas; (right) density in synthetically generated retinas

represented higher occurrence of features, whereas the lighter shade represented lower occurrence.

Using the application described in the previous chapters, 1,000 images were generated in which the crossover and bifurcation were found. The occurrence frequencies were merged with the SW_2 described in Sect. 11.2.2 and graphically represented result seen in Fig. 11.34 left.

It was then possible to visually compare the results of synthetic and real retinal images. In Fig. 11.34 right, there are visible features on the blind spot. It's a side effect. On real retinas, there were no marked features inside the blind spot.

Figure 11.34 shows the summarised occurrences of crosses and bifurcations for real (left) and synthetic (right) retinal images. Picture (left) is marked manually and picture (right) is marked automatically. Both pictures are made up of about a thousand retinas. The shades' range of the right picture is expanded because automated search for markers included features inside the blind spot. Features inside the blind spot in the left image were removed during manual labelling. Although the application generates blood vessels in the synthetic retina symmetrically, some similarities with the summation from the real retina can be traced.

The application is composed only of basic algorithms. As a result, there could be regular shapes seen in Fig. 11.34 right. We assume that, based on real retinas research, we can better specify the distribution of crossings and bifurcations in the model.

11.4.6 Generating Synthetic Images Via Neural Network

In another application, we first generate healthy images, where we can train algorithms for detection and extraction of the optical disc and fovea. Furthermore, we generate diseased retinal images with manifestations of ARMD and diabetes, e.g.

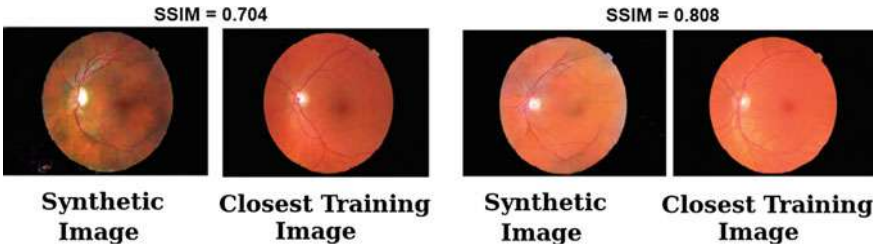


Fig. 11.35 Comparison of synthetic image and the closest training image from the database

haemorrhages, exudates. The neural network learns such images, which we have in the training set. At the moment, we only have images only for ARMD and diabetes; however, new images are stored in the database, i.e. it is possible to add new features representing new ophthalmologic diseases.

In biometric systems, it is often the case that a damaged image does not pass through the recognition. However, there is often not enough training data for detecting algorithms. Therefore, it is advisable to create large databases of synthetic meshes damaged by disease.

We have trained Generative Adversarial Networks (GANs) [40] to generate synthesised retinal images. A GANs-based retinal image synthesiser consists of two neural networks: a Generator (G) and a Discriminator (D). We have not used any extra information (such as blood vessel trees) to generate retinal images using GANs. However, we have emphasised maintaining a balance between the two competitors (i.e. G) and (D) during training. We have found that if this balance is not kept, G may end up generating only blurry retina images without high-level structures, such as blood vessel trees, optic discs, macula, etc.

Algorithm of GANs-based Retinal Synthesiser is as follows:

- For k times
  - Prepare a mini-batch of retinal images $\left\{\left(x, \hat{x}\right)_{i=1}^{m}\right\}^{kn}$ where m is the mini-batch size.
  - Update D using $\left\{\left(x, \hat{x}\right)_{i=1}^{m}\right\}^{kn}$.
- For r times
  - Prepare a mini-batch of noise vectors, $\left\{(z)_{i=1}^{m}\right\}^{rn}$.
  - Update G using $\left\{(z)_{i=1}^{m}\right\}^{rn}$.

We have used 1,200 images from the public database *Messidor* [18]. These images were acquired by three ophthalmologic departments using a colour video 3CCD camera on a Topcon TRC NW6 non-mydriatic retinograph with a 45-degree field of view. The images were captured using 8 bits per colour plane. Among these 1,200 images, 588 images were 960×1440, 400 images were 1488×2240 and 212 images were 1536×2304. In our experiments, we resized all of the images to the same size (i.e. 256×256) by bicubic interpolation.

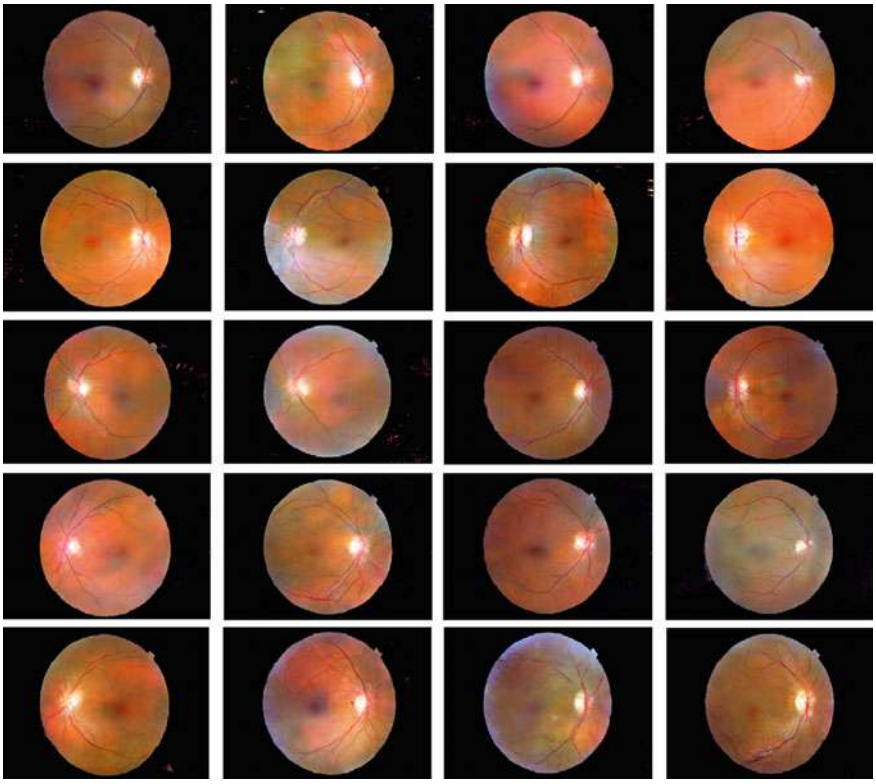


Fig. 11.36 Examples of generated synthetic retinal images

We have followed the deep convolutional neural network-based architecture suggested in [41] with minor modifications. Table 2 shows the model architecture for 256 $\times$ 256-sized images. The mini-batch size was set to 32 (i.e. m = 32). Noise vectors were drawn from the uniform distribution. As a loss function, binary cross-entropy was used. As an optimiser, RMSProp with a learning rate of 0:0001 and a decay of $3e^{-8}$ was used. The dropout value was set to 0:5. For batch normalisation, momentum was set to 0:5 instead of default value 0:99. For LeakyReLU, it was set to 0.2 instead of the default value of 0:3. For all convolutional and transposed convolutional layers, stride = 2, kernel size = 5 and padding = same was used. l2 regularisation was applied only for weights and biases of the transposed convolutional layers. For all other settings, the default values of Tensor Flow's Keras API were used.

After training, the generator is used to generate synthesised retinal images from noise vectors. The Structural SIMilarity (SSIM) measure shows how similar the synthesised images are to the training data. SSM = 0 means there is no similarity and SSIM = 1 means that two images are the same. You can see some achieved results from this GAN generator of synthetic retinal images in Figs. 11.35 and 11.36.

A sample database of generated images is available at https://strade.fit.vutbr.cz/databases/synthetic_retina.

The database is separated into two parts: healthy images and disease-affected images, which is especially diabetes and ARMD.

Acknowledgements This work was supported by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II); project "*IT4Innovations excellence in science—LQ1602*" and the BUT project "*Secure and Reliable Computer Systems*", FIT-S-17-4014.

References

1. Barajas H (2015) Atlas of the human eye: anatomy and biometrics. Palibrio, p 74. ISBN 978-1506510330
2. Snell RS, Lemp MA (2013) Clinical anatomy of the eye, 2nd edn. Wiley-Blackwell. ISBN 978-0632043446
3. Bergmüller T, Christopoulos E, Schnöll M, Uhl A (2015) Recompression effects in iris segmentation. In: Proceedings of the 8th IAPR/IEEE international conference on biometrics (ICB'15). Phuket, Thailand, pp 1–8
4. Nguyen K et al (2018) Iris recognition with off-the-shelf CNN features: a deep learning perspective. IEEE Access 6:18848–18855
5. Trokielewicz M, Czajka A, Maciejewicz P (2019) Iris recognition after death. IEEE Trans Inf Forensics Secur 14(6):1501–1514
6. Goldstein I, Simon C (1935) A new scientific method of identification. N Y State J Med 35
7. Tower P (1955) The fundus oculi in monozygotic twins: report of six pairs of identical twins. AMA Arch Ophthalmol 54:225–239
8. Farzin H, Abrishami-Moghaddam H, Moin MS (2008) A novel retinal identification system. EURASIP J Adv Signal Process Hindawi 2008:10. https://doi.org/10.1155/2008/280635
9. Hill RB (1996) Retina identification. In: Biometrics: personal identification in networked society. Springer, New York, pp 123–141
10. Timberlake GT, Kennedy M (2005) The direct ophthalmoscope—how it works and how to use it. University of Kansas, p 39. Available online on http://web.media.mit.edu/~raskar/Eye/TheDirectOphthalmoscope.pdf
11. Hill RB (1986) U.S. Patent 4,620,318. http://www.freepatentsonline.com/4620318.pdf
12. Arndt JH (1990) U.S. Patent 4,923,297. http://www.freepatentsonline.com/4923297.pdf
13. Johnson JC, Hill RB U.S. Patent 5,532,771. http://www.freepatentsonline.com/5532771.pdf
14. Fuhrmann T, Hämmerle-Uhl J, Uhl A (2009) Usefulness of retina codes in biometrics. In: Advances in image and video technology, 3rd Pacific Rim symposium, Japan, Springer LNCS 5414. ISSN 0302-9743
15. Arakala A, Culpepper JS, Jeffers J, Turpin A, Boztas S, Horadami KJ, McKendrick AM Entropy of the retina template. RMIT University, Melbourne, Australia
16. Shih HP 2008 Two algorithms for maximum and minimum weighted bipartite matching. Department of Computer Science and Information Engineering National Taiwan University
17. Carmona E, Rincón M, García-Feijoo J, Martínez-de-la Casa J (2008) Identification of the optic nerve head with genetic algorithms. Artif Intell Med 43:243–259
18. Decencière E, Zhang X, Cazuguel G, Lay B, Cochener B, Trone C, Gain P, Ordonez R, Massin P, Erginay A, Charton B, Klein JC (2014) Feedback on a publicly distributed database: the messidor database. Image Anal Stereol 33(3):231–234
19. Kohler T, Budai A, Kraus M, Odstrčilík J, Michelson G, Hornegger J (2013) Automatic no-reference quality assessment for retinal fundus images using vessel segmentation. In: 26th IEEE international symposium on computer-based medical systems. pp 95–100

20. Ortega M, Penedo MG, Rouco J, Barreira N, Carreira MJ (2009) Retinal verification using a feature points based biometric pattern. EURASIP J Adv Signal Proc 2009. Article ID 235746:13 pp
21. https://github.com/Lukass2/RetinaFeatureVectorExtractor.git
22. Albert DM, Miller JW et al (2008) Principles and practice of ophthalmology, 3rd edn. ISBN 978-1-4160-0016-7
23. Ryan SJ (2006) Retina. Elsevier Mosby. ISBN 0323043232
24. Diagnosis and classification of diabetes mellitus. Diabetes Care, American Diabetes Association, Issue 33, 2010, pp 62–69. https://doi.org/10.2337/dc10-s062
25. Scanlon PH, Wilkinson CP, Aldington SJ, Matthews DR (2009) A practical manual of diabetic retinopathy management. Wiley-Blackwell. ISBN 978-1-405-17035-2
26. Sosna T (2016) Diabetická retinopatie (Diabetic retinopathy), 2nd edn. Prague, Axonite CZ. ISBN 9788088046059
27. Cavallotti CAP, Cerulli L (2008) Age-related changes of the human eye. Humana Press, p 400
28. Kolář P (2008) Věkem podmíněná makulární degenerace (Aged-related macular degeneration), 1st edn. Prague, Grada. ISBN 9788024726052
29. Kanski JJ (2007) Clinical ophthalmology—a systematic approach, 6th edn. Elsevier, p. 942. ISBN 978-0-08-044969-2
30. Ernest J. *Makulární degenerace* (*Macular degeneration*). Prague, Mladá fronta, 1st Edition, 2010, ISBN 9788020423634
31. Pašta J (2017) Základy očního lékařství (Basics of ophthalmology). Charles University, Karolinum, Prague. ISBN 978-80-246-2460-0
32. Staal J, Abramoff M, Niemeijer M (2004) Ridge based vessel segmentation in color images of the retina. IEEE Trans Med Imaging 23(4):501–509
33. Daugman J (2006) Probing the uniqueness and randomness of iriscodes: results from 200 billion iris pair comparisons. Proc IEEE 94(11):1927–1935
34. Adler A, Youmaran R, Loyka S (2006) Towards a measure of biometric information. In: Canadian conference on electrical and computer engineering, (CCECE'06). pp 210–213
35. Nautsch A, Rathgeb C, Saeidi R, Busch C (2015) Entropy analysis of I-vector feature spaces in duration-sensitive speaker recognition. In: 40th IEEE ICASSP conference, 19–24 April 2015. Brisbane, Australia
36. Decencière E, Cazuguel G, Zhang X, Thibault G, Klein JC, Meyer F, Marcotegui B, Quellec G, Lamard M, Danno R, Elie D, Massin P, Viktor Z, Erginay A, La B, Chabouis A (2013) Teleophta: machine learning and image processing methods for teleophthalmology. IRBM, Elsevier Masson 34(2):196–203
37. Pres M (2016) Bifurcation localization in retina images. MSc thesis, supervised by: Lukas Semerad, Brno University of Technology, Faculty of Information Technology, Czech Republic
38. Mortenson ME (1999) Mathematics for computer graphics applications. Industrial Press Inc., p 264. ISBN 9780831131111
39. Zamir M, Medeiros JA, Cunningham TK (1979) Arterial bifurcations in the human retina. J Gen Physiol 74(4):537–548
40. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville AC, Bengio Y (2014) Generative adversarial networks. CoRR, abs/1406.2661
41. Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR, abs/1511.06434
42. Anatomy and physiology, Connexions Web site. http://cnx.org/content/col11496/1.6/
43. Optimis Fusion. http://www.askin.cz/prednesegmentove/. Accessed 6 June 2018
44. Kowa VX-20. http://dfv.com.au/products/diagnostic/diagnostic-imaging/kowa-vx-20-mydriatic-non-mydriatic-integrated-fundus-camera/. Accessed 6 June 2018
45. https://cryptologicfoundation.org/visit/museum/acquisitions/acquisitionarchivessection/individualequipmentitems/rfsignalgenerator.html. Accessed 11 June 2018
46. http://retinagallery.com/displayimage.php?album=228&pid=2596#top_display_media
47. Nugroho HA et al (2015) Automated microaneurysms (MAs) detection in digital colour fundus images using matched filter. In: 2015 international conference on computer, control, informatics and its applications (IC3INA). pp 104–108

48. Dalton M (2012) Evaluating the risks of retinal detachment in cataract patients. Eyeworld. https://www.eyeworld.org/article-evaluating-the-risks-of-retinal
49. Vlhká forma s edémem (Wet form with edema). Prague, Mladá frona. https://img.mf.cz/062/617/c.jpg. Accessed 25 June 2018
50. Retina Image Bank®. https://imagebank.asrs.org/file/1428/lacquer-cracks

Chapter 12
Vascular Biometric Graph Comparison: Theory and Performance

Arathi Arakala, Stephen Davis and K. J. Horadam

Abstract Vascular biometric templates are gaining increasing popularity due to simple and contact free capture and resilience to presentation attacks. We present the state of the art in Biometric Graph Comparison, a technique to register and compare vascular biometric templates by representing them as formal graphs. Such graphs consist of a set of vertices, representing the branch, termination and crossover points in the vascular pattern, and a set of edges. An edge represents the relationship between a pair of feature points that are directly connected by a vessel segment in a vascular biometric image. We summarise how this information has been successfully used over the past 8 years to improve registration and recognition performance for the vasculature under the palm, wrist, hand and retina. The structural properties of biometric graphs from these modalities differ, with retina graphs having the largest number of vertices on average and the most complex structure, and hand graphs having the smallest number of vertices on average and being the least connected. All vascular graphs have similarities to trees, with the ratio of edges to vertices being close to 1. We describe our most recent algorithms for biometric graph registration and comparison, and our performance results. We are interested in the possibility of using biometric graphs in a template protection scheme based on the paradigm of dissimilarity vectors. As a first step, we wish to improve registration. Certain modalities like retina have an intrinsic reference frame that makes registration more straightforward. Other modalities may not have an intrinsic reference frame. To overcome this, we introduce the notion of anchors—subgraphs of a biometric graph, having between 5 and 10 vertices, that occur consistently in samples from the same individual—that would enable the dissimilarity vector scheme to be applied to any vascular modality. Experiments on palm and wrist databases show that all individuals had at least some sets of 6 captures which could be used to identify an anchor,

A. Arakala (✉) · S. Davis · K. J. Horadam
Discipline of Mathematical Sciences, School of Science, RMIT University, Melbourne, Australia
e-mail: arathi.arakala@rmit.edu.au

S. Davis
e-mail: stephen.davis@rmit.edu.au

K. J. Horadam
e-mail: kathy.horadam@rmit.edu.au

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_12

and anchors were identified in 94% and 88% for the palm and wrist databases, respectively.

Keywords Biometric graphs · Graph comparison · Dissimilarity vector representation · Vascular graphs

12.1 Introduction

The purpose of this Chapter is to provide a single resource for biometric researchers to learn and use the current state of the art in Biometric Graph Comparison[1] for vascular modalities.

Vascular biometric recognition is the process of identifying and verifying an individual using the intricate vascular pattern in the body. Sources of vascular patterns for personal identification and verification are the palm, dorsal hand, wrist, retina, finger and face. Traditionally, vascular patterns have been compared using feature-based or image-based templates. Here we work with feature-based templates only. The basic feature points in a vascular network are vessel terminations (where the vessels leave the image frame of reference or become too fine to be captured in the image), vessel bifurcations (where one vessel splits into two) or (in two-dimensional images) vessel crossovers, where two vessels appear to intersect.

Biometric Graph Comparison (BGC) is a feature-based process, which enhances and improves on traditional point pattern matching methods for many vascular modalities. Its key idea is the replacement of a feature point based representation of a biometric image by a spatial graph based representation, where the graph edges provide a formal and concise representation of the vessel segments between feature points, thus incorporating connectivity of feature points into the biometric template. This added dimension makes the concepts and techniques of graph theory newly available to vascular biometric identification and verification.

In particular, the comparison process is treated as a noisy graph comparison problem, involving local minimisation of a graph editing algorithm. From this, we can extract a Maximum Common Subgraph (MCS), the noisily matched part found to be common to the two graphs being compared. Part of the fascination and value of working with BGC has been to investigate the topology of the MCS: MCSs from two vascular images from the same biometric instance usually look very different from those from different instances.

Over the years since its introduction, BGC has been shown by ourselves and colleagues to improve recognition accuracy, and if more of the topology of the MCS is used to discriminate between genuine and impostor comparisons, this improvement can be quite dramatic. It is also possible to exploit specific graphical characteristics of different modalities to speed up the recognition process.

[1] Previously we used the non-standard term Biometric Graph Matching (BGM).

The Chapter is organised as follows. In Sect. 12.2, we define the vascular Biometric Graph and explain its background and context. A very brief description is given of its extraction from a vascular image. Section 12.3 outlines the formal description of the two components, registration and comparison, of BGC, with some history of its development from its earliest form in [7] to its newest form presented here. (Pseudocode for our Algorithms appears in the Appendix.) In Sect. 12.4, we summarise the body of results in [6–8, 20, 21]. We compare the graph topology of the public retina, hand, palm and wrist databases we use, and describe the topological features of MCSs we have identified from which to derive comparison scores. We provide the supporting evidence for our view that the Biometric Graph representation increases the speed and accuracy of registration, accuracy of comparison, and that using multiple graph structures in the MCS can improve comparison scores over single structures.

Section 12.5 presents one stage of an application of BGC to the problem of privacy protection of vascular templates. The key idea is a feature transformation using a dissimilarity vector approach. Preliminary investigation of the comparison performance of this approach has given encouraging results for retina databases, where an intrinsic alignment exists in the images [5]. A new problem is faced if no such alignment exists. Here we present our first results on a potential solution to this problem, where we look for small but characteristic structures we call "anchors", which appear in sufficiently many of an individual's samples to be used for registration.

12.2 The Biometric Graph

This section presents the Biometric Graph we use for application to vascular biometric modalities. We describe our motivation for using a spatial graph representation over more traditional feature point based templates. We provide a formal definition of a vascular Biometric Graph and give a brief overview of the extraction process.

12.2.1 The Biometric Graph

Biometric Graphs, as we define them, were first introduced in 2011 [17] for the fingerprint modality. Extraction of ridge bifurcations and terminations as feature points is a fundamental technique in a ridge-based modality, and usually, ridge skeletons are also extracted from images. The novelty of the Biometric Graph concept lies in constructing a formal spatial graph from these extracted feature points only. Each feature point is represented as a vertex (also called a node). An edge (also called a link) is a straight line drawn between adjacent pairs of feature points on the skeleton. The edge preserves, in summary form, the connectivity relationship between feature points typically found by tracing along the ridge skeleton. (This differs from the earlier ISO/IEC 19794–8:2006 standard, in which additional "virtual minutiae" and

"continuation minutiae" are inserted along the skeleton, to facilitate piecewise-linear representation of the connecting ridgeline.) A disadvantage of our representation is that more detailed information held by a ridgeline curving between feature points is lost, particularly in regions of high curvature where an edge forms a shortcut between feature points. Figure 12.9 in Appendix 1 demonstrates this. An advantage of our spatial graph representation which can outweigh this loss of information is computational efficiency. An edge can be represented in code concisely by its two end vertices. Furthermore, the full repertoire of graph theoretical techniques is available for data analysis.

12.2.1.1 Vascular Graphs

Direct observation of two-dimensional images of vessel-based modalities shows the physical branching and crossing network of vessels strongly resembles a formal spatial graph drawn in the plane. For example, there is some visible similarity between the pattern of the principal retinal vessels and a rooted tree (with the root vertex in the optic disc), and some visible similarity between the pattern of the principal wrist vessels and a ladder graph or lattice. These similarities to spatial graphs are more pronounced to the naked eye for vascular modalities than in the ridge-based modalities for which we first studied Biometric Graphs. Fundamentally, this is because blood vessels do not often exhibit high curvature, so in most cases the vessel segment between adjacent feature points is quite well represented by a straight line. This was our motivation in [7] for introducing Biometric Graphs and Biometric Graph Comparison into vascular biometric modalities.

The idea of a vascular graph has arisen independently (and at approximately the same time) in the biomedical literature. Drechsler and Laura [13], working with three-dimensional hepatic vessel CT (computed tomography) images of the liver, extract a three-dimensional vascular graph from the vessel skeleton (using voxels not pixels—crossovers do not occur). They classify voxels into three classes: regular, end (terminations) and branch (bifurcations). Branch and end voxels are represented by vertices in the graph, while regular voxels are grouped and represented by edges. The vascular graph provides data for further image recognition, registration and surgical planning. Deng et al. [12] extract a vascular graph (which they term a vascular structure graph model) from the skeleton of the vessel tree in two-dimensional retinal fundus images, to register the images for clinical diagnosis and treatment of retina diseases.

Definition 12.1 A *vascular graph* extracted from a vascular image is a spatial graph with the vessel features of terminations and bifurcations (and crossovers if the image is two-dimensional) forming the graph vertices. A pair of vertices will have an edge between them if and only if we can trace along a vessel from one feature to another, without encountering any other feature in between. More formally, if I is a vascular image then its vascular graph is $g = (V, E, \mu, \nu, A)$, where V is a set of vertices representing the feature points extracted from I, E is a set of edges between those

pairs of vertices representing feature points which are adjacent in I, μ is the vertex labelling function, ν is the edge labelling function and A is the attribute set (which may be empty) comprising a set of vascular attributes that apply to feature points or to the vessel segments connecting them. The *order* of g is the number of vertices $|V|$ and the *size* of g is the number of edges $|E|$. If the vascular image I is of a biometric modality then g is a *(vascular) Biometric Graph (BG)*.

For the BGs in our research, μ associates each vertex with its unique two-dimensional spatial coordinates (x, y) while ν associates each edge with its two-dimensional Euclidean length ℓ and slope θ.

12.2.2 Biometric Graph Extraction

To construct the Biometric Graph from a two-dimensional biometric image, the vessel skeleton is extracted from the image and the feature points are found. The feature points are labelled to form the vertex set, and their coordinates are recorded. The existence of an edge between vertices is determined by tracing the skeleton from each feature point until another is encountered. The length and slope of each edge is calculated and recorded. Other feature point and vessel segment attributes can be calculated at the same time.

Differences in image capture device and lighting source require different image processing techniques for different modalities to reduce noise. There are some common image processing steps in skeleton extraction for any vascular modality, including grayscale conversion, Region-of-Interest (ROI) selection, noise reduction, binarisation and skeleton thinning. Those we employed for palm, dorsal hand, wrist and retina images are described in [6, 8, 20, 21] and the references therein, and will not be further detailed here. For skeleton extraction from finger images, see [23].

A specific problem encountered with extracted skeletons has been the existence of genuine short spurs due to tiny vessels and spurious short spurs due to noise [6, 8, 13, 23]. This is overcome in post-processing by pruning the skeleton of branches shorter than a heuristically selected threshold such as 5, 10 or 15 pixels. For palm vessels, an additional complication has been the inclusion of short to medium length spurs in the skeleton which correspond to skin ridges or flexion creases. Palm principal ridges and creases can be considered as part of the biometric pattern and are difficult to remove completely. However, our experiments have shown that removing the short to medium spurs after the detection of vertices and edges improves the process of registration and comparison. See [8] for details. Wrist vessel skeletons often have segments running perpendicular to the main direction of the vessels, some of which are due to flexion creases, but as some are vessels, these segments are not removed [6].

Feature points are extracted from the 1-pixel-wide skeleton by counting neighbouring pixels in a standard 3×3 pixel window moving across the skeleton. One neighbour indicates a termination pixel, two neighbours indicate a vessel pixel, three neighbours indicate a bifurcation pixel and four or more neighbours indicate a crossover pixel. As a consequence of image noise, neighbouring pixels in the same 3×3 pixel region may be labelled as bifurcation points. To handle this, if a central

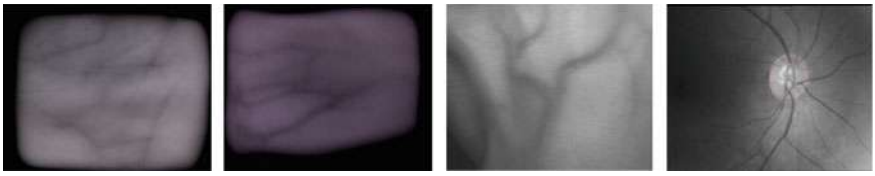
pixel is a bifurcation point and there are two or more neighbours which are bifurcation points on different sides of the central pixel, then only the central pixel is listed as the bifurcation point.

A much faster method of extracting feature points from the vessel skeleton, which may be preferable to the above, is the use of convolutional kernels as in [1].

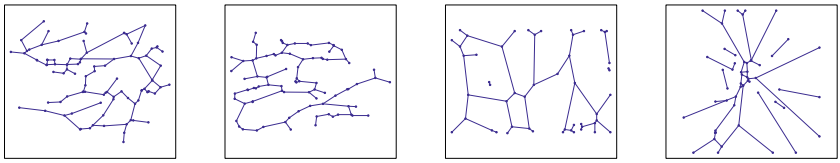
The vertex and edge labels form the basic biometric template. Additional attributes can be extracted from the skeleton to create richer templates. Vertex attributes can include type (termination, branching or crossover). Edge attributes can include the length (as a pixel count) of the skeleton segment between two feature points and the vessel segment average width (or calibre) which can be measured before thinning the skeleton.

Figure 12.1 shows typical vascular pattern images from the databases of each of the four modalities we have investigated and their corresponding Biometric Graphs, extracted as above.

Biometric Graphs have been similarly extracted from skeletons of finger vessels by Nibbelke [23] and from skeletons of face vessels by Gouru [16]. Whilst skeleton tracing is probably the best technique in current use for identifying adjacent feature points in the image skeleton, it is possible that alternatives may prove useful. Khakzar and Pourghassem [19], working with retina images, determine for each pair of feature points whether they are adjacent or not by deleting the two points from the skeleton and checking if the remaining connected components of the skeleton all contain feature points. Existence of a component without feature points means the two points are connected in the skeleton, otherwise they are not. Connectivity is recorded in (the upper half of) an adjacency matrix. However, edge attributes aren't extracted in this approach, and since the adjacency matrix can be found immediately from the edges found by skeleton tracing, it is not clear if the approach has advantages over skeleton tracing.



(a) Palm Image (b) Wrist Image (c) Hand Image (d) Retina Image



(e) Palm Graph (f) Wrist Graph (g) Hand Graph (h) Retina Graph

Fig. 12.1 Vascular patterns from four modalities **a** Palm **b** Wrist **c** Hand and **d** Retina vessels and their corresponding spatial graphs in (**e**–**h**)

12.3 The Biometric Graph Comparison Algorithm

In this section, we present a formal description of the Biometric Graph Comparison Algorithm. The algorithm has two parts: BGR (Registration) which requires 4 steps; and BGC (Comparison), in which the 3 steps are finding the graph edit distance, identifying the Maximum Common Subgraph (MCS) and scoring comparisons using graph-based difference measures.

In our opinion, graph registration is the key component of the algorithm, and is more critical than the graph comparison component. Although it can often be assumed that the capture mechanism enforces an approximate alignment of biometric images in the first place, experience tells us that alignment is seldom ideal, and large differences can occur between captures from the same person, particularly as the time between captures increases. Unless two extracted BGs from the same biometric instance can be aligned well, comparison cannot be effective. Essentially this is because we need a good similarity score for a genuine match, in order to minimise the number of false non-matches. The variance of genuine similarity scores across a population tends to be higher than the variance of impostor similarity scores, which have a distribution of low scores that is roughly independent of registration.

Alignment on a point pattern, such as the set of vertices in a BG, is a standard matching technique. Commonly used methods are the Iterative Closest Point (ICP) algorithm and the Modified Hausdorff Distance (MHD) algorithm. Registration using point pattern alignment algorithms has been previously studied for hand and palm vasculature. In 2009, Chen et al. [10] showed that ICP provided better alignment and consequently superior recognition results than either MHD or point-to-point comparison for palm veins.

In 2014, we showed [21] that for hand veins, registering on edges of BGs using our Biometric Graph Registration (BGR) algorithm gives as good or better recognition performance than either ICP or MHD applied to the point patterns of vertices, especially when the BGs are small. Subsequently, we have modified BGR to permit registration on structures larger than single edges.

12.3.1 BGR-Biometric Graph Registration

Our registration algorithm, in essence, a greedy RANSAC algorithm, looks for structural similarities in a pair of graphs on which to align them, so that the two graphs are in the same spatial frame, free from the effects of translation and rotation of their images during capture.

There is no restriction on what type of structure (i.e. subgraph) can be used for alignment within a particular modality and database. For instance, the algorithm could be tested on a database for different choices of alignment structure, so that the structure giving the best performance could be identified. Or, the frequency of occurrence of different types of structure within the database could be used to select

a preferred structure. Or, if a particular structure was found to be characteristic of a database, appearing more frequently than might be expected in a random spatial graph with comparable numbers of vertices and edges such a "motif" structure could be identified and chosen to align on. Or, it is possible that for a particular modality, each biometric instance exhibits a characteristic structure in most of its images, and such an "anchor" structure could be used for registration.

If the modality possesses an intrinsic coordinate system which can be identified in each database image, registration by the structure might not be required.

To take advantage of the additional structural information in a BG, we align on an edge, or a more complex subgraph such as a *claw*[2] (a degree 3 vertex plus its 3 adjacent edges and 3 neighbouring vertices), a pair of claws joined by a common edge (which we call a *two-claw*), or we could choose a cycle of length 3 or 4. In theory there is no restriction to the type of subgraph chosen for alignment, but computational limits, time constraints and the smaller number of more complex structures present in a BG usually dictate that simpler structures are preferable.

The BGR algorithm is described in more detail in Appendix 2. The algorithm is flexible so that any structure could be used for alignment. It has four steps which are outlined in the following subsection. The four design parameters in the BGR algorithm are a *structure* S, a *similarity score function* f depending on the structure selected, a structure pair *shortlist length* L and a vertex comparison *tolerance* ε. The structures S we have used are: *Edges (E), Claws (C)* and *Two-claws (T)*. If we need to specify the parameters we denote the algorithm by BGR (S, f, L, ε).

Our initial implementation of BGR in 2011 was for BGR (E, f, L, ε) [7]. This has undergone some modification in the intervening years, so that in 2015 we introduced an improved shortlisting mechanism [8] for edge pairs in Step 3 of BGR rather than simply selecting the L highest scoring pairs. We discovered that most edge pairs (in palm BGs) were short and often scored a high rank compared to longer pairs. This prevented longer pairs that gave a better registration from appearing in the top L shortlist. To overcome this, for BGR (E, f, L, ε) we split the set of edge pairs into long and short edge pairs. The mean of the medians of the edge lengths in the two graphs is selected as the threshold. If both edges of an edge pair are longer than this threshold, the edge pair is categorised as long. All other edge pairs are labelled as short. The shortlist consists of the $L/2$ top scoring long edge pairs and the $L/2$ top scoring short edge pairs. This modification ensures that long edge pairs that potentially give better alignment can be included in the shortlist to get a better registration of the graphs. This modification implies that lines 13–19 in the general algorithm in Appendix 2 are run twice, once each for the $L/2$ long and $L/2$ short edges.

In our earlier work [5–8, 20, 21] we assumed that the images in a database are roughly pre-aligned. Here, to provide the most generally applicable registration algorithm, we have modified the similarity scoring of edge pairs in Step 2 of BGR to remove any dependence on pre-alignment. This modification means that in lines

[2]Previously we called this a star, inaccurately, as it is formally a 3-star: an n-star is a vertex of degree $n \geq 1$, plus its adjacent edges and neighbouring vertices.

29–31 of the algorithm in Appendix 2, only the edge lengths are used and edge slopes are not.

12.3.1.1 BGR Algorithm Outline

Step 1: Initialisation Select S, f, L and ε. The two graphs g and g' to be registered are inputs to the algorithm. The registration process begins by identifying and listing all the structures of the selected type S in each graph.

Step 2: Similarity scoring structure pairs Each structure in the first graph g and structure in the second graph g' is compared using f to obtain a similarity score. The similarity function chosen depends on the structure. For example, when edge pairs are compared they are scored based on the similarity of their lengths only (if no pre-alignment is assumed) or of their lengths and slopes (if some pre-alignment is assumed). When claw pairs are compared they are scored based on the similarity of the lengths of their three edges and two included angles. When two-claw pairs are compared, the similarity of the corresponding claw structures and connecting edges determines the score.

Step 3: Shortlisting structure pairs and aligning on them The structure pairs are ordered based on decreasing order of similarity score. The top L high scoring structure pairs (for $S = C$ or $S = T$) or the top $L/2$ short and top $L/2$ long edges (for $S = E$) are shortlisted for further processing. For every shortlisted structure pair, the two graphs are translated and rotated so that a specific part of the structure becomes the origin of the reference frame. For example, if edges are used, the vertex with smaller x coordinate becomes the centre of the coordinate system and the other vertex defines the direction of the positive x-axis. If claws are used, the centre of the claw becomes the origin while the longest edge defines the direction of the positive x-axis. If two-claws are used, the connecting edge defines the coordinate system, again taking the vertex with smaller x coordinate as the origin of the reference frame.

Step 4: Pair alignment scoring and graph registration With both graphs in the same coordinate system, aligned on a shortlisted pair, each vertex in the first graph g is matched to a vertex in the second graph g' by finding the first vertex in g' that is within ε pixels from it. If a vertex in g does not find a corresponding vertex in g' within ε pixels of it, it will not be matched. The total number of matched vertices is normalized by the geometric mean of the number of vertices in the two graphs to provide a rough measure of alignment we call *QuickScore* (QS). That is, if g has n vertices, g' has n' vertices and the aligned graphs have c matched vertices within tolerance ε, the distance between g and g' is calculated to be

$$QS(g, g') = 1 - \frac{c}{\sqrt{n \times n'}}. \tag{12.1}$$

The pair of structures that gives the smallest score is chosen to register g and g'. The resulting registered graphs are denoted g_a and g'_a.

12.3.1.2 Other Approaches to Registration of BGs

Deng et al. [12] in 2010, working with retina BGs, used a two-stage process for registration, also based on edge-to-edge correspondence. Their first (global) registration stage is also a RANSAC variant, where a vertex plus its neighbours in g is compared in g'. In practice, they restrict to degree 2 and 3 vertices, which corresponds to us choosing 2-stars and claws, respectively, as the structure (Their second stage registers vessel shape so is not in the scope of BGR). Using the BG vertex set, they compare the registration performance of several spatial topological graph structures commonly used in computer vision and graph-matching research: the Delaunay triangulation graph (DT), the minimum spanning tree of the DT graph, the k-nearest neighbour graph (KNN) and the minimum spanning tree of the KNN graph. They show that the BG technique substantially outperforms these other topological graph structures in graph registration, and state this is because BG characterises anatomical properties of the retinal vessels while the others do not.

Lupascu et al. [22], working with manually extracted retina BGs and $S = E$, enlarge the feature vector describing each edge from 2 to 9 dimensions by adding further spatial information relating to end vertices and midpoint of the edge, and vary f to be the Euclidean distance in 9-dimensional space. They set $L = 30$ to test g against g' and also test g' against g, choosing only the edge pairs which appear in both lists. Then they use a quadric model to estimate the global transformation between the images using the endpoints of the matched edges.

Nibbelke [23], works with the earlier version of BGR (E, f, L, ε) for finger vessel BGs. He systematically tests alternatives to steps 2 and 3 of the algorithm. First, he tries to improve the rough pre-orientation of images provided by the capture system by testing if the midline of the finger provides an intrinsic reference frame, but finds this not to be robust, leading to worse recognition performance than BGR in several experiments. Orienting all edges in the same direction before comparison does improve performance, as does sorting edge pairs using only their 1-dimensional difference in slope (i.e. using $f = \Delta\theta$ and ignoring their difference in length). He also varies f to include weighting the difference in slope, to overcome the same problem of not finding the best edges for registration in the top L. His best results are found for $f = \Delta\theta$.

If an intrinsic reference frame does exist for pre-alignment in a particular vascular modality, it can be used to register the BGs. We have used this approach effectively with retina BGs in [5] (see Sect. 12.5) taking the centre of the optic disc as the centre of the graph coordinate system while the frame orientation is kept the same.

If no intrinsic reference frame exists for pre-alignment in a particular vascular modality, and we cannot even assume rough pre-alignment by virtue of the capture mechanism, then the BG may provide topological information we can use instead. We investigate this approach in our search for "anchors" in Sect. 12.5.

12.3.2 BGC-Biometric Graph Comparison

The second part of our algorithm is *noisy graph comparison*, to quantify the similarity between a pair g_a and g'_a of registered BGs. If we take advantage of the topology of the BGs in both the registration and noisy graph comparison algorithms, the speed and accuracy of graph comparison can be greatly enhanced.

The algorithm we use is based on using edges as structures as in [20], which is generalised in [6], and further generalised here. The BGC algorithm is flexible, so that any structure can be used. It has three steps: determination of the *minimum graph edit path* between g_a and g'_a, construction of the *Maximum Common Subgraph* (MCS) of g_a and g'_a, and finally, measurement of the difference between g_a and g'_a using the MCS.

We have previously demonstrated that the topology of MCSs generated from pairs of graphs from the same biometric instance (mated comparison) is different from that of MCSs generated from graphs from different instances (non-mathed comparison) [6, 21].

The four design parameters in the BGC algorithm are: a *structure* S, cost matrix *weights* α_1 and α_2 used in the edit distance computation and *measure* d for scoring the distinctiveness or difference of g_a and g'_a. The structures S we have used are *Vertices (V), Edges (E), Claws (C)* and *Two-claws (T)*. If we need to specify the parameters, we denote the algorithm by BGC(S, α_1, α_2, d).

12.3.2.1 BGC Algorithm Outline

Step 1: Graph Edit Distance The comparison process assumes that we have identified and listed all the structures of the selected type S in each registered graph. The registered graphs are compared using an inexact graph matching technique that computes the minimum cost *graph edit path* that converts g_a to g'_a. To do this, we use the Hungarian algorithm based method proposed by Riesen and Bunke [26]. One graph can be converted to another by 3 types of edit operations—insertions, deletions and substitutions. Each edit operation will incur a cost and the *graph edit distance* is the sum of the edit costs.

Selection of the right costs for these operations is critical to getting a meaningful measure of edit distance. The form of cost matrix we use is

$$\mathbf{C} = \begin{bmatrix} \mathbf{C_1} & \mathbf{C_2} \\ \mathbf{C_3} & \mathbf{C_4} \end{bmatrix} \tag{12.2}$$

and depends on the choice of S. If the number of structures in g_a is m and in g'_a is m', $\mathbf{C}$ is a $(m + m') \times (m' + m)$ square matrix, $\mathbf{C_1} = [c_{ij} | 1 \leq i \leq m, 1 \leq j \leq m']$ and c_{ij} represents the the cost of substituting structure u_i of g_a with structure v_j of g'_a. The sub-matrices $\mathbf{C_2}$ and $\mathbf{C_3}$ are square $m \times m$ and $m' \times m'$ matrices, respectively, with all elements outside the main diagonal equal to ∞. The diagonal

elements, $c_{i\delta}$ of $\mathbf{C_2}$ and $c_{\delta j}$ of $\mathbf{C_3}$ indicate the cost of deleting structure i from g_a and inserting structure j into g'_a, respectively. $\mathbf{C_4}$ is an all zero matrix.

Cost matrix $\mathbf{C}$ is fed into the suboptimal optimisation algorithm, which finds a local minimum edit cost. Output will be this lowest cost of converting g_a to g'_a and the list of edit operations that achieve it. The larger the number of structures in each pair of graphs, the bigger the matrices will be and the longer it will take the Hungarian algorithm to compute the optimum result.

The cost matrix entries we use depend on structure S and two weights α_1 and α_2. The case $S = V$ appears below as Example 12.1. Cost matrices for other structures are defined on similar lines (see Appendix 3) , where α_2 will be weighted by the sum of the degrees of all the vertices in the structures.

Example 12.1 (Vertex-based cost matrix, i.e. $m = |V|, m' = |V'|$.) Denote the degree of a vertex by $D(.)$ and the Euclidean distance between two vertex labels (spatial coordinates) by $||.||$. The cost of substituting a vertex v_i of g_a with a vertex v'_j of g'_a is given by

$$c_{ij} = ||v_i, v'_j|| + \varpi_{ij}. \tag{12.3}$$

where ϖ_{ij} is the cheapest cost obtained as output when applying the Hungarian algorithm on a cost matrix for subgraphs g_{v_i} and $g'_{v'_j}$ (see [7] for details). These subgraphs are constructed from the vertices v_i and v'_j and their first-hop neighbourhoods, respectively. The total cost of deleting a vertex will be the sum of the cost of deleting the vertex itself (α_1) and the cost of deleting its neighbourhood vertices (α_2 for each neighbouring vertex),

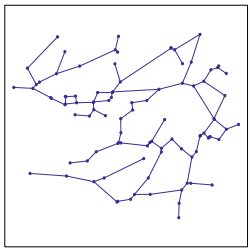
$$c_{i\delta} = \alpha_1 + (\alpha_2 \times D(v_i)). \tag{12.4}$$

Similarly, the cost of inserting a vertex is

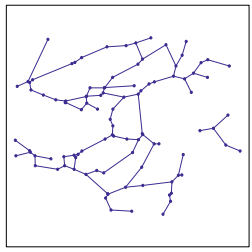
$$c_{\delta j} = \alpha_1 + (\alpha_2 \times D(v'_j)). \tag{12.5}$$

Step 2: Maximum Common Subgraph We use the locally optimal edit path output by the Hungarian algorithm to define a subgraph of g'_a. It includes all those structures of g'_a that are included in the list of substitutions. The structures deleted from g_a and the structures inserted into g'_a are excluded, but any additional corresponding edges are included. This subgraph is called the Maximum Common Subgraph (MCS) of g_a and g'_a as it represents all those structures in g'_a that are "matched" to structures in g_a. We also call it an *S-induced subgraph* of g'_a as the subgraph is induced by the substituted structures in g'_a (Note that defining the MCS as a subgraph of g_a is equivalent.).

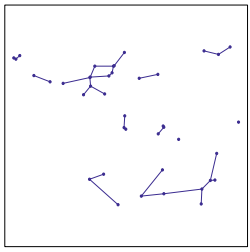
Definition 12.2 Assume BGC$(S, \alpha_1, \alpha_2, -)$ has been applied to registered graphs g_a and g'_a in Step 1 above. Their (S-induced) *Maximum Common Subgraph (MCS)* is the subgraph of g'_a consisting of all structures in g'_a that are included in the list of



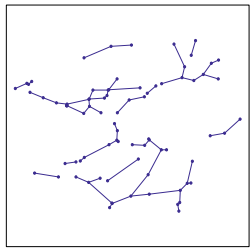
(a) Palm graph 1



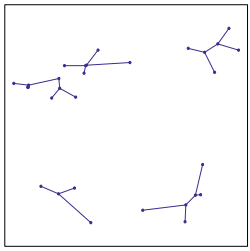
(b) Palm graph 2



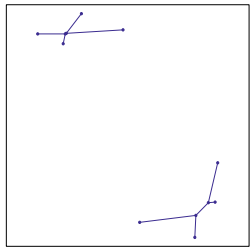
(c) Vertex induced MCS



(d) Edge induced MCS



(e) Claw induced MCS



(f) Two-claw induced MCS

Fig. 12.2 This figure shows the Maximum Common Subgraph between the palm vessel graphs in **a** and **b** resulting from applying BGC with the structure S to be **c** vertices, **d** edges, **e** claws and **f** two-claws. Vertex- and edge-induced MCSs are bigger than claw- and two-claw-induced MCSs as the conditions for the structures to match in the former cases are not as strict as in the latter

substitutions, *together with* any edges that exist between these substituted structures in g'_a, for which a corresponding edge exists in g_a.

Depending on the structure used, the MCS can be vertex induced, edge induced, claw induced or two-claw induced. Figure 12.2 shows each type of MCS for a typical pair of palm BGs from the same biometric instance. The edge induced MCS is the most connected with the richest structure of the four. As S gets more complex than E, the corresponding MCS will be sparser, but the nodes and edges that form part of the MCS will be more reliable. In our experience, the node-induced subgraph tends to miss out on some of the structure that is present in the edge-induced subgraph. Therefore, overall for the biometric graphs in the databases we studied, we prefer S to be edges.

Table 12.1 Difference measures between g_1 and g_2, determined by counts of structures in their MCS

d	d_v	d_e	d_t	$d_{c_1c_2}$	ρ_{c_1}
M	$\lvert V_m \rvert$	$\lvert E_m \rvert$	$\lvert T_m \rvert$	$\lvert V_{c_1} \rvert + \lvert V_{c_2} \rvert$	$\lvert E_{c_1} \rvert$
N_i or N	$\lvert V_i \rvert$	$\lvert E_i \rvert$	$\lvert T_i \rvert$	$\lvert V_i \rvert$	$\lvert V_{c_1} \rvert$

Step 3: Difference Measures The MCS topology is used to define difference measures between g_a and g'_a. There are many potential score functions to separate genuine and impostor comparisons. We have tested 20 which are described in Sect. 12.4.3. A selection of 5, that have proved the most effective, is presented in Table 12.1. One of them, the Bunke–Shearer metric d_v, is already known [9].
Call the two aligned graphs being compared $g_1 = (V_1, E_1)$ and $g_2 = (V_2, E_2)$, with $g_m = (V_m, E_m)$ as their MCS. All sets from g_i, $i \in \{1, 2, m\}$, are subscripted with i. Corresponding sets used to define the measures are the vertex set V_i, the edge set E_i and the set of two-claws T_i. We are also interested in $c_i = (V_{c_i}, E_{c_i})$, $i = 1, 2$, the first and second largest connected components of g_m. The measures have two forms, a distance

$$d = 1 - \frac{M}{\sqrt{N_1 \times N_2}} \tag{12.6}$$

or density

$$\rho = M/N \tag{12.7}$$

as detailed in Table 12.1.

The previous Sections have dealt with the formal aspects of vascular Biometric Graph Comparison. In the next Section, we summarise the performance and practical advantages and disadvantages already discovered using BGC.

12.4 Results

This section will describe the public vascular databases used for BGC so far and compare key BG statistics across them. We summarise experimental results we have obtained by applying BGC to BGs from databases of the four modalities we have studied. The important outcomes from this work are

- that using graph structure in the registration algorithm can increase the speed and accuracy of registration;
- that graph structure in the MCS can be exploited to increase recognition accuracy; and
- that using multiple graph structures can improve similarity scores over single structures.

12.4.1 Vascular Databases

To our knowledge, the BGC algorithm has been tested on five vascular modalities: *Palm* vessels representing the vascular pattern under the palm of the hand; *Wrist* vessels representing the vascular pattern on the inside of the wrists; *Hand* vessels representing the vascular pattern under the skin on the back (dorsal surface) of the hand; *Retina* vessels representing the vascular pattern supplying blood to the retina; and *Finger* vessels representing the vascular pattern under the skin of the finger. We have tested the first four modalities. Finger vessel has been tested by Nibbelke [23], who found that in this case BGC was not competitive with standard point pattern comparison techniques. Gouru [16] in his work on *Face* vessels representing the vascular pattern under the skin of the face, uses a database collected by the University of Houston and extracts BGs. He claims to test BGC but no details are given in [16].

Details of the databases used are summarised in Table 12.2. All are either available for download or on request from the researchers who collected them. The palm and wrist image databases are obtainable from the Poznan University of Technology (PUT) [18] and can be downloaded at http://biometrics.put.poznan.pl/vein-dataset. The hand image databases are from Singapore's Nanyang Technical University [27] with images captured in the near-infrared (SNIR) and far-infrared (SFIR) wavelengths over three sessions each separated by a week. This database exemplifies the kind of variation that can be expected in captures taken across sessions. This is typical of a biometric scenario, where translation and rotation of the images occur between captures due to human factors. Access to this database was obtained by emailing the authors of [27]. Retina images are from the publicly available VARIA database [24] accessible at http://www.varpa.es/research/biometrics.html. In Sect. 12.5 we also refer to the ESRID retina database collected by RMIT University (c.f. [2]). This database can be accessed by emailing the second author of [2]. The finger image database used by Nibbelke [23] is from the University of Twente (UT) and can be accessed by emailing the lead author of [23].

12.4.2 Comparison of Graph Topology Across Databases

In principle, there is no restriction on the structure used by the BG registration and comparison algorithms. In practice, there are restrictions imposed by both the physical form of the vasculature and by the limitations of image resolution and image processing. How do we know what range of options is available?

We have already noted the visible similarity of vascular graphs to trees or ladders. This results from the way the vasculature forms physically. Its purpose is to deliver blood to and from tissues, with the capillaries forming the very fine vessels connecting the arterial system to the venous system. Capillaries are so fine that this interconnection is lost in many images, and vessels appear to terminate rather than

Table 12.2 Vessel image databases used for BGC

Database	Subjects × instances	No. of sessions	Samples/session	Total samples
PUT palm	50 × 2 (left, right)	3	4	600
PUT wrist	50 × 2 (left, right)	3	4	600
SFIR hand[a]	34	1	≥2	173
SNIR hand[a]	123	1	≥2	732
VARIA retina[a]	37	1	≥2	97
ESRID retina	46	1	9	414
UT finger	60 × 6 (different fingers)	2	2	1440

[a]Subset obtained after removal of subjects who had only 1 sample present

rejoin. Typically, vessels do not branch into more than two sub-branches at the same point. As well, while distinct principal veins and arteries might enter the biometric ROI at separate points, all of the vasculature derived from each such vessel will be connected. No sub-branches will actually be disconnected from a parent vessel.

Consequently, in a BG that is perfectly extracted from a high-quality two-dimensional vascular image, there will be relatively few cycles, which will mostly result from vessel crossovers. Vertices will have a low degree (most likely ≤ 4 with maximum degree 4 occurring at crossovers). There will be no isolated vertices (i.e. minimum degree will be 1) and the ratio of edges to vertices (the *density* of the BG) will be similar to that of a tree and so, close to 1. The BG will be connected.

In practice, the image quality will affect the connectivity of the BG, as the image processing algorithm will be unable to extract features from poor quality regions of the image. The more complex the structure of interest, the greater the chance that an occurrence of it will not be extracted in the BG from a particular image, because a component vertex or edge is missing as a result of noise in the image, or suboptimal sensing or image processing. For this reason we are also interested in the largest connected component C_1 of the BG. The size of the largest component is an indication of the amount of noise in the image that has not been compensated for by the image processing.

12.4.2.1 BG Statistics

A very basic question is how much the underlying BG statistics vary for different databases for the same modality, as well as how much they vary for different modalities. In Table 12.3, we record fundamental statistics for different BG databases: numbers of vertices, edges, claws and two-claws, density and number of vertices in the largest connected component C_1 of the BG.

Table 12.3 Mean (standard deviation) of BG topologies for each database. All data except for the last row appear in [6]. Here V is the vertex set, E the edge set, C the claw set, VC_1 the vertex set of the largest connected component C_1, and T the two-claw set

Database	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert E\rvert/\lvert V\rvert$	$\lvert VC_1\rvert$	$\lvert C\rvert$	$\lvert T\rvert$
Palm Left	103 (27.9)	109.9 (19.47)	1.05 (0.03)	93.7 (21.68)	37.6 (7.97)	28.21 (8.39)
Palm Right	98.5 (18.8)	104.1 (20.75)	1.05 (0.03)	89.5 (21.35)	35.4 (8.3)	26.39 (8.53)
Wrist Left	83.07 (16.66)	86.2 (18.61)	1.03 (0.04)	72.1 (19.76)	27.01 (7.85)	19.18 (8.14)
Wrist Right	81.4 (16.11)	83.9 (17.82)	1.02 (0.04)	70.5 (19.43)	25.6 (7.31)	17.66 (7.54)
Hand SFIR	51.7 (9.38)	48.8 (9.6)	0.94 (0.06)	34.6 (12.94)	21.6 (5.35)	19.94 (6.59)
Hand SNIR	39.4 (13.07)	37.6 (13.1)	0.95 (0.03)	27.9 (11.76)	17.3 (6.38)	15.36 (6.47)
Ret. VARIA	70.3 (27.9)	67.1 (29.23)	0.94 (0.07)	48.6 (20.98)	28.9 (13.7)	29.65 (15.28)
Ret. ESRID	146.2 (86.7)	152.6 (92.7)	1.03 (0.04)	109 (68.9)	73.3 (45.1)	75.4 (48.7)

Table 12.3 shows some interesting differences and similarities between the different vascular graphs. All the graphs have density quite close to 1, reflecting their similarity to trees, as expected. The maximum degree of a vertex for each BG was also determined but not recorded here as for every database the mode of the maximum degrees is 3. Between 30 and 40% of vertices in the BGs on average in every database form claws. This indicates that bifurcations are commonplace in our vascular modalities while crossovers are not as commonly seen.

Within modalities, the far-infrared images (SFIR) for hand vessels are superior to the near-infrared (SNIR) as far as being able to extract BGs with usable structure is concerned. With retina, the ESRID graphs are much larger and more connected than VARIA graphs. There is also a large variation across the sizes of the graphs in ESRID when compared to VARIA. The probability of finding a two-claw structure in a retina BG is higher on average than for the other modalities.

The hand BGs are, nonetheless, the smallest and least structured of all modalities, with lower connectivity evidenced by only 70% of their vertices belonging to the largest component. The palm BGs are the second largest (after retina BGs) and most structured, with a higher connectivity than the other graphs demonstrated both by density and the fact that over 90% of the vertices belong to the largest component.

12.4.2.2 Proximity Graphs

Another topological measure we use to characterise the different BG modalities is the distance a BG is from a *proximity graph* on the same vertex set. Proximity graphs were defined by Davis et al. [11]. A proximity graph p_ε on spatial vertex set V is one where a pair of vertices in V have an edge between them if and only if they are less than ε units apart. That is, for a proximity graph, the edges are completely defined by the spatial arrangement of its vertices. The closer a graph is to a proximity graph, the more predictable its edges are.

Table 12.4 [6] The mean (standard deviation) distance of a BG to its nearest proximity graph

Database	Proximity graph distance
Palm	0.017 (0.004)
Wrist	0.022 (0.006)
Hand SFIR	0.032 (0.007)
Hand SNIR	0.046 (0.018)
Retina VARIA	0.031 (0.010)

Thus, if $g = (V, E, \mu, \nu, A)$ is a BG there is a family of proximity graphs $\{p_\varepsilon, \varepsilon \geq 0\}$ defined by V. For each ε, a normalised distance between g and p_ε can be determined from their adjacency matrices, using formulas described in [11]. The value of the proximity graph distance varies from 0 to 1, where zero implies that the graph is a proximity graph. The minimum of these distances over the available range of ε decides the specific value of the bound ε and the closest proximity graph p_ε to g. Table 12.4 shows the average and standard deviation of this distance from a BG to its nearest proximity graph, for each of the databases.

The BGs from palm and wrist vessels have the lowest average distances to a proximity graph, implying that their edges are more predictable than the other BG modalities. Edges are more likely to occur between nearby vertices in palm and wrist BGs than for other modalities, which suggests that the relational information in the graph representation is less surprising (has lower entropy). In principle, the higher the distance, the more promising the vascular pattern is as a biometric modality.

12.4.3 Comparison of MCS Topology in BGC

In previous work [6–8, 20, 21], we have investigated many potential structures and graph statistics in MCSs for their usefulness in BGC for finding information that will satisfactorily separate genuine MCSs from impostor MCSs. Genuine MCSs usually look quite different from impostor MCSs, the latter appearing fragmented and broken as seen in Fig. 12.3. We have attempted in numerous ways to find measures that capture this visually striking difference.

Here, we summarise our findings and discuss reasons for restricting to the structures and corresponding similarity score measures we now use.

Our initial application of BGC [7] was to the retina modality, which has been repeatedly shown (on very small databases) to have high accuracy, with complete separation of genuine and impostor scores typically being demonstrated for vertex comparison approaches. In [7], with manually extracted BGs from the VARIA retina database, we introduced the original BGC (with $S = V$ in the comparison step). We tested 8 measures based on the MCSs for both genuine and impostor comparisons. The 6 normalised quantities were d_v, d_e and the differences $n2$, $n3$, $p2$, $p3$ using Eq. (12.6) corresponding to numbers of vertices of degree $\geq$2, vertices of degree

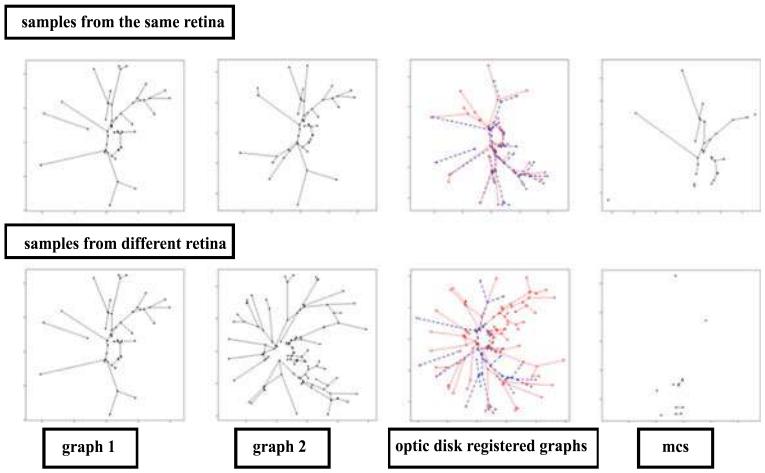


Fig. 12.3 This is an example of the BGC algorithm when two samples from the same retina instance are compared (genuine comparison) versus when two samples from different retina instances are compared (impostor comparison). Note the MCSs are visually different, with the genuine MCS having more vertices and a more complex structure than the impostor MCS

$\geq$3, paths of length 2 and paths of length 3 in g_1, g_2 and g_m, respectively. The 2 unnormalised quantities were the density $\rho_m = |E_m|/|V_m|$ of g_m and the variance σ_D^2 of the degree distribution of g_m. Of these, the score distances for genuine comparisons using vertices of degree $\geq$3 and paths of length 3 were too high to warrant further use. Vertices of degree $\geq$2 and paths of length 2 were also not further considered, as they correlated too highly with either d_v or d_e.

Score fusion using d_v and d_e gave better, but not significantly better, performance than either single measure, probably because these measures are highly correlated. In fact the least correlated measures are d_v, ρ_m and σ_D^2. These measures completely separated scores in two or three dimensions, an improvement on separation in one dimension which is expected to become significant in larger retina databases.

In [20], we developed the first full BGC system to automatically extract retina BGs and compare them, again using the VARIA database. Our intention was to see if the results of [7] could be improved using automatic extraction of BGs. We retained the measure d_v, introduced $d_{c_1c_2}$ based on the two largest connected components of g_m, and replaced σ_D^2 by the maximum degree D_{max} of a vertex in g_m (another unnormalised quantity). Again we showed that using d_v alone gave complete separation in the training set. Using two or all three measures in a combination of an SVM classifier and KDE curves [20] or surfaces gave dramatic improvements in False Match rate (FMR) (up to several orders of magnitude), when False Non-Match Rate (FNMR) was very low.

For hand vessel BGs using the SNIR and SFIR databases in [21], we tested the 7 measures d_v, d_e, $|V_{c_1}|$, $|V_{c_1}| + |V_{c_2}|$, σ_D^2, D_{max} and, for the first time, the average degree μ_D of the vertices in the MCS. The best-separating individual measures

were d_v, d_e and $|V_{c_1}| + |V_{c_2}|$, but as d_v and d_e are highly correlated, the relatively uncorrelated measures d_v, $|V_{c_1}| + |V_{c_2}|$ and σ_D^2 were tested to see if multidimensional scoring would improve performance over individual measures. In contrast to the case for retina, we found little advantage in increasing the number of measures used. We attribute this to the fact that hand BGs are appreciably smaller and more fragmented than retina BGs (see Table 12.3 and [21, Fig. 3]) and will have correspondingly less available topology in their MCSs to exploit.

As a consequence of these experiments, the measures we focussed on were d_v, d_e, $d_{c_1c_2}$, ρ_m; and d_{c_1} and d_{c_2}, the measures using Eq. (12.6) corresponding to the number of vertices in c_1 and c_2, respectively.

For the larger palm vessel BGs, in [8] we test these 6 measures[3] and a further 4: ρ_{c_1}; the ratio of the number of isolated vertices I to the number of connected vertices; the normalised total length d_ℓ of the edges in c_1; and the ratio $n4$ of the number of vertices with degree ≥ 4 in g_m, to $|V_m|$. Equal Error rates using single measures were competitive (under 5%) for within session comparisons for the measures d_v, d_e, d_{c_1}, $d_{c_1c_2}$, ρ and d_ℓ, with three of these, d_v, d_e and $d_{c_1c_2}$, having competitive EERs across sessions as well. The measure d_e outperformed all others. Testing score pairs showed that pairing d_e with any of d_{c_1}, $d_{c_1c_2}$ and d_ℓ improved performance over the single score d_e, with (d_e, d_ℓ) having the maximum gain.

In [6], we tested our ideas on all four modalities using a uniform approach. Our results are outlined in the Sect. 12.4.4, which explains the selection of difference measures in Table 12.1.

- Our attempts to quantify our observation that higher degree vertices occur more frequently in genuine MCSs than in impostor MCSs ($n2, n3, \mu_D, \sigma_D^2, n4$) coalesced in the single measure d_C of claws (i.e. of degree 3 vertices).
- Our efforts to quantify our observation that connected components are larger in genuine MCSs than in impostor MCSs led to the measures d_{c_1}, d_{c_2}, $d_{c_1c_2}$, d_I.
- Our wish to capture some spatial information rather than counts alone resulted in d_ℓ and a new measure d_a found using Eq. (12.6) from the area of the smallest rectangle containing the entire graph.
- Our efforts to quantify our observation that genuine MCSs have higher complexity than impostor MCSs led us to use ρ_m, ρ_{c_1}, D_{max} and a new measure d_t using Eq. (12.6) for the number of two-claws.

For convenience this subsection is summarised in Table 12.5. Measures that we have only tested once before 2017 ($p2$, $p3$, μ_D, $n4$) are not included. Plainly this topic is by no means exhausted.

12.4.4 Comparison of BGC Performance Across Databases

In this subsection, we outline the results and conclusions of our paper [6], in which we evaluated the performance of BGC for the databases of Sect. 12.4.1. The individuals

[3] In fact the corresponding similarity measure $1 - d$ was used for the normalised measures.

Table 12.5 Difference measures used in BGC

Year	Mode	d_v	d_e	d_C	$d_{c_1c_2}$	d_{c_1}	d_{c_2}	ρ_m	ρ_{c_1}	σ_D^2	D_{max}	d_I	d_ℓ	d_a	d_t
2011 [7]	Retina	✓	✓	✓[a]				✓		✓					
2013 [20]	Retina	✓			✓						✓				
2014 [21]	Hand	✓	✓		✓[b]	✓[b]				✓	✓				
2015 [8]	Palm	✓	✓		✓	✓	✓	✓	✓			✓[c]	✓		
2017 [6]	All[d]	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓

[a] $n2$ and $n3$ count the degree ≥ 2 and ≥ 3 vertices. d_C counts the degree 3 vertices (or claws)
[b] Un-normalised counts used
[c] Ratio of I to number of connected vertices, rather than normalised using Eq. (12.6)
[d] Retina, hand, palm, wrist

in each of the five databases (2 for hand) were divided in two, with BGs for one half used for training and the other for testing, to maintain independence. For full details of the experiments, see [6].

The first training experiment was to tune for BGR: to identify the best structure $S \in \{E,\ C,\ T\}$ for graph registration for each database, the optimal pair shortlist length L and the tolerance ε. This list was selected based on observation. For each S, L was varied by steps of 40 through the range [20, 220]. Because accurate registration is crucial to the performance of BGC, we selected the L leading to highest registration accuracy. There is a consequent trade-off in speed versus accuracy, as Table 12.6 demonstrates.

The second training experiment was to tune the parameters of BGC: the structure $S \in \{V,\ E,\ C,\ T\}$ and parameters $\alpha_1,\ \alpha_2$ for the graph edit computations and the difference measure d for scoring MCSs. The parameters were each stepped by 2 in the range [3, 9]. For each database, a subset of 1000 genuine and 1000 impostor comparisons was selected at random and their MCSs computed and scored with the 13 graph measures (see Table 12.5) to find the values giving optimal separation. To check if any combination of measures would improve separation, we combined all 13 measures and used LDA to check this, but found no significant improvement over single measures. For all databases, selecting V for the cost matrix structure and d_v

Table 12.6 [6] The chosen registration structures S and shortlist values L for each database and the average registration times

Database	S	L	Time (s)
Palm	E	220	20
Wrist	E	60	7
Hand SFIR	T	60	0.8
Hand SNIR	E	60	1.9
Retina VARIA	T	100	1.8

Table 12.7 [6] The graph matching parameters chosen based on best performance on the training set

Database	α_1	α_2	Best d	2nd best d	3rd best d
Palm	5	3	d_e	d_v	ρ_{c_1}
Wrist	3	7	d_e	d_v	ρ_{c_1}
SFIR Hand	3	7	d_v	d_e	d_t
SNIR Hand	3	5	d_v	d_e	ρ_{c_1}
Retina VARIA	3	9	d_v	d_e	$d_{c_1c_2}$

Table 12.8 [6] Comparison performance using BGC on the test set at 2 specific thresholds obtained from the *training set experiments*—FMR100 and FMR1000

Threshold	Palm	Wrist	Hand—SFIR	Hand—SNIR	Retina
	FNMR %	FNMR %	FNMR %	FNMR %	FNMR %
FMR100	3.63	26.9	4.39	0.54	0.07
FMR1000	6.242	44.06	8.79	99.72	0.86

or d_e gave the best separation. Table 12.7 summarises the results. The five graph measures on the MCS that we found to be the best difference measures, are d_v, d_e, ρ_{c_1}, $d_{c_1c_2}$ and d_t.

After tuning, we tested BGC on the remaining half of the individuals and determined FMR and FNMR of comparisons at three distance thresholds chosen from the training experiments—EER, FMR100 and FMR1000. ROCs for the SNIR Handvein database training set do not appear in [6] and are given in Appendix 4. All databases other than the wrist, gave error rates under 5% at the EER threshold. Those for palm, hand and retina were comparable with our previous results or the literature. Table 12.8 records our results.

We have already shown for hand vessels [21] that including edge information in BGC improves recognition performance over point pattern comparison. Our final experiment was to apply ICP to register graph pairs, then apply Step 4 of BGR to count matched vertices in the two graphs, again scoring using QuickScore (Eq. (12.1)) for consistency. In all cases, BGC outperformed point pattern comparison using ICP registration. See Table 6 of [6] for exact values.

12.5 Anchors for a BGC Approach to Template Protection

The purpose of biometric authentication is to link a subject unequivocally to the authentication token. The biometric template used to form the token comprises personal and sensitive information and is often encrypted when stored. However, as biometric data is noisy, comparison with an incoming biometric sample cannot be done in the encrypted domain using cryptographic hash functions as these require exactness of data. Consequently, most authentication systems decrypt the stored bio-

metric data, compare the unencrypted templates and make an authentication decision. This makes the biometric template vulnerable during comparison.

Thus, finding a template protection scheme which permits direct comparison of protected templates is desirable. In any such scheme, performance degradation over unprotected comparison must be marginal. Further, the ISO/IEC 24745:2011 standard [25] states the following two criteria to protect biometric information: (a) *Irreversibility* where the biometric raw data cannot be retrieved from the template or token, and (b) *Unlinkability* where multiple independent instances of a subject cannot be linked to identify the subject.

We are interested in the possibility of using biometric graphs in a template protection scheme based on a dissimilarity vector model.

12.5.1 Dissimilarity Vector Templates for Biometric Graphs

We want to investigate the feasibility of protecting a BG template by representing it as a vector of dissimilarities from a fixed set of reference BGs extracted from a separate, external set of instances. Such reference graphs are termed "cohorts". The reason that cohort-based dissimilarity vectors may be a solution to having template-protected biometric comparison for automatic identity authentication is that the biometric sample data need not be stored. Only the cohort graphs and the dissimilarity vector are required for authentication. On the face of it, neither of these reveal any direct information about the biometric sample data of enrolled individuals.

In preliminary work [5], we use retina as an example to conduct the first step of this investigation: to test if the comparison performance of the dissimilarity vector templates is similar to that of unprotected template comparison.

Cohorts are typically not used in existing dissimilarity vector implementations because of the expectation that graphs which are not a member of any class will be dissimilar to all classes and hence not useful for classification. Contrary to this, we found that when retina graphs are registered on the optic disc then graphs extracted from images of the same retina are surprisingly and consistently dissimilar, or similar, to other retina graphs external to the classification set, when the dissimilarity is defined by the BGC algorithm with slack graph comparison parameters.

Figure 12.4 shows an example of a dissimilarity vector for a retina graph.

We have shown that the dissimilarity vector approach is accurately able to compare and verify samples with only a small loss in performance over direct comparison using BGC. Once performance is established, the next step would be to establish rigorous security bounds on irreversability and unlinkability as conducted by Gomez-Barrero et al. [14, 15]. This is an area of future work.

12.5.2 Anchors for Registration

Amongst the modalities presented here, retinae have an intrinsic reference frame defined by the location of optic disk and fovea. Palm vein patterns have a reference

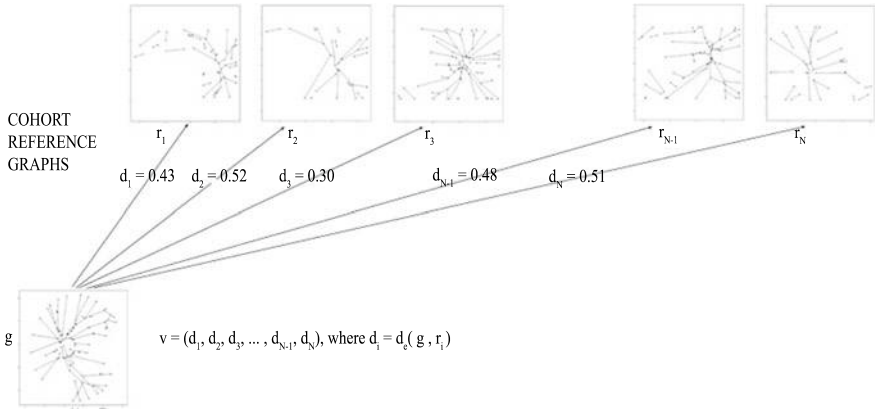


Fig. 12.4 An example of a dissimilarity vector for a retina graph g in ESRID from a set of cohort graphs in VARIA. The dissimilarity vector $v = (d_1, d_2, \cdots, d_N)$ is the vector of dissimilarities from the ordered set of cohort graphs $(r_1, r_2, \cdots, r_N)$. Each $d_i \; \forall 1 \leq i \leq N$ is calculated as $d_i = d_e(g, r_i)$, where d_e is some measure of dissimilarity between graphs g and r_i

frame defined by the hand contour. For other vascular patterns, an intrinsic reference frame has not been identified (for finger graphs, the midline of the finger was found by Nibbelke [23] not to be robust), and because of the noise associated with presentation of a biometric sample and graph extraction, graphs extracted from images from the same individual do not consistently register with reference graphs in the same way when using BGR and are not consistently dissimilar. The retina graphs in both the ESRID and VARIA databases are roughly pre-aligned because the presentation of the retina is always with the head upright, and so a common reference frame for a pair of retina graphs extracted from these images can be found by centring each graph on the centre of the optic disk (also extracted from the associated retina image).

Hence, a barrier to generalising the dissimilarity vector approach to template protection to other vascular graphs is the ability to register presentations of a vascular pattern from the same individual in the same way so that their dissimilarity from a set of reference graphs has the possibility to be consistent. The alternative, which is to use BGR, gives a set of scores that are essentially drawn from a distribution of impostor comparison scores and are different from one sample to the next.

In an attempt to achieve consistent registration, we consider identifying subgraphs of a smaller size that are consistently extracted in multiple presentations of a subject's biometric data despite the noise in the image presentation and extraction process. We term this small subgraph, should it exist, the *anchor* for a set of biometric graphs from an individual.

Definition 12.3 A BG *anchor* for an individual is a small connected subgraph that appears consistently in BGs extracted from multiple good samples from the individual and that does not by itself reveal identifying information about the individual.

Whether such an anchor exists for every enrolled subject is the first question, which we attempt to answer here for two of the databases we have studied. Whether registration on such an anchor then leads to dissimilarity vectors that can be used for accurate classification is a separate question and is future work.

12.5.3 The Search for Anchors

The BGC algorithm can be used recursively to find anchors. Let $g_1, g_2, \ldots, g_n$ be the BGs of the n samples of a subject for which we need to find an anchor.

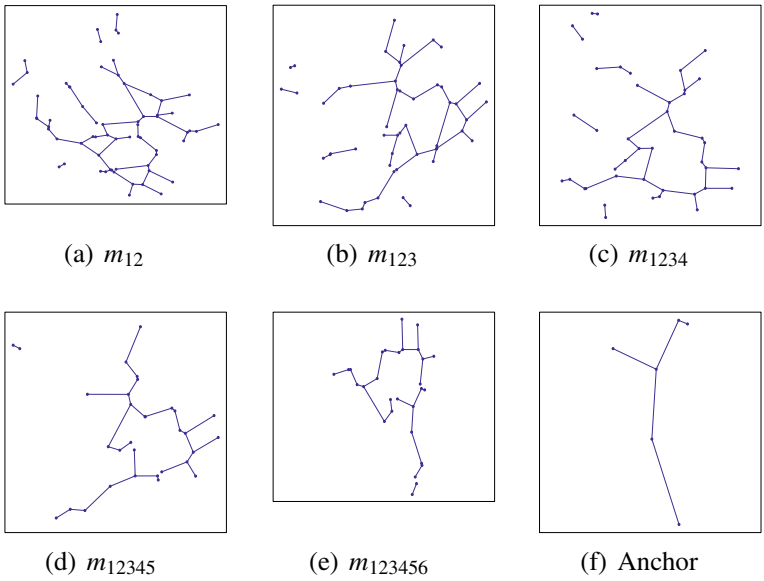
The first step is to use the BGC algorithm to find the MCS between a pair of graphs. Let m_{12} be the MCS of the graphs g_1 and g_2. BGC is then used to find the MCS between m_{12} and the third graph in the list g_3. Let this be denoted by m_{123}. This is the common graph between g_1, g_2 and g_3. If we continue this process, the common graph between the n graphs $g_1, g_2, \ldots, g_n$ is the MCS between $m_{123\cdots n-1}$ and g_n and is denoted by $m_{123\ldots n}$. This graph represents the graph structure that is common to the n samples from a subject. If the graph samples are of high quality, we often find this common graph to be large with significant amount of structure. Therefore, the entire common graph would be inappropriate to use as an anchor associated with a template protection scheme. On the basis of observation and experimentation, we have isolated two criteria to derive an anchor from $m_{123\ldots n}$:

- It is the largest connected component of $m_{123\cdots n}$ that has a minimum of at least 5 vertices and maximum of 10 vertices. This criteria ensures that the anchor is not so large as to reveal significant structure of a subject's BG.
- This connected component must have at least one claw. In cases where there was an absence of a claw (i.e. the component was a path) we observed that the anchor was not uniquely found.

One way to satisfy the above two criteria is to vary the weights α_1 and α_2 in the cost matrix $\mathbf{C}$ of the BGC algorithm used when finding anchors. When α_1 and α_2 are small, the MCS returned will be very small and sparse. As we want to have recursively generated MCSes to have a bit more structure, we found it beneficial to recursively slacken α_2 until we find a common graph of the n graphs that will give an anchor that satisfies the above two conditions.

To study the possibility of finding anchors and the various factors that impact this for a database, we need a database that has multiple samples of the same subject. The PUT datasets of palm and wrist vessels had 12 samples per subject across 3 sessions and were satisfactory for our experiments.

For both databases we chose n, the number of graphs of a subject used to find an anchor, as $n = 6$. We used the remaining 6 samples as test samples to determine if an anchor can be found in a new incoming sample. We set $\alpha_1 = 1$ in the cost matrix $\mathbf{C}$ and recursively increased α_2 from 4 to 16 in steps of 2 in the anchor-finding algorithm.



(a) m_{12} (b) m_{123} (c) m_{1234}

(d) m_{12345} (e) m_{123456} (f) Anchor

Fig. 12.5 This figure shows the common graphs and the final anchor obtained when BGC is used recursively, pairwise on a set of BGs from an individual in the PUT palm database to create the anchor for that individual. Observe that as expected, the size of the common graph as we increase the number of BGs gets smaller. **f** shows the extracted anchor (Graphs are not on the same scale)

Figure 12.5a–e shows the process of recursively applying the BGC algorithm to obtain a common graph among 6 BGs of a subject in the PUT Palm database. We observe that as the number of samples used increases, the common graph tends to get smaller and sparser compared to previous common graphs. For a graph to become part of the common graph it must exist in all the BGs used to form it. The criteria get harder to satisfy as the number of BGs increase. Figure 12.5f and shows the anchor, a subgraph of m_{123456} in Fig. 12.5e, which is the largest connected component of maximum order 10 with at least one claw.

12.5.4 Queries and Discoveries for Anchors

To understand if the use of anchors is practical for registering BGs, we used the palm and wrist databases to investigate the following questions:

1. How likely is it that an anchor cannot be found for a subject in the database and what are the possible reasons for failure to find an anchor?
2. If an anchor is generated using a few samples of a subject, how do we determine if it exists in a new probe sample of the same subject. How reliable is this anchor?

3. How often will an anchor fail to be found in a new probe sample of an enrolled subject? If this happens, what are the causes?

For both databases, we chose 6 BGs from the 12 BGs of each subject in 4 ways giving 4 different attempts at finding an anchor. As the PUT database had 50 subjects, we had 200 trials to find an anchor and we noted the number of trials that failed to find an anchor (first column of Table 12.9).

Once an anchor is found, it needs to be reliably found in a new sample of the same subject. The existence of an anchor in a larger graph can be determined using the BGR algorithm described in Sect. 12.3.1.1. The BGR algorithm will attempt to find an aligning edge between the anchor and a BG of an individual. *Anchor overlap* is defined as the fraction of vertices in the anchor that found a comparison vertex in the BG. 100% overlap indicates the anchor has been exactly found in the BG and can be reliably used to establish a coordinate frame of registration. Figure 12.6 shows an anchor and its overlap in a new probe sample for the palm and wrist BGs. Figure 12.6b, d show an example where the anchor overlap is less than 50%. These are both situations when the anchor has not been found as the anchor just did not exist in the BG. The mean and standard deviation anchor overlap for the palm and wrist databases is shown in column 2 of Table 12.9.

Based on the distribution of anchor overlap in a database, it is possible to choose a minimum value O_t for the anchor overlap to consider an anchor to be reliable. Choosing a specific O_t for each database, we measure for each individual, the number of times in the 6 BGs where the anchor is reliably found. This result is shown in column 3 of Table 12.9.

The distributions of anchor overlap and success rates of finding an anchor reliably for both databases is shown in Fig. 12.8. The source code for the anchor-finding algorithms are available at [3].

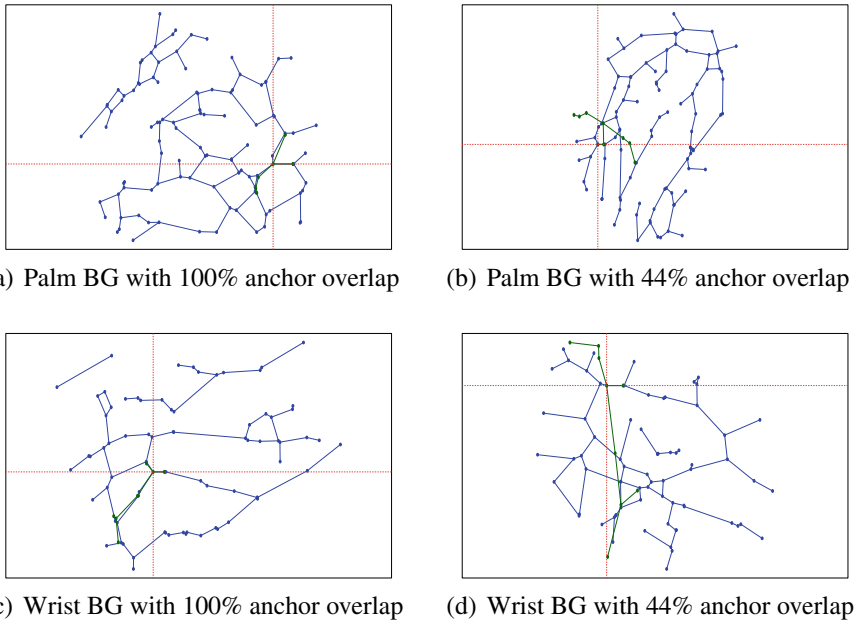
12.5.5 Results

Column 1 of Table 12.9 shows that BGs of an individual in the palm database had a greater chance of generating an anchor than BGs of an individual in the wrist database. Anchors are not generated when the BGs from the samples of the individual fail to find a common subgraph among all of them. This happens if even one BG does not have enough common area of capture amongst the six. Figure 12.7a shows an example where 6 BGs from the wrist vein graph could not generate an anchor. Figure 12.7b shows the BGC applied recursively to get a common graph that did not satisfy the two conditions for an anchor, i.e. there was no component of size between 5 and 10 that had at least one claw.

We next wanted to test, if for every failure in getting an anchor, when the selection of BGs changed, would we be able to get an anchor for the individual? We found that out of the 10 individuals whose trials failed to give an anchor in the palm database, only 2 of the individuals failed again when the selection of BGs changed. For the

Table 12.9 Results from experiments on finding anchors in the PUT palm and wrist databases

Database PUT	Trials that failed to generate anchors (%)	Anchor overlap across the database (%)	Number of reliable anchor registrations per person
Palm	6	75.17 (16.94)	3.68 (1.96)
Wrist	14	76.6 (14.6)	3.96 (1.64)



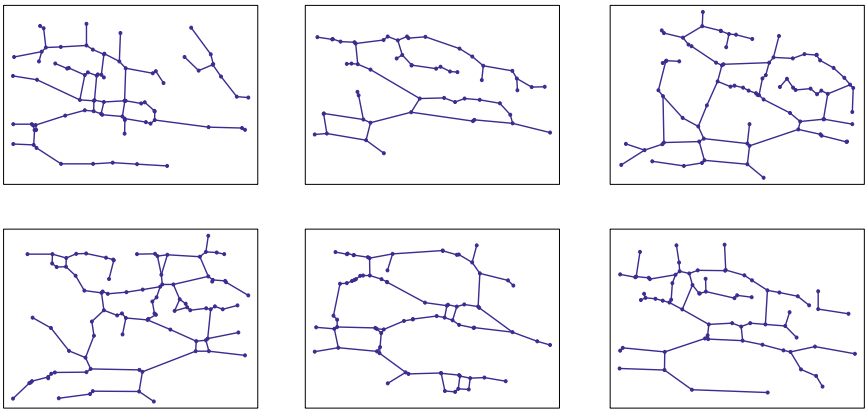
(a) Palm BG with 100% anchor overlap (b) Palm BG with 44% anchor overlap

(c) Wrist BG with 100% anchor overlap (d) Wrist BG with 44% anchor overlap

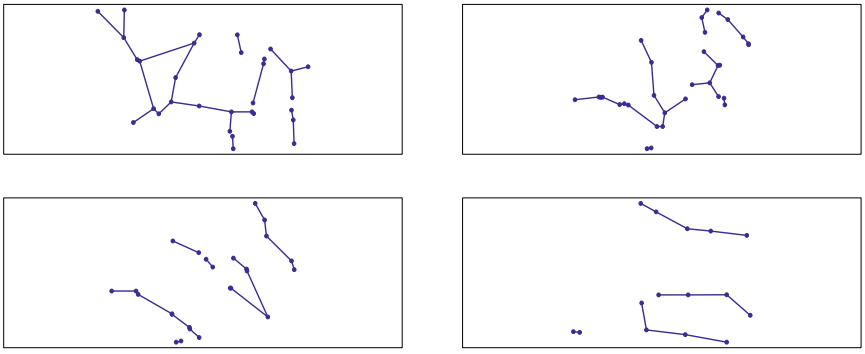
Fig. 12.6 This figure shows examples of Palm and Wrist BGs where the overlap is 100% **(a)** and **(c)**, and where the overlap is less than 50% **(b)** and **(d)**. The anchors are in green and the BGs are in blue

wrist database, 21 individuals failed in a trial to get an anchor, out of them only 3 failed again when the BGs selected were changed. This shows that in practice, if an anchor is not found in a set of samples, it is possible to get an individual to re-enrol until their set of enrolled BGs can give an anchor.

Figure 12.8a, c show the distribution of the anchor overlap measure in the palm and wrist databases. Table 12.9 shows that the mean value of the overlap is over 75% for both. Based on this distribution, we choose O_t to be 70% and measure the number of times we could reliably find an anchor among the remaining 6 BGs that were not used to get the anchor. Figure 12.8b, d show the distribution of number of times the anchor is found reliably in the remaining samples of an individual in the palm and wrist databases, when O_t is set to be 70%. Table 12.9 shows that while the palm BGs were more successful overall in finding anchors, once anchors were found, the



(a) Set of 6 wrist BGs that failed to give an anchor



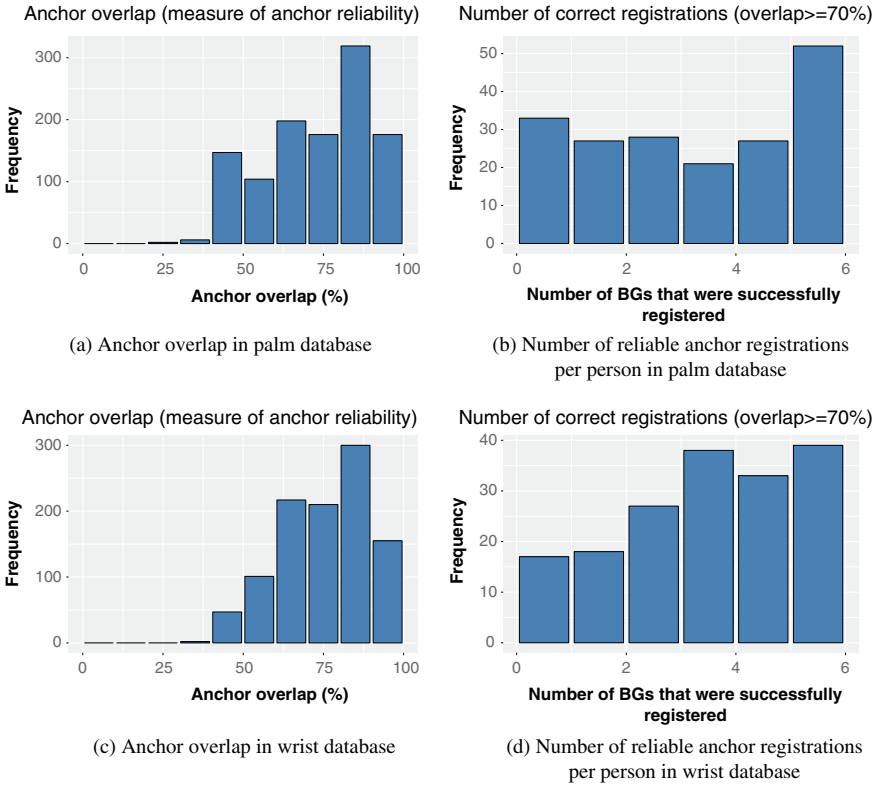
(b) BGC recursively applied to get common graphs

Fig. 12.7 This figure illustrates how 6 wrist BGs can fail to give an anchor. The final common graph did not have a component of maximum size 10 with at least one claw

wrist BGs had a greater chance of finding the anchor in the remaining BGs from the individual. In practice, it would be possible to request resubmission of the biometric sample if the previously identified anchor wasn't found.

12.5.6 Conclusion

This chapter has explained the basic foundations of representing vascular biometric samples as formal graphs. It has generalised the graph registration and comparison algorithms, BGR and BGC, respectively, and summarised our findings from testing the efficiency and effectiveness of BGR and BGC on 4 different modalities–



(a) Anchor overlap in palm database

(b) Number of reliable anchor registrations per person in palm database

(c) Anchor overlap in wrist database

(d) Number of reliable anchor registrations per person in wrist database

Fig. 12.8 This figure shows the histograms of the anchor overlap in the palm and wrist databases. Once an anchor is found, the number of reliable registrations of the anchor per subject, when $O_t = 70\%$ is also shown for both databases. Here test set denotes those 6 BGs not used to get the anchor

palm, wrist, hand and retina. The results show that the relational information in BGs provides better recognition accuracy compared to point pattern approaches. We introduced a modification of BGC with the potential to create a template protection scheme using dissimilarity vectors. We also introduced the concept of anchors, a method to register a BG with a consistent reference frame when, unlike retina, there is no intrinsic reference frame. The choice of anchor and structural restrictions are necessary for them to be used to implement biometric template protection using the dissimilarity vector paradigm. We tested the ease of finding anchors and the likelihood for one to be found reliably in BGs that were not used to identify the anchor. The results show us that with proper selection of BGs, we can always find an anchor for an individual.

In the future we want to apply the concept of anchors to test the accuracy of the dissimilarity vector representation for other modalities like palm vein and hand vein. We also plan to conduct a thorough security analysis of the dissimilarity vector rep-

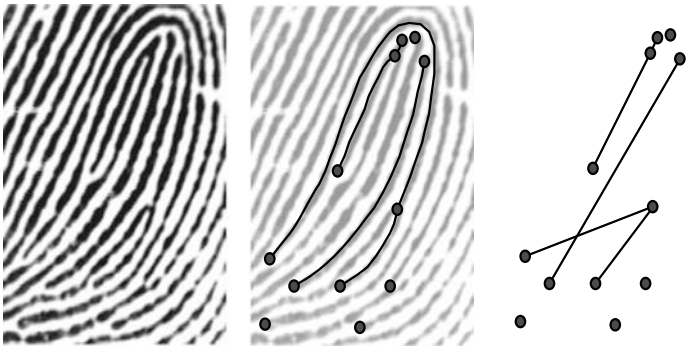


Fig. 12.9 The extraction of a Biometric Graph from a section of fingerprint image. Note that the BG edges represent the ridgeline connectivity relationships between pairs of minutiae, not the ridgeline itself

resentation as a template protection scheme by establishing empirical and theoretical bounds on the irreversibility and unlinkability of the templates on the lines of work conducted by Gomez et al. [14, 15].

Acknowledgements We thank Drs. Hao, Jeffers and Lajervadi for the image processing involved in extracting Biometric Graphs. We thank Dr. Aliahmad and Prof. Kumar for providing the ESRID database. The work has been partly funded by the Australian Research Council grant DP120101188.

Appendix 1

Here is an example of the original Biometric Graph, as introduced in [17] (Fig. 12.9).

Appendix 2

This section gives the pseudocode for the BGR algorithm described in Sect. 12.3.1. It is a corrected and updated version of the algorithm in [6]. The source code for the BGR and BGC algorithms is available at [4].

Require: Graphs g and g' with vertex sets $\mathbf{V} = \{v_1, v_2, \cdots, v_m\}$ and $\mathbf{V}' = \{v'_1, v'_2, \cdots, v'_{m'}\}$ and vertex sets $\mathbf{E} = \{e_1, e_2, \cdots, e_n\}$ and $\mathbf{E}' = \{e'_1, e'_2, \cdots, e'_{n'}\}$, respectively. Let L be the number of structure pairs to shortlist and let ε be the tolerance to match vertex pairs.

Ensure: Aligned graphs g_a and g'_a having same edge links as g and g' but with new spatial coordinates.

$g_a \leftarrow \varnothing$ and $g'_a \leftarrow \varnothing$. ▷ Initialise the registered graphs that will be returned at the end of the algorithm

S $= \{ s_1, s_2, \cdots, s_n \}$ is the list of structures in g.
S' $= \{ s'_1, s'_2, \cdots, s'_{n'} \}$ is the list of structures in g'.
$M_{dist} \leftarrow 0$ ▷ Initialise a matrix of size $n \times n'$ with zeros.
for $a = 1$ to n **do**
  for $b = 1$ to n' **do**
    $d_{ab} =$ StructDist(s_a, s_b, F) ▷ This function returns the distance between the two structures. The flag F indicates if the structure is an edge, claw or two-claw.
    $M_{dist}[a, b] \leftarrow d_{ab}$
  end for
end for

Sort the contents of M_{dist} in increasing order.
$M_{shortlist}$ is a matrix with 3 columns.
Every row m_i stores the 3-tuple (d_{abi}, a_i, b_i).
d_{abi} is taken from the sorted M_{dist} with the first row of $M_{shortlist}$, m_1 having d_{ab1}, the smallest distance.
a_i and b_i indicate the corresponding row and column of d_{abi} in M_{dist}.

$d_{struct} \leftarrow (0, 0, \cdots, 0)_{1 \times L}$ ▷ A vector to store the distances between graphs when aligned on each of the shortlisted structure pairs
for $i = 1$ to L **do**
  $a = a_i, b = b_i$ where $m_i \in M_{shortlist}$
  $g_o =$ TransRot(g, e_a). ▷ Translate and rotate g with respect to the specific edge in the shortlisted structure
  $g'_o =$ TransRot(g', e'_b).
  $d_{struct}[i] =$ QuickScore(g_o, g'_o, ε) ▷ Compute a distance based on vertex correspondence between the translated and rotated graphs
end for

$d_{min} =$ MIN (d_{struct}).
a_{min} and b_{min} are the row and column in $M_{shortlist}$ corresponding to d_{min}.
$g_a =$ TransRot$(g, e_{a_{min}})$.
$g'_o =$ TransRot$(g', e'_{b_{min}})$.
return g_a, g'_a and d_{min}.

function EucDist$(A = (a_1, a_2, \cdots, a_z), B = (b_1, b_2, \cdots, b_z))$
  $d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + ...(a_z - b_z)^2}$ **return** d
end function

function StructDist(s_a, s_b, F) $d_{structPair} \leftarrow \varnothing$
  if $F ==$"edge" **then** ▷

$E_a \leftarrow (l_a, \theta_a)$ ▷ The length and slope of the edge
$E_b \leftarrow (l'_b, \theta'_b)$
$d_{structPair} = \frac{1}{0.5(l_a + l'_b)}$EUCDIST$(E_a, E_b)$ ▷ The Euclidean distance between the lengths l and l' and slopes θ and θ' of the vertex pair.
end if
if $F ==$"claw" **then**
$L_a \leftarrow (l_{1a}, l_{2a}, l_{3a})$ ▷ The three edges of the claw in decreasing order of edge length
$\Theta_a \leftarrow (\theta_{12a}, \theta_{23a})$ ▷ The angles between first and second vertex and the second and third vertex.
$L'_b \leftarrow (l'_{1b}, l'_{2b}, l'_{3b})$
$\Theta_b \leftarrow (\theta'_{12b}, \theta'_{23b})$
$l_\delta \leftarrow$ EUCDIST(L_a, L'_b)
$a_\delta \leftarrow$ EUCDIST(Θ_a, Θ'_b)
$d = l_\delta + a_\delta$
$d_{structPair} = d$
end if
if $F ==$"two-claw" **then** ▷ A two-claw has two-claw structures connected by a common edge
$L_a \leftarrow (l_{1a}, l_{2a}, l_{3a}, l_{4a}, l_{5a}, l_{6a})$
▷ l_1 and l_4 are the longest edges of the first and second claw structures. The other two edges follow the longest edge in decreasing order of length.
$\Theta_a \leftarrow (\theta_{12a}, \theta_{23a}, \theta_{45a}, \theta_{56a})$ ▷ The four internal angles, two each from each of the two-claws.
l_{*a} is the length of the connecting edge between the two-claws in structure a where $* \in \{1, 2, 3, 4, 5, 6\}$.
$L'_b \leftarrow (l'_{1b}, l'_{2b}, l'_{3b}, l'_{4b}, l'_{5b}, l'_{6b})$
$\Theta'_b \leftarrow (\theta'_{12b}, \theta'_{23b}, \theta'_{45b}, \theta'_{56b})$
l'_{*b} is the length of the connecting edge between the two-claws in structure b where $* \in \{1, 2, 3, 4, 5, 6\}$.
$d_1 =$ EUCDIST$(L_a[1:3], L'_b[1:3])$ + EUCDIST$(\Theta_a[1:3], \Theta'_b[1:3])$
$d_2 =$ EUCDIST$(L_a[4:6], L'_b[4:6])$ + EUCDIST$(\Theta_a[4:6], \Theta'_b[4:6])$
$d_3 =$ EUCDIST(l_{*a}, l'_{*b}).
$d_{structPair} = d_1 + d_2 + d_3$
end if
return $d_{structPair}$
end function

function TRANSROT(g, e)
$g_o \leftarrow g$
The vertex of e with the smaller x coordinate will be the origin of the coordinate system.
The edge e will be define the positive direction of the x-axis.

Recalculate all the vertex attributes of g_o in the new coordinate system. **return** g_o.
end function

function QUICKSCORE(g, g', ε)
  Label all vertices of g and g' as unmatched.
  $C = 0$ ▷ Counter for number of vertex pair matches between g and g'
  for $i = 1$ to m **do**
    for $j = 1$ to m' **do**
      if v_i is labelled unmatched and v'_j is labelled unmatched and EUCDIST(q_i, q'_j) $\leq \varepsilon$ **then**
        $C = C + 1$. ▷ v_i matches with v'_j.
        Label v_i and v'_j as matched. ▷ $q_i = (q_{1i}, q_{2i})$ is the vertex attribute of v_i and q'_i is the vertex attribute of v'_i .
      end if
    end for
  end for
  $d = 1 - \frac{C}{\sqrt{m \times m'}}$. **return** d.
end function

Appendix 3

This section presents details of the cost matrices that use complex structures like edges (E), claws (C) and two-claws (T) as structures, as described in Sect. 12.3.2.

Edge-based cost matrix:

Let u_i, v_i be the start and end vertices of e_i in g and u'_i, v'_i be the start and end vertices of e'_j in g'. The cost of substituting e_i with e'_j given by

$$c_{ij} = ||u_i, u'_j|| + ||v_i, v'_j|| \tag{12.8}$$

where $||.||$ denotes Euclidean distance between the spatial coordinates of the vertices. The cost of deleting e_i is

$$c_{i\delta} = \alpha_1 + (\alpha_2 \times (D(u_i) + D(v_i))) \tag{12.9}$$

The cost of inserting e'_j is

$$c_{\delta j} = \alpha_1 + (\alpha_2 \times (D(u'_i) + D(v'_i))) \tag{12.10}$$

where $D()$ denotes vertex degree. α_1 denotes the cost for deleting or inserting an vertex. α_2 denotes the cost for deleting or inserting the vertices neighbouring the start and end vertices of the vertex. The cost matrix will have size $|E| \times |E'|$, where $|.|$ denotes cardinality of the set.

Claw-based cost matrix:

Let c_i and c'_j be the centres of the claws s_i and s'_j in g and g'. Let u_i, v_i, w_i and u'_j, v'_j, w'_j be the end vertices of the three vertices ordered in decreasing order of length for each of the claw structures.

The cost of substituting s_i with s'_j given by

$$c_{ij} = ||c_i, c'_j|| + ||u_i, u'_j|| + ||v_i, v'_j|| + ||w_i, w'_j|| \tag{12.11}$$

where $||.||$ denotes Euclidean distance between the spatial coordinates of the vertices. The cost of deleting s_i is

$$c_{i\delta} = \alpha_1 + (\alpha_2 \times (D(u_i) + D(v_i) + D(w_i))) \tag{12.12}$$

The cost of inserting s'_j is

$$c_{\delta j} = \alpha_1 + (\alpha_2 \times (D(u'_i) + D(v'_i) + D(w'_i))) \tag{12.13}$$

where $D()$ denotes vertex degree. α_1 denotes the cost for deleting or inserting a claw. α_2 denotes the cost for deleting or inserting the vertices neighbouring the end vertices of the claw. The cost matrix will have size $|S| \times |S'|$, where $|.|$ denotes cardinality of the set.

Two-claw-based cost matrix:

Let t_i and t'_j be two-claw structures in g and g'. Each two-claw structures has two-claws connected by a common vertex. Let b_i and c_i be the centre vertices of t_i and $u_i, v_i, w_i, x_i, y_i, z_i$ be the 6 end vertices of two-claw structures ordered on vertex length. u_i and x_i will represent the longest vertices of the claw structures centred on b_i and c_i. Similarly let b'_j and c'_j represent the centres of the claws and $u'_j, v'_j, w'_j, x'_j, y'_j z'_j$ represent the end vertices of the vertices belonging t'_j. The cost of substituting t_i with t'_j given by

$$c_{ij} = ||b_i, b'_j|| + ||c_i, c'_j|| + ||u_i, u'_j|| + ||v_i, v'_j|| + ||w_i, w'_j|| + ||x_i, x'_j|| + ||y_i, y'_j|| + ||z_i, z'_j|| \tag{12.14}$$

where $||.||$ denotes Euclidean distance between the spatial coordinates of the vertices. The cost of deleting t_i is

$$c_{i\delta} = \alpha_1 + (\alpha_2 \times (D(u_i) + D(v_i) + D(y_i) + D(z_i))) \tag{12.15}$$

where the u_i, v_i, y_i, z_i represent the vertices that do not connect the two-claw centres.

The cost of inserting t'_j is

$$c_{\delta j} = \alpha_1 + (\alpha_2 \times (D(u'_i) + D(v'_i) + D(y'_i) + D(z'_i))) \tag{12.16}$$

where $D()$ denotes vertex degree. u'_j, v'_j, y'_j, z'_j represent the vertices that do not connect b'_j and c'_j. α_1 denotes the cost for deleting or inserting a two-claw. α_2 denotes the cost for deleting or inserting the vertices neighbouring the end vertices of the two-claw vertices. The cost matrix will have size $|T| \times |T'|$, where $|.|$ denotes cardinality of the set.

Appendix 4

In [6] we compared the performance of BGC with standard point pattern based comparison algorithms. Each vascular database was divided into a training and

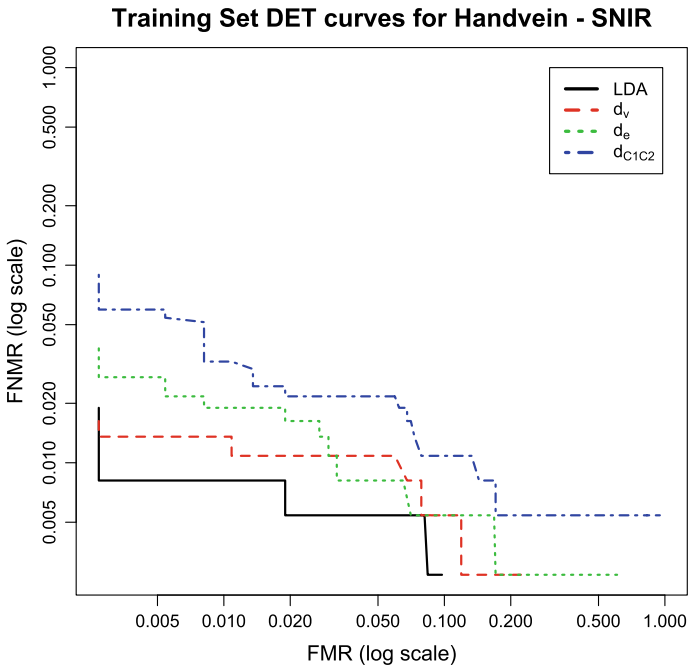


Fig. 12.10 DET curve for the top 3 best performing distance measures in the SNIR handvein training dataset. The performance of each distance measure is compared to that obtained when combining the 3 features using an LDA classifier. Results showed that combining the features did not cause a significant improvement in performance over the best performing measure d_v

testing set. The training set was used to determine the best structure for registration, parameters for the graph comparison algorithm and the best distance measure. Once these parameters were picked they were used to test the performance on the testing database at three thresholds corresponding to three specific points from the training database Detection Error Tradeoff (DET) curves—EER, FMR100 and FMR1000. Figure 12.10 shows the DET curves from the SNIR Handvein training dataset. This was not published in [6]. The DETs for all other modalities are available in Fig. 7 in [6].

References

1. Aastrup Olsen M, Hartung D, Busch C, Larsen R (2011) Convolution approach for feature detection in topological skeletons obtained from vascular patterns. In: 2011 IEEE workshop on computational intelligence in biometrics and identity management (CIBIM), pp 163–167 (2011)
2. Aliahmad B, Kumar DK, Hao H, Kawasaki R (2013) Does fractal properties of retinal vasculature vary with cardiac cycle? In: 2013 ISSNIP biosignals and biorobotics conference: biosignals and robotics for better and safer living (BRC), pp 1–4 (2013). https://doi.org/10.1109/BRC.2013.6487480
3. Arakala A. Source code for anchors for graph registration. https://github.com/ArathiArakalaRMIT/anchor_dissimilarityVector
4. Arakala A. Source code for biometric graph matching for vascular biometrics. https://github.com/ArathiArakalaRMIT/BGM_for_vascular_graphs
5. Arakala A, Davis SA, Hao H, Aliahmad B, Kumar DK, Horadam KJ (2019) Template protected biometrics using dissimilarity vectors (preprint 2019)
6. Arakala A, Davis SA, Hao H, Horadam KJ (2017) Value of graph topology in vascular biometrics. IET Biom 6(2):117–125. https://doi.org/10.1049/iet-bmt.2016.0073
7. Arakala A, Davis SA, Horadam KJ (2011) Retina features based on vessel graph substructures. In: 2011 international joint conference on biometrics (IJCB), pp 1–6 (2011). https://doi.org/10.1109/IJCB.2011.6117506
8. Arakala A, Hao H, Davis S, Horadam K (2015) The palm vein graph-feature extraction and matching. In: Proceedings of the first international conference on information systems security and privacy (ICISSP), pp 56–64. Loire Valley, France. https://doi.org/10.5220/0005239102950303
9. Bunke H, Shearer K (1998) A graph distance metric based on the maximal common subgraph. Pattern Recogn Lett 19(3–4):255–259. https://doi.org/10.1016/S0167-8655(97)00179-7
10. Chen H, Lu G, Wang R (2009) A new palm vein matching method based on ICP algorithm. In: Proceedings of the 2nd international conference on interaction sciences: information technology, culture and human, ICIS '09, pp 1207–1211. ACM, New York, NY, USA. https://doi.org/10.1145/1655925.1656145. URL http://doi.acm.org.ezproxy.lib.rmit.edu.au/10.1145/1655925.1656145
11. Davis S, Abbasi B, Shah S, Telfer S, Begon M (2014) Spatial analyses of wildlife contact networks. J R Soc Interface 12(102). https://doi.org/10.1098/rsif.2014.1004. http://rsif.royalsocietypublishing.org/content/12/102/20141004

12. Deng K, Tian J, Zheng J, Zhang X, Dai X, Xu M (2010) Retinal fundus image registration via vascular structure graph matching. J Biomed Imaging 2010(14):1–13. https://doi.org/10.1155/2010/906067
13. Drechsler K, Laura CO (2010) Hierarchical decomposition of vessel skeletons for graph creation and feature extraction. In: 2010 IEEE international conference on bioinformatics and biomedicine (BIBM), pp 456–461. https://doi.org/10.1109/BIBM.2010.5706609
14. Gomez-Barrero M, Galbally J, Rathgeb C, Busch C (2018) General framework to evaluate unlinkability in biometric template protection systems. IEEE Trans Inf Forensics Secur 13(6):1406–1420. https://doi.org/10.1109/tifs.2017.2788000. https://app.dimensions.ai/details/publication/pub.1100133741. https://app.dimensions.ai. Accedded 28 Feb 2019
15. Gomez-Barrero M, Rathgeb C, Galbally J, Busch C, Fierrez J (2016) Unlinkable and irreversible biometric template protection based on bloom filters. Inf Sci 370:18–32. https://doi.org/10.1016/j.ins.2016.06.046. https://app.dimensions.ai/details/publication/pub.1052550524. https://app.dimensions.ai. Accessed 28 Feb 2019
16. Gouru R (2013) A vascular network matching algorithm for physiological face recognition. Master's thesis, Houston, TX, USA
17. Horadam KJ, Davis SA, Arakala A, Jeffers J (2011) Fingerprints as spatial graphs-nodes and edges. In: Proceedings of international conference on digital image computing techniques and applications, DICTA, pp 400–405. Noosa, Australia
18. Kabacinski R, Kowalski M (2011) Vein pattern database and benchmark results. Electron Lett 47(20):1127–1128
19. Khakzar M, Pourghassem H (2017) A retinal image authentication framework based on a graph-based representation algorithm in a two-stage matching structure. Biocybern Biomed Eng 37(4):742–759. https://doi.org/10.1016/j.bbe.2017.09.001
20. Lajevardi S, Arakala A, Davis S, Horadam K (2013) Retina verification system based on biometric graph matching. IEEE Trans Image Process 22(9):3625–3635
21. Lajevardi S, Arakala A, Davis S, Horadam K (2014) Hand vein authentication using biometric graph matching. IET Biom 3(4):302–313. https://doi.org/10.1049/iet-bmt.2013.0086
22. Lupacu CA, Tegolo D, Bellavia F, Valenti C (2013) Semi-automatic registration of retinal images based on line matching approach. In: Proceedings of the 26th IEEE international symposium on computer-based medical systems, pp 453–456. https://doi.org/10.1109/CBMS.2013.6627839
23. Nibbelke V (2017) Vascular pattern recognition for finger veins using biometric graph matching. Master's thesis, Twente, Netherlands
24. Ortega M, Penedo MG, Rouco J, Barreira N, Carreira MJ (2009) Retinal verification using a feature points-based biometric pattern. EURASIP J Adv Signal Process 2009:1–13
25. Rathgeb C, Busch C (2012) Multi-biometric template protection: issues and challenges. In: Yang J, Xie SJ (eds) New trends and developments in biometrics (chap. 8). IntechOpen, Rijeka. https://doi.org/10.5772/52152
26. Riesen K, Bunke H (2010) Graph classification and clustering based on vector space embedding, 1st edn. World Scientific
27. Wang L, Leedham G, Cho SY (2007) Infrared imaging of hand vein patterns for biometric purposes. IET Comput Vis 1(3–4):113–122

Chapter 13
Deep Sclera Segmentation and Recognition

Peter Rot, Matej Vitek, Klemen Grm, Žiga Emeršič, Peter Peer and Vitomir Štruc

Abstract In this chapter, we address the problem of biometric identity recognition from the vasculature of the human sclera. Specifically, we focus on the challenging task of multi-view sclera recognition, where the visible part of the sclera vasculature changes from image to image due to varying gaze (or view) directions. We propose a complete solution for this task built around Convolutional Neural Networks (CNNs) and make several contributions that result in state-of-the-art recognition performance, i.e.: (i) we develop a cascaded CNN assembly that is able to robustly segment the sclera vasculature from the input images regardless of gaze direction, and (ii) we present ScleraNET, a CNN model trained in a multi-task manner (combining losses pertaining to identity and view-direction recognition) that allows for the extraction of discriminative vasculature descriptors that can be used for identity inference. To evaluate the proposed contributions, we also introduce a new dataset of ocular images, called the *Sclera Blood Vessels, Periocular and Iris* (SBVPI) dataset, which represents one of the few publicly available datasets suitable for research in multi-view sclera segmentation and recognition. The datasets come with a rich

P. Rot and M. Vitek are first authors with equal contributions.

P. Rot · M. Vitek (✉) · Ž. Emeršič · P. Peer
Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, 1000 Ljubljana, Slovenia
e-mail: matej.vitek@fri.uni-lj.si

P. Rot
e-mail: peter.rot93@gmail.com

Ž. Emeršič
e-mail: ziga.emersic@fri.uni-lj.si

P. Peer
e-mail: peter.peer@fri.uni-lj.si

K. Grm · V. Štruc
Faculty of Electrical Engineering, University of Ljubljana, Tržaška cesta 25, 1000 Ljubljana, Slovenia
e-mail: klemen.grm@fe.uni-lj.si

V. Štruc
e-mail: vitomir.struc@fe.uni-lj.si

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_13

set of annotations, such as a per-pixel markup of various eye parts (including the sclera vasculature), identity, gaze-direction and gender labels. We conduct rigorous experiments on SBVPI with competing techniques from the literature and show that the combination of the proposed segmentation and descriptor-computation models results in highly competitive recognition performance.

Keywords Ocular biometrics · Vascular biometrics · Deep learning · Sclera segmentation · Sclera recognition · Dataset · Eye recognition

13.1 Introduction

With the growing need for secure authentication systems, forensic applications and surveillance software, biometric recognition techniques are attracting interest from research groups and private companies trying to improve the current state of the technology and exploit its immense market potential. Among the existing biometric characteristics used in automated recognition systems, ocular traits offer a number of advantages over other modalities such as contactless data acquisition, high recognition accuracy and considerable user acceptance. While iris recognition is the predominant technology in this area, recent research [1, 2] is looking increasingly at additional ocular characteristics that can complement iris-based features and contribute towards more secure and less-spoofable authentication schemes within this branch of biometrics [3].

One trait that presents itself as a particularly viable option in this context is the vasculature of the sclera. The eye's sclera region contains a rich vascular structure that is considered unique for each individual, is relatively stable over time [4] and can hence be exploited for recognition and authentication purposes, as also evidenced by recent research efforts [1, 5]. As suggested in [6], the vascular patterns also exhibit other desirable properties that make them appealing for recognition systems, e.g. the patterns are discernible despite potential eye redness and also in the presence of contact lenses that may adversely affect iris recognition systems. Despite the potential of the sclera vasculature for biometric recognition, research on this particular trait is still in its infancy and several research problems need to be addressed before the technology can be deployed in commercial systems, e.g.:

- The sclera vasculature contains distinct, but also finer blood vessels that need to be segmented from the input ocular images to ensure competitive recognition performance. As emphasised in the introductory chapter of the handbook, these vessels feature very different border types and have a complex texture that is difficult to model, which makes vasculature segmentation highly challenging. To approach this problem, existing solutions typically adopt a two-stage procedure, where the sclera region is first identified in the ocular images and the vasculature structure is then extracted using established (typically unsupervised) algorithms based, for example, on Gabor filters, wavelets, gradient operators and alike [1, 7–9]. While these approaches have shown promise, recent research suggests that supervised techniques result in much better segmentation performance [5, 10], especially if challenging off-angle ocular images need to be segmented reliably. However,

next to the difficulty of sclera vasculature segmentation task itself, the lack of dedicated and suitably annotated datasets for developing supervised techniques has so far represented one of the major roadblocks in the design of competitive sclera recognition systems.

- Due to the particularities (and potentially unconstrained nature) of the image acquisition procedure, ocular images are in general not aligned well with respect to a reference position. Additionally, as the gaze direction may vary from image to image, not all parts of the sclera vasculature are necessarily visible in every captured image. To efficiently compare sclera images and facilitate recognition, discriminative features need to be extracted from the segmented vasculature. These features have to be robust with respect to variations in position, scale and rotation and need to allow for comparisons with only parts of the located vascular structure. Existing solutions, therefore, commonly rely on hand-crafted image descriptors, such as Scale-Invariant Feature Transforms (SIFTs), Histograms of Oriented Gradients (HOGs), Local Binary Patterns (LBPs) and related descriptors from the literature [5, 8, 9]. These local descriptor-based approaches have dominated the field for some time, but, as indicated by recent trends in biometrics [11–14], are typically inferior to learned image descriptors based, for example, on Convolutional Neural Networks (CNNs).

In this chapter, we try to address some of the challenges outlined above and present a novel solution to the problem of sclera recognition built around deep learning and Convolutional Neural Networks (CNNs). Specifically, we first present a new technique for segmentation of the vascular structure of the sclera based on a cascaded SegNet [15] assembly. The proposed technique follows the established two-stage approach to sclera vasculature segmentation and first segments the sclera region from the input images using a discriminatively trained SegNet model and then applies a second SegNet to extract the final vascular structure. As we show in the experimental section, the technique allows for accurate segmentation of the sclera vasculature from the input images even under different gaze directions, thus facilitating feature extraction and sclera comparisons in the later stages.

Next, we present a deep-learning-based model, called ScleraNET, that is able to extract discriminative image descriptors from the segmented sclera vasculature. To ensure that a single (learned) image descriptor is extracted for every input image regardless of the gaze direction and amount of visible sclera vasculature, we train ScleraNET within a multi-task learning framework, where view-direction recognition is treated as a side task for identity recognition. Finally, we incorporate the segmentation and descriptor-computation approaches into a coherent sclera recognition pipeline.

To evaluate the proposed segmentation and descriptor-computation approaches, we also introduce a novel dataset of ocular images, called *Sclera Blood Vessels, Periocular and Iris (SBVPI)* and make it publicly available to the research community. The dataset represents one of the few existing datasets suitable for research in (multi-view) sclera segmentation and recognition problems and ships with a rich set of annotations, such as a pixel-level markup of different eye parts (including the sclera vasculature) or identity, gaze-direction and gender labels. Using the SBVPI dataset, we evaluate the proposed segmentation and descriptor-computation techniques in

rigorous experiments with competing state-of-the-art models from the literature. Our experimental results show that the cascaded SegNet assembly achieves competitive segmentation performance and that the ScleraNET model generates image descriptors that yield state-of-the-art recognition results.

In summary, we make the following contributions in this chapter:

- We propose a novel model for sclera vasculature segmentation based on a cascaded SegNet assembly. To the best of our knowledge, the model represents the first attempt to perform sclera vasculature segmentation in a supervised manner and is shown to perform well compared to competing solutions from the literature.
- We present ScleraNET, a CNN-based model able to extract descriptive image representations from ocular images with different gaze directions. Different from existing techniques, the model allows for the description of the vascular structure of the sclera using a single high-dimensional image descriptor even if the characteristics (position, scale, translation, visibility, etc.) of the vascular patterns vary from image to image.
- We introduce the *Sclera Blood Vessels, Periocular and Iris* (SBVPI) dataset—a dataset of ocular images with a distinct focus on research into sclera recognition. We make the dataset publicly available: http://sclera.fri.uni-lj.si/.

The rest of the chapter is structured as follows: In Sect. 13.2, we survey the relevant literature and discuss competing methods. In Sect. 13.3, we introduce our sclera recognition pipeline and elaborate on the segmentation procedure and ScleraNET models. We describe the novel dataset and its characteristics in Sect. 13.4. All parts of our pipeline are evaluated and discussed in rigorous experiments in Sect. 13.5. The chapter concludes with a brief summary and directions for future work in Sect. 13.6.

13.2 Related Work

In this section, we survey the existing research work relevant to the proposed segmentation and descriptor-computation approaches. The goal of this section is to provide the necessary context for our contributions and motivate our work. The reader is referred to some of the existing surveys on ocular biometrics for a more complete coverage of the field [8, 16–18].

13.2.1 Ocular Biometrics

Research in ocular biometrics dates back to the pioneering work of Daugman [19–21], who was the first to show that the texture of the human iris can be used for identity recognition. Daugman developed an iris recognition system that used Gabor filters to encode the iris texture and to construct a discriminative template that could be used for recognition. Following the success of Daugman's work, many other hand-crafted feature descriptors were proposed [22–25] to encode the texture of the iris.

With recent research on iris recognition moving towards unconstrained image acquisition settings and away from the Near-Infrared (NIR) spectrum towards visible light (VIS) imaging, more powerful image features are needed that can better model the complex non-linear deformations of the iris typically seen under non-ideal lightning conditions and with off-angle ocular images. Researchers are, therefore, actively trying to solve the problem of iris recognition using deep learning methods, most notably, with Convolution Neural Networks (CNNs). The main advantage of using CNNs for representing the iris texture (compared to the more traditional handcrafted image descriptors) is that features can be learned automatically from training data typically resulting in much better recognition performance for difficult input samples. Several CNN-based approaches have been described in the literature over the last few years with highly promising results, e.g. [26–30].

Despite the progress in this area and the introduction of powerful (learned) image descriptors, there are still many open research question related mostly to unconstrained image acquisition conditions (e.g. the person is not looking straight into the camera, eyelashes cover the iris, reflections appear in the images, etc.). To improve robustness of ocular biometric systems in such settings, additional ocular traits can be integrated into the recognition process, such as the sclera vasculature [1] or information from the periocular region [31, 32] . These additional modalities have received significant attention from the research community and are at the core of many ongoing research projects—see, for example, [1, 16, 33–40].

The work presented in this chapter adds to the research outlined above and introduces a complete solution to the problem of multi-view sclera recognition with distinct contributions for vasculature segmentation and descriptor computation from the segmented vascular structure.

13.2.2 Sclera Recognition

Recognition systems based on the vasculature of the sclera typically consist of multiple stages, which in the broadest sense can be categorised into a (i) a vasculature segmentation stage that extracts the vascular structure of the sclera from the image, and (ii) a recognition stage, where the vascular structure is represented using suitable image descriptors and the descriptors are then used for comparisons and subsequent identity inference.

The first stage (aimed at vasculature segmentation) is commonly subdivided into two separate steps, where the first step locates the sclera in the image and the second extracts the vasculature needed for recognition . To promote the development of automated segmentation techniques for sclera segmentation (the first step), several competitions were organised in the scope of major biometric conferences [5, 10, 41, 42]. The results of these competitions suggest that supervised segmentation techniques, based on CNN-based models represent the state of the art in this area and significantly outperform competing unsupervised techniques. Particularly successful here are Convolutional Encoder–Decoder (CED) networks (such as Seg-

Net [15]) , which represent the winning techniques from the 2017 and 2018 sclera segmentation competitions—see [5, 10] for details. In this chapter, we build on these results and incorporate multiple CED models into a cascaded assembly that is shown in the experimental section to achieve competitive performance for both sclera and vasculature segmentation.

To extract the vascular structure from the segmented sclera region, image operators capable of emphasising gradients and contrast changes are typically used. Solutions to this problem, therefore, include standard techniques based, for example, on Gabor filters, wavelets, maximum curvature, gradient operators (e.g. Sobel) and others [1, 7–9]. As suggested in the sclera recognition survey in [8], a common aspect of these techniques is that they are unsupervised and heuristic in nature. In contrast to the outlined techniques, our approach uses (typically better performing) supervised segmentation models, which are possible due to the manual markup of the sclera vasculature that comes with the SBVPI dataset (introduced later in this chapter) and, to the best of our knowledge, is not available with any of the existing datasets of ocular images.

For the recognition stage, existing techniques usually use a combination of image enhancement (e.g. histogram equalisation, Contrast-Limited Adaptive Histogram Equalization (CLAHE) or Gabor filtering [1, 43]) and feature extraction techniques, with a distinct preference towards local image descriptors, e.g. SIFT, LBP, HOG, Gray-level Co-occurrence Matrices, wavelet features or other hand-crafted representations [6, 8, 44–46]. Both dense and sparse (keypoint) image descriptors have already been considered in the literature. With ScleraNET, we introduce a model for the computation of the first learned image descriptor for sclera recognition. We also make the model publicly available to facilitate reproducibility and provide the community with a strong baseline for future research in this area.

13.2.3 Existing Datasets

A variety of datasets is currently available for research in ocular biometrics [16] with the majority of existing datasets clearly focusing on the most dominant of the ocular modalities—the iris [5, 9, 47, 48, 48–55]. While these datasets are sometimes used for research into sclera recognition as well, a major problem with the listed datasets is that they are commonly captured in the Near-Infrared (NIR) spectrum, where most of the discriminative information contained in the sclera vasculature is not easily discernible. Furthermore, existing datasets are not captured with research on vascular biometrics in mind and, therefore, often contain images of insufficient resolution or images, where the Region-Of-Interest (ROI) needed for sclera recognition purposes is not well visible. While some datasets with characteristics suitable for sclera recognition research have been introduced recently (e.g. MASD [5]), these are, to the best of our knowledge, not publicly available.

Table 13.1 shows a summary of some of the most popular datasets of ocular images and also lists the main characteristics of the SBVPI dataset introduced in this

Table 13.1 Comparison of the main characteristics of existing datasets for ocular biometrics. Note that most of the datasets have been captured with research in iris recognition in mind, but have also been used for experiments with periocular (PO) and sclera recognition techniques. The dataset introduced in Sect. 13.4 of this chapter is the first publicly available dataset dedicated to sclera recognition research

Dataset	Modality	Public	NIR/VIS	Image size	# Subjects	# Images	SC-M‡	VS-M*	Gaze
CASIA Iris v1 [47]	Iris	Yes	NIR	320 × 280	54	756	No	No	Static
CASIA Iris v2 [47]	Iris	Yes	NIR	640 × 480	60	2400	No	No	Static
CASIA Iris v3 [47]	Iris	Yes	NIR	640 × 480	> 700	22034	No	No	Static
CASIA Iris v4 [47]	Iris	Yes	NIR	640 × 480	> 2800	54601	No	No	Static
ND-IRIS-0405 [49]	Iris	Yes	NIR	640 × 480	356	64980	No	No	Static
UTIRIS [50]	Sclera, iris	Yes	Both	2048 × 1360	79	1540	No	No	Static
UBIRIS v1 [48]	Sclera, iris	Yes	VIS	800 × 600	241	1877	No	No	Static
UBIRIS v2 [52]	Sclera, PO†, iris	Yes	VIS	400 × 300	261	11102	No	No	Variable
IITD [51]	Iris	Yes	NIR	320 × 240	224	1120	No	No	Static
MICHE-I [53]	Sclera, PO, iris	Yes	VIS	2048 × 1536	92	3732	No	No	Static
UBIPr [54]	PO	Yes	VIS	500 × 400	261	10950	No	No	Variable
IMP [55]	PO	Yes	Both	260 × 270	62	930	No	No	Static
IUPUI [9]	Sclera, PO, iris	No	Both	n/a	44	352	No	No	Variable
MASD [5]	Sclera	No	VIS	7500 × 5000	82	2624	Partial	No	Variable
SBVPI (ours)	Sclera, PO, iris	Yes	VIS	3000 × 1700	55	1858	Full	Partial	Variable

†PO—periocular, ‡SC-M—sclera markup, *VS-M—vasculature markup

chapter. While researchers commonly resort to the UBIRISv1 [48], UBIRISv2 [52], UTIRIS [56], or MICHE-I [53] datasets when conducting experiments on sclera recognition, their utility is limited, as virtually no sclera-specific metadata (e.g. sclera markup, vasculature markup, etc.) is available with any of these datasets. SBVPI tries to address this gap and comes with a rich set of annotations that allow for the development of competitive segmentation and descriptor-computation models.

13.3 Methods

In this section, we present our approach to sclera recognition. We start with a high-level overview of our pipeline and then describe all of the individual components.

13.3.1 Overview

A high-level overview of the sclera recognition pipeline proposed in this chapter is presented in Fig. 13.1. The pipeline consist of two main parts: (i) a cascaded SegNet assembly used for Region-Of-Interest (ROI) extraction and (ii) a CNN model (called ScleraNET) for image-representation (or descriptor) computation.

The cascaded SegNet assembly takes an eye image as input and generates a probability map of the vascular structure of the sclera using a two-step segmentation procedure. This two-step procedure first segments the sclera from the input image and then identifies the blood vessels within the sclera region using a second segmentation step.

The CNN model of the second part of the pipeline, ScleraNET, takes a probability map describing the vascular patterns of the sclera as input and produces a discriminative representation that can be used for matching purposes. We describe both parts of our pipeline in detail in the next sections.

13.3.2 Region-Of-Interest (ROI) Extraction

One of the key steps of every biometric system is the extraction of the Region-Of-Interest (ROI) . For sclera-based recognition systems, this step amounts to segmenting the vascular structure from the input image. This structure is highly discriminative for every individual and can, hence, be exploited for recognition. As indicated in the previous section, we find the vasculature of the sclera in our approach using a two-step procedure built around a cascaded SegNet assembly. In the remainder of this section, we first describe the main idea behind the two-step segmentation procedure, then briefly review the main characteristics of the SegNet model and finally describe

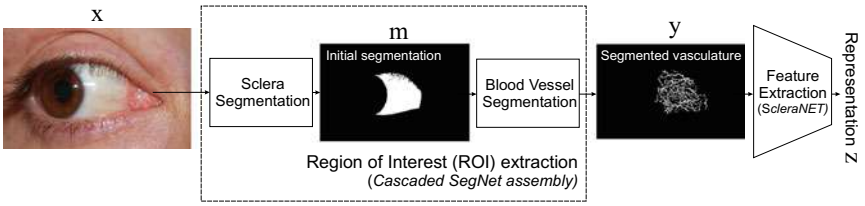


Fig. 13.1 Block diagram of the proposed sclera recognition approach. The vascular structure of the sclera is first segmented from the input image **x** using a two-step procedure. A probability map of the vascular structure **y** is then fed to a CNN model (called ScleraNET) to extract a discriminative feature representation that can be used for sclera comparisons and ultimately recognition. Note that **m** denotes the intermediate sclera region (or masks) generated by the first segmentation step and **z** represent the learned vasculature descriptor extracted by ScleraNET

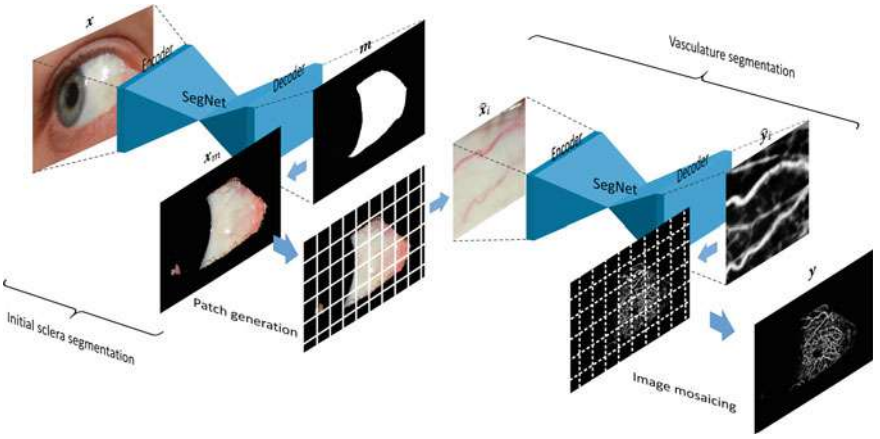


Fig. 13.2 Illustration of the two-step segmentation procedure. In the initial segmentation step, a binary mask of the sclera region is generated by a SegNet model. The mask is used to conceal irrelevant parts of the input image for the second step of the segmentation procedure, where the goal is to identify the vascular structure of the sclera by a second SegNet model. To be able to capture fine details in the vascular structure the second step is implemented in a patch-wise manner followed by image mosaicing. Please refer to the text for an explanation of the symbols used in the image

the training procedure used to learn the parameters of the cascaded segmentation assembly.

13.3.2.1 The Two-Step Segmentation Procedure

The cascaded SegNet assembly used for ROI extraction in our pipeline is illustrated in Fig. 13.2. It consists of two CNN-based segmentation models, where the first tries to generate a binary mask of the sclera region from the input image and the second aims to extract the vascular structure from within the located sclera. The segmenta-

tion models for both steps are based on the recently introduced SegNet architecture from [15]. SegNet was chosen as the backbone model for our segmentation assembly, because of its state-of-the-art performance for various segmentation tasks, competitive results achieved in the recent sclera segmentation competitions [5, 10] and the fact that an open- source implementation is publicly available.[1]

Note that our two-step procedure follows existing unsupervised approaches to sclera vasculature segmentation, where an initial sclera segmentation stage is used to simplify the segmentation problem and constrain the segmentation space for the second step, during which the vasculature is extracted. Our segmentation procedure is motivated by the fact that CNN-based processing does not scale well with image size. Thus, to be able to process high-resolution input images, we initially locate the sclera region from down-sampled images in the first segmentation step and then process image patches at the original resolution in the second segmentation step with the goal of capturing the fine-grained information on the vascular structure of the sclera. Note that this information would otherwise get lost if the images were down-sampled to a size manageable for CNN-based segmentation.

If we denote the input RGB ocular image as $\mathbf{x}$ and the binary mask of the sclera region generated by the first SegNet model as $\mathbf{m}$, then the first (initial) segmentation step can formally be described as follows:

$$\mathbf{m} = f_{\theta_1}(\mathbf{x}), \tag{13.1}$$

where f_{θ_1} denotes the mapping from the input $\mathbf{x}$ to the segmentation result $\mathbf{m}$ by the first CNN model and θ_1 stands for the model parameters that need to be learned during training.

Once the sclera is segmented, we mask the input image $\mathbf{x}$ with the generated segmentation output $\mathbf{m}$ and, hence, exclude all image pixels that do not belong to the sclera from further processing, i.e.:

$$\mathbf{x}_m = \mathbf{x} \odot \mathbf{m}, \tag{13.2}$$

where $\odot$ denotes the Hadamard product. The masked input image $\mathbf{x}_m$ is then used as the basis for the second segmentation step.

Because the vasculature of the sclera comprises large, but also smaller (finer) blood vessels, we use a patch-wise approach in the second segmentation step. This patch-wise approach allows us to also locate large blood vessels within the sclera region, but also the finer ones that would get lost (or overseen) within a holistic segmentation approach due to poor contrast and small spatial area these vessels occupy. Towards this end, we split the masked input image $\mathbf{x}_m$ into M non-overlapping patches $\{\hat{\mathbf{x}}_i\}_{i=1}^{M}$ and subject them to a second segmentation model f_{θ_2} that locates the vascular structure $\hat{\mathbf{y}}_i$ within each patch:

$$\hat{\mathbf{y}}_i = f_{\theta_2}\left(\hat{\mathbf{x}}_i\right), \quad \text{for } i = 1, \ldots, M. \tag{13.3}$$

[1] SegNet on GitHub: https://github.com/alexgkendall/caffe-segnet.

Here, θ_2 denotes the model parameters of the second SegNet model that again need to be learned on some training data.

The final map of the vascular structure $\mathbf{y}$ is generated by re-assembling all generated patches $\hat{\mathbf{y}}_i$ using image mosaicing. Note that different from the first segmentation step, where a binary segmentation mask $\mathbf{m}$ is generated by the segmentation model, $\mathbf{y}$ represents a probability map, which was found to be better suited for recognition purposes than a binary mask of the vasculature (details on possible segmentation outputs are given in Sects. 13.3.2.2 and 13.3.2.3).

To ensure robust segmentation results when looking for the vascular structure of the sclera in the second segmentation step, we use a data augmentation procedure at run-time. Thus, the masked image $\mathbf{x}_m$ is randomly rotated, cropped and shifted to produce multiple versions of the masked sclera. Here, the run-time augmentation procedure selects all image operations with a probability of 0.5 and uses rotations in the range of $\pm 8°$, crops that reduce the image size by up to 1% of the spatial dimensions, and shifts up to ± 20 pixels in the horizontal and up to ± 10 pixels in the vertical direction. Each of the generated images is then split into M patches which are fed independently to the segmentation procedure. The output patches $\hat{\mathbf{y}}_i$ are then reassembled and all generated maps of the vascular structure are averaged to produce the final segmentation result.

As indicated above, the basis for the ROI extraction procedure is the SegNet architecture, which is used in the first, but also the second segmentation step. We, therefore, briefly describe the main SegNet characteristics in the next section.

13.3.2.2 The SegNet Architecture

SegNet [15] represents a recent convolutional encoder–decoder architecture proposed specifically for the task of semantic image segmentation. The architecture consists of two high-level building blocks: an encoder and a decoder. The goal of the encoder is to compress the semantic content of the input and generate a descriptive representation that is fed to the decoder to produce a segmentation output [57, 58].

SegNet's encoder is inspired by the VGG-16 [59] architecture, but unlike VGG-16, the encoder uses only convolutional and no fully connected layers. The encoder consists of 13 convolutional layers (followed by batch normalisation and ReLU activations) and 5 pooling layers. The decoder is another (inverted) VGG-16 model again without fully connected layers, but with a pixel-wise softmax layer at the top. The softmax layer generates a probability distribution for each image location that can be used to classify pixels into one of the predefined semantic target classes. During training, the encoder learns to produce low-resolution semantically meaningful feature maps, whereas the decoder learns filters capable of generating high-resolution segmentation maps from the low-resolution feature maps produced by the encoder [57].

A unique aspect of SegNet are so-called skip-connections that connect the pooling layers of the encoder with the corresponding up-sampling layers of the decoder. These skip-connections propagate spatial information (pooling indices) from one part of the model to the other and help avoid information loss throughout the network.

Consequently, SegNet's output probability maps have the same dimensions (i.e. width and height) as the input images, which allows for relatively precise segmentation. The number of output probability maps is typically equal to the number of semantic target classes—one probability map per semantic class [57]. The reader is referred to [15] for more information on the SegNet model.

13.3.2.3 Model Training and Output Generation

To train the two SegNet models, f_{θ_1} and f_{θ_2}, and learn the model parameters θ_1 and θ_2 needed by our segmentation procedure, we use categorical cross-entropy as our training objective. Once the models are trained, they return a probability distribution over the $C = 2$ target classes (i.e. *sclera vs. non-sclera* for the first SegNet and *blood vessels vs. other* for the second SegNet in the cascaded assembly) for each pixel location. This is, for every location $s = [x, y]^T$ in the input image, the model outputs a distribution $\mathbf{p}_s = [p_{sC_1}, p_{sC_2}]^T \in \mathbb{R}^{C\times 1}$, where p_{sC_i} denotes the probability that the pixel at location s belongs to the ith target class C_i and $\sum_{i=1}^{C} p_{sC_i} = 1$ [57]. In other words, for each input image the model returns two probably maps, which, however are only inverted versions of each other, because $p_{sC_1} = 1 - p_{sC_2}$.

When binary segmentation results are needed, such as in the case of our sclera region $\mathbf{m}$, the generated probability maps are thresholded by comparing them to a predefined segmentation threshold Δ.

13.3.3 ScleraNET for Recognition

For the second part of our pipeline, we rely on a CNN model (called ScleraNET) that serves as a feature extractor for the vasculature probability maps. It needs to be noted that recognition techniques based on the vascular structure of the sclera are sensitive to view (or gaze) direction changes, which affect the amount of visible vasculature and consequently the performance of the final recognition approach. As a consequence, the vasculature is typically encoded using local image descriptors that allow for parts-based comparisons and are to some extent robust towards changes in the appearance of the vascular structure. Our goal with ScleraNET is to learn a single discriminative representation of the sclera that can directly be used for comparison purposes regardless of the given gaze direction. We, therefore, use a Multi-Task Learning (MTL) objective that takes both identity, but also gaze direction into account when learning the model parameters. As suggested in [60], the idea of MTL is to improve learning efficiency and prediction accuracy by considering multiple objectives when learning a shared representation. Because domain information is shared during learning due to the different objectives (pertaining to different tasks), the representations learned by the model offer better generalization ability than representations that rely only on a single objective during training. Since

we try to jointly learn to recognise gaze direction and identity from the vascular structure of the sclera with ScleraNET, the intermediate layers of the model need to encode information on both tasks in the generated representations.

In the following sections, we elaborate on ScleraNET and discuss its architecture, training procedure and deployment as a feature (or descriptor) extractor.

13.3.3.1 ScleraNET Architecture

The ScleraNET model architecture builds on the success of recent CNN models for various recognition tasks and incorporates design choices from the AlexNet [61] and VGG models [59]. We design the model as a (relatively) shallow network with a limited number of trainable parameters that can be learned using a modest amount of training data [11], but at the same time aim for a network topology that is able to generate powerful image representations for recognition. Consequently, we built on established architectural design choices that have proven to work well for a variety of computer vision tasks.

As illustrated in Fig. 13.3 and summarised in Table 13.2, the architecture consists of 7 convolutional layers (with ReLU activations) with multiple max-pooling layers in between followed by a global average pooling layer, one dense layer and two softmax classifiers at the top.

The first convolutional layer uses 128 reasonably large 7×7 filters with a stride of 2 to capture sufficient spatial context and reduce the dimensionality of the generated feature maps. The layer is followed by a max-pooling layer that further reduces the size of the feature maps by $2\times$ along each dimension. Next, three blocks consisting of two convolutional and one max-pooling layer are utilised in the ScleraNET model. Due to the max-pooling layers, the spatial dimensions of the feature maps are halved after each block. To ensure a sufficient representational power of the feature maps, we double the number filters in the convolutional layers after each max-pooling operation. The output of the last of the three blocks is fed to a global average pooling layer and subsequently to a 512-dimensional Fully Connected (FC) layer. Finally, the FC layer is connected to two softmax layers, upon which an identity-oriented and a view-direction-oriented loss is defined for the MTL training procedure. The softmax layers are not used during run-time.

13.3.3.2 Learning Objective and Model Training

We define a cross-entropy loss over each of the two softmax classifiers at the top of ScleraNET for training. The first cross-entropy loss L_1 penalises errors when classifying subjects based on the segmented vasculature, and the second L_2 penalises errors when classifying different gaze directions. The overall training loss is a Multi-Task Learning (MTL) objective:

$$L_{total} = L_1 + \lambda L_2. \tag{13.4}$$

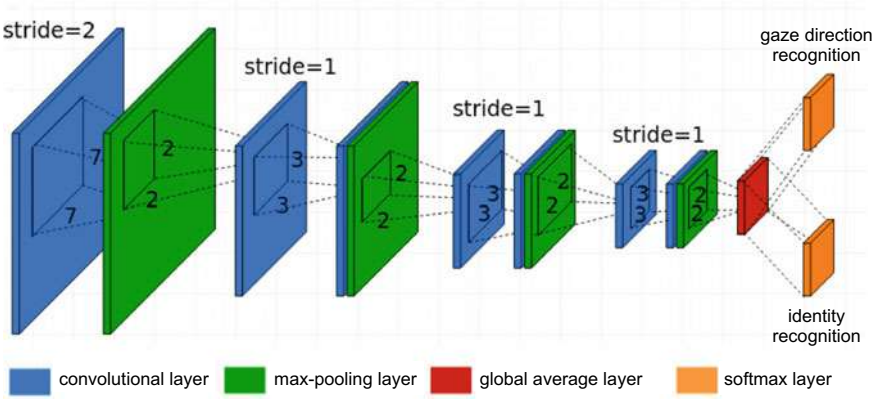


Fig. 13.3 Overview of the ScleraNET model architecture. The model incorporates design choices from the AlexNet [61] and VGG [59] models and relies on a Multi-Task Learning (MTL) objective that combines an identity and gaze-direction-related loss to learn discriminative vasculature representations for recognition

Table 13.2 Summary of the ScleraNET model architecture

No.	Layer type	# Filters	Description
1.	conv	128	7×7 (stride of 2)
2.	max-pooling		2×2
3.	conv	128	3×3 (stride of 1)
4.	conv	128	3×3 (stride of 1)
5.	max-pooling		2×2
6.	conv	256	3×3 (stride of 1)
7.	conv	256	3×3 (stride of 1)
8.	max-pooling		2×2
9.	conv	512	3×3 (stride of 1)
10.	conv	512	3×3 (stride of 1)
11.	max-pooling		2×2
12.	global average pooling		
13.	dense	512	
14.	softmax ($2\times$)		Multi-task objective

To learn the parameters θ of ScleraNET, we minimise the combined loss over some training data and when doing so give equal weights to both loss terms, i.e. $\lambda = 1$.

As suggested earlier, the intuition behind the MTL objective is to learn feature representations that are useful for both tasks and, thus, contribute to (identity) recognition performance as well as to the accuracy of gaze-direction classification. Alternatively, one can interpret the loss related to gaze-direction classification as a regularizer for the identity recognition process [62]. Hence, the additional term helps to learn (to a certain extent) view-invariant representations of the vasculature, or to put it dif-

ferently, it contributes towards more discriminative feature representations across different views.

13.3.3.3 Identity Inference with ScleraNET

Once the ScleraNET model is trained, we make it applicable to unseen identities by performing network surgery on the model and removing both softmax layers. We then use the 512-dimensional output from the fully connected layer as the feature representation of the vascular structure fed as input to the model.

If we again denote the probability map of the vascular structure produced by our two-step segmentation procedure as $\mathbf{y}$ then the feature representation calculation procedure implemented by ScleraNET can be described as follows:

$$\mathbf{z} = g_\theta(\mathbf{y}), \tag{13.5}$$

where g_θ again denotes the mapping from the vascular structure $\mathbf{y}$ to the feature representation $\mathbf{z}$ by the ScleraNET model and θ stands for the model's parameters. The feature representation can ultimately be used with standard similarity measures to generate comparison scores for recognition purposes.

13.4 The Sclera Blood Vessels, Periocular and Iris (SBVPI) Dataset

In this section, we describe a novel dataset for research on sclera segmentation and recognition called *Sclera Blood Vessels, Periocular and Iris* (SBVPI) , which we make publicly available for research purposes from http://sclera.fri.uni-lj.si/. While images of the dataset contain complete eyes, including the iris and periocular region, the focus is clearly on the sclera vasculature, which makes SBVPI the first publicly available dataset dedicated specifically to sclera (segmentation and) recognition research. As emphasised in the introductory chapter of the handbook, currently there exists no dataset designed specifically for sclera recognition, thus, SBVPI aims to fill this gap.

In the remainder of this section, we describe the main characteristics of the introduced dataset, discuss the acquisition procedure and finally elaborate on the available annotations.

13.4.1 Dataset Description

The SBVPI (Sclera Blood Vessels, Periocular and Iris) dataset consists of two separate parts. The first part is a dataset of periocular images dedicated to research in periocluar biometrics and the second part is a dataset of sclera images intended for

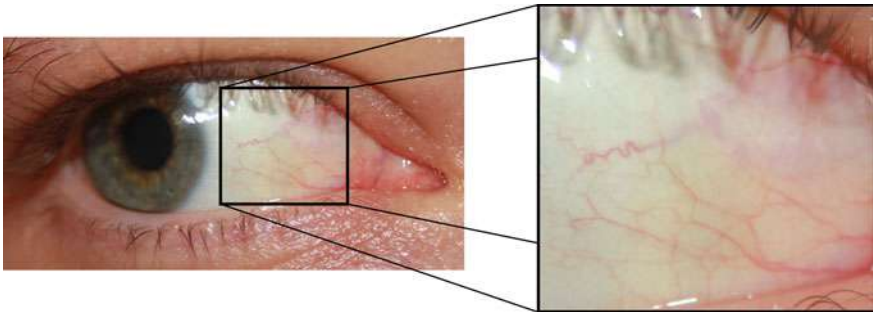


Fig. 13.4 An example image from the SVBPI dataset with a zoomed in region that shows the vascular patterns of the sclera

research into vascular biometrics. We focus in this chapter on the second part only, but a complete description of the data is available from the webpage of SBVPI.

The sclera-related part of SBVPI contains 1858 RGB images of 55 subjects. Images for the dataset were captured during a single recording session using a Digital Single-Lens Reflex camera (DSLR) (Canon EOS 60D) at the highest resolution and quality setting. Macro lenses were also used to capitalise on the quality and details visible in the captured images. The outlined capturing setup was chosen to ensure high-quality images, on which the vascular patterns of the sclera are clearly visible, as shown in Fig. 13.4.

During the image capturing process, the camera was positioned at a variable distance between 20 and 40 centimetres from the subjects. Before acquiring a sclera sample, the camera was always randomly displaced from the previous position by moving it approximately 0–30 cm left/right/up/down. During the camera-position change, the subjects also slightly changed the eyelid position and direction of view. With this acquisition setup, we ensured that the individual samples of the same eye looking at the same direction is always different from all other samples of the same eye looking in the same direction. It is known that the small changes in view direction cause complex non-linear deformations in the appearance of the vascular structure of the sclera [7] and we wanted our database to be suitable for the development of algorithms robust to such kind of changes.

The captured samples sometimes contained unwanted facial parts (e.g. eyebrows, parts of the nose, etc.). We, therefore, manually inspected and cropped (using a fixed aspect ratio) the captured images to ensure that only a relatively narrow periocluar region was included in the final images as shown in the samples in Fig. 13.5. The average size of the extracted Region-Of-Interest (ROI) was around 1700×3000 pixels, which is sufficient to also capture the finer blood vessels of the sclera in addition to the more expressed vasculature. Thus, 1700×3000 px was selected as the target size of the dataset and all samples were rescaled (using bicubic interpolation) to this target size to make the data uniform in size.

The image capturing process was inspired by the MASD dataset [5]. Each subject was asked to look in one of four directions at the time, i.e. straight, left, right and up. For each view direction, one image was captured and stored for the dataset. This

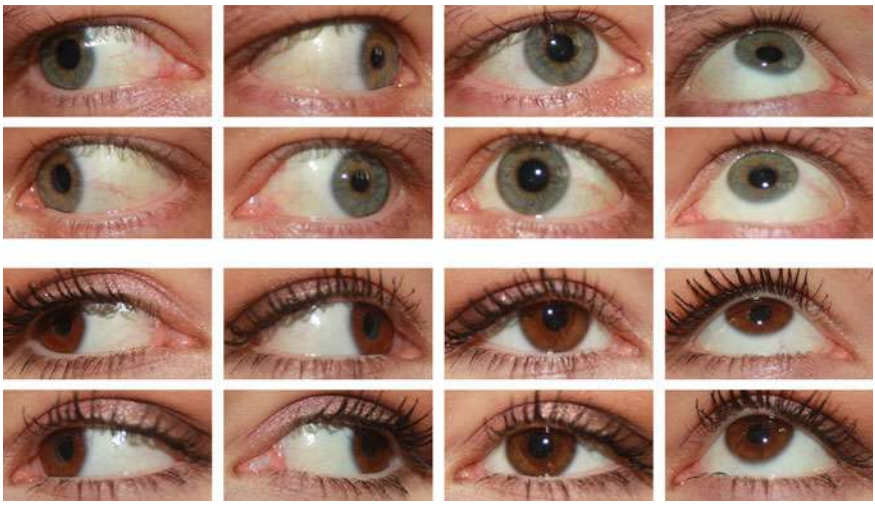


Fig. 13.5 Sample images from the SVBP dataset. The dataset contains high-quality samples with a clearly visible sclera vasculature. Each subject has at least 32 images covering both eyes and 4 view directions, i.e. up, left, right and straight. The top two rows show 8 sample images of a male subject and the bottom two rows show 8 sample images of a female subject from the dataset

process was repeated four times, separately for the left and right eye, and resulted in a minimum of 32 images per subject (i.e. 4 repetitions $\times$ 4 view directions $\times$ 2 eyes)—some subjects were captured more than four times. The images were manually inspected for blur and focus and images not meeting subjective quality criteria were excluded during the recording sessions. A replacement image was taken if an image was excluded. Subjects with sight problems were asked to remove prescription glasses, while contact lenses, on the other hand, were allowed. Care was also taken that no (or minimal) reflections caused by the camera's flash were visible in the images.

The final dataset is gender balanced and contains images of 29 female and 26 male subjects all of Caucasian origin. The age of the subjects varies from 18 to 80 with the majority of subjects being below 35-year old. SBVP contains eyes of different colours, which represents another source of variability in the dataset. A summary of the main characteristics of SBVP is presented in Table 13.3. For a high-level comparison with other datasets of ocular images, including those used for research in sclera recognition, please refer to Table 13.1.

13.4.2 Available Annotations

The dataset is annotated with identity (one of 55 identities), gender (male or female), eye class (left eye or right eye) and view/gaze-direction labels (straight, left, right, up), which are available for each of the 1858 SVBPI sclera images. Additionally,

Table 13.3 Main characteristics of the SVBP dataset

Characteristic	Description
Acquisition device	DSLR camera, Canon EOS 60D + macro lenses
Number of images	1858
Number of subjects	55
Number of images per subject	32 minimum, but variable
Image size	1700 × 3000 px
Available annotations	Identity, gender, view direction, eye markup (segmentation masks)

Fig. 13.6 Examples of the markups available with the SBVPI dataset. All images contain manually annotated irises and sclera regions and a subset of images has a pixel-level markup of the sclera vasculature. The images show (from left to right): a sample image from SBVPI, the iris markup, the sclera markup and the markup of the vascular structure

ground truth information about the location of certain eye parts is available for images in the dataset. In particular, all 1858 images contain a pixel-level markup of the sclera and iris regions, as illustrated in Fig. 13.6. The vascular structure and pupil area are annotated for a subset of the dataset i.e. 130 images. The segmentation masks were generated manually using the GNU Image Manipulation Program (GIMP) and stored as separate layers for all annotated images. The markups are included in SBVPI in the form of metadata.

The available annotations make our dataset suitable for research work on sclera recognition, but also segmentation techniques, which is not the case with competing datasets. Especially the manual pixel-level markup of the sclera vasculature is a unique aspect of the sclera-related part of SBVPI.

13.5 Experiments and Results

In this section, we evaluate our sclera recognition pipeline. We start the section with a description of the experimental protocol and performance metrics used, then discuss the training procedure for all parts of our pipeline and finally proceed to the presentation of the results and corresponding discussions. To allow for reproducibility of our results, we make all models, data, annotations and experimental scripts publicly available through http://sclera.fri.uni-lj.si/.

13.5.1 Performance Metrics

The overall performance of our recognition pipeline depends on the performance of the segmentation part used to extract the vascular structure from the input images and on the discriminative power of the feature representation extracted from the segmented vasculature. In the experimental section we, therefore, conduct separate experiments for the segmentation and feature extraction parts of our pipeline. Next, we describe the performance metrics used to report results for these two parts.

Performance metrics for the segmentation experiments: We measure the performance of the segmentation models using standard performance metrics, such as *precision*, *recall* and the $F1$-*score*, which are defined as follows [57, 58, 63]:

$$precision = \frac{TP}{TP + FP}, \tag{13.6}$$

$$recall = \frac{TP}{TP + FN}, \tag{13.7}$$

$$F1\text{-}score = 2 \cdot \frac{precision \cdot recall}{precision + recall}, \tag{13.8}$$

where TP denotes the number of true positive pixels, FP stands for the number of false positive pixels and FN represents the number of false negative pixels.

Among the above measures, precision measures the proportion of correctly segmented pixels with respect to the overall number of true pixels of the target class (e.g. the sclera region) and, hence, provides information about how many segmented pixels are in fact relevant. Recall measures the proportion of correctly segmented pixels with respect to the overall number of pixels assigned to the target class and, hence, provides information about how many relevant pixels are found/segmented. Precision and recall values are typically dependent—it is possible to increase one at the expense of the other and vice versa by changing segmentation thresholds. If a simple way to compare two segmentation models is required, it is, therefore, convenient to combine precision and recall into a single metric called F1-score, which is also used as an additional performance metric in this work [57].

Note that when using a fixed segmentation threshold Δ, we obtain fixed precision and recall values for the segmentation outputs, while the complete trade-off between precision and recall can be visualised in the form of precision–recall curves by varying the segmentation threshold Δ over all possible values. This trade-off shows a more complete picture of the performance of the segmentation models and is also used in the experimental section [57].

Performance metrics for the recognition experiments: We measure the performance of the feature extraction (and recognition) part of our pipeline in verification experiments and report performance using standard False Acceptance (FAR) and False Rejection error Rates (FRR). FAR measures the error over the illegitimate verification attempts and FRR measures the error over the legitimate verification

attempts. Both error rates, FAR and FRR, depend on the value of a decision threshold (similar to the precision and recall values from the previous section) and selecting a threshold that produces low FAR values contributes towards high FRR scores and vice versa, selecting a threshold that produced low FRR values generates high FAR scores. Both error rates are bounded between 0 and 1. A common practice in biometric research is to report Verification Rates (VER) instead of FRR scores, where VER is defined as 1-FRR [11, 64–66]. We also adopt this practice in our experiments.

Toshow the complete trade-off between FAR and FRR (or VER), we generate Receiver Operating Characteristic (ROC) curves by sweeping over all possible values of the decision threshold. We then report on several operating points from the ROC curve in the experiments, i.e. the verification performance at a false accept rate of 0.1% (VER@0.1FAR), the verification performance at a false accept rate of 1% (VER@1FAR) and the so-called Equal Error Rate (EER), which corresponds to the ROC operating point, where FAR and FRR are equal. Additionally, we provide Area Under the ROC Curve (AUC) scores for all recognition experiments, which is a common measure of the accuracy of binary classification tasks, such as biometric identity verification.

13.5.2 Experimental Protocol and Training Details

We conduct experiments on the SBVPI dataset introduced in Sect. 13.4 and use separate experimental protocols for the segmentation and recognition parts of our pipeline. The protocols and details on the training procedures are presented below.

13.5.2.1 Segmentation Experiments

The segmentation part of our pipeline consists of two components. The first generates an initial segmentation result and locates the sclera region in the input image, whereas the second segments the vasculature from the located sclera.

Sclera segmentation: To train and test the segmentation model for the first component of our pipeline, we split the sclera-related SBVPI data into two (image and subject) disjoint sets:

- *A training set* consisting of 1160 sclera images. These images are further partitioned into two subsets. The first, comprising 985 images, is used to learn the model parameters and the second, comprising 175 images, is employed as the validation set and used to observe the generalization abilities of the model during training and stop the learning stage if the model starts to over-fit.
- *A test set* consisting of 698 sclera images. This set is used to test the final performance of the trained segmentation model and compute performance metrics for the experiments.

To avoid over-fitting, the training data (i.e. 985 images) is augmented by a factor of 40 by left–right flipping, cropping, Gaussian blurring, changing the image brightness and application of affine transformations such as scale changes, rotations (up to $\pm 35°$) and shearing.

Training of the SegNet model for the initial segmentation step (for sclera segmentation) is conducted on a GTX 1080 Ti with 11GB of RAM. We use the Caffe implementation of SegNet made available by the authors[2] for the experiments. The input images are rescaled to fixed size of 360×480 pixels for the training procedure. The model weights are learned using Stochastic Gradient Descent (SGD) and Xavier initialization [67]. The learning rate is set to 0.001, the weight decay to 0.0005, the momentum to 0.9 and the batch size to 4. The model converges after 26,000 iterations.

Vasculature segmentation: The second component of our pipeline requires a pixel-level markup of the vascular structure of the sclera for both the training and the testing procedure. The SBVP dataset contains a total of 130 such images, which are used to learn the SegNet model for this part and assess its performance. We again partition the data into two (image and subject) disjoint sets:

- *A training set* of 98 images, which we split into patches of manageable size, i.e. 360×480 pixels. We generate a total of 788 patches by sampling from the set of 98 training images and randomly select 630 of these patches for learning the model parameters and use the remaining 158 patches as our validation set during training. To avoid over-fitting, we again augment the training patches 40-fold using random rotations, cropping and colour manipulations.
- *A test set* consisting of 32 images. While the test images are again processed patch-wise, we report results over the complete images and not the intermediate patch representations.

To train the segmentation model for the vascular structure of the sclera, we use the same setup as described above for the sclera segmentation model.

13.5.2.2 Recognition Experiments

The vascular structure of the sclera is an epigenetic biometric characteristic with high discriminative power that is known to differ between the eyes of the same subject. We, therefore, treat the left and right eye of each subject in the SBVPI dataset as a unique identity and conduct recognition experiments with 110 identities. Note that such a methodology is common for epigenetic biometric traits and has been used regularly in the literature, e.g. [68, 69].

For the recognition experiments, we split the dataset into subject disjoint training and test sets, where the term subject now refers to one of the artificially generated 110 identities. The training set that is used for the model learning procedure consists of 1043 images belonging to 60 different identities. These images are divided between

[2] Available from: https://github.com/alexgkendall/caffe-segnet.

the actual training data (needed for the learning model parameters) and the validation data (needed for the early stopping criterion) in a ratio of 70% versus 30%. The remaining 815 images belonging to 50 subjects are used for testing purposes.

For the training procedure, we again use a GTX 1080 Ti GPU. We implement our ScleraNET model in Keras and initialize its weights in accordance with the method from [67]. We use the Adam optimizer with a learning rate of 0.001, beta1 equal to 0.9 and beta2 equal to 0.999 to learn the model parameters. We augment the available training data on the fly to avoid over-fitting and to ensure sufficient training material. We use random shifts (±20 pixels in each direction) and rotations (±20°) for the augmentation procedure. The model reaches stable loss values after 70 epochs. As indicated in Sect. 13.3.3.3, once trained, the model takes 400 × 400 px images as input and returns a 512-dimensional feature representation at the output (after network surgery). The input images to the model are complete probability maps of the sclera vasculature down-sampled to the target size expected by ScleraNET. Note that because the down-sampling is performed after segmentation of the vasculature, information on the smaller veins is not completely lost when adjusting for the input size of the descriptor-computation model.

13.5.3 Evaluation of Sclera Segmentation Models

We start our experiments with an evaluation of the first component of the sclera recognition pipeline, which produces the initial segmentation of the sclera region. The goal in this series of experiments is to show how the trained SegNet architecture performs for this task and how it compares to competing deep models and existing sclera segmentation techniques. We need to note that while the error from this stage is propagated throughout the entire pipeline to some extent, these errors are not as critical as long as the majority of the sclera region is segmented from the input images. Whether the segmentation is precise (and able to find the exact border between the sclera region and fine details such as the eyelashes, eyelids, etc.) is not of paramount importance at this stage.

To provide a frame of reference for the performance of SegNet, we implement 4 additional segmentation techniques and apply them to our test data. Specifically, we implement 3 state-of-the-art CNN-based segmentation models and one segmentation approach designed specifically for sclera segmentation. Note that these techniques were chosen, because they represent the top performing techniques from the sclera segmentation competitions of 2017 and 2018. Details on the techniques are given below:

- *RefineNet-50* and *RefineNet-101*: RefineNet [70] is recent deep segmentation model built around the concept of residual learning [71]. The main idea of RefineNet is to exploit features from multiple levels (i.e. from different layers) to produce high-resolution semantic feature maps in a coarse-to-fine manner. Depending on the depth of the model, different variants of the model can be trained.

In this work, we use two variants, one with 50 model layers (i.e. RefineNet-50) and one with 101 layers (i.e. RefineNet-101). We train the models on the same data and with the same protocol as SegNet (see Sect. 13.5.2.1) and use a publicly available implementation for the experiments.[3] Note that RefineNet was the top performer of the sclera 2018 segmentation competition held in conjunction with the 2018 International Conference on Biometrics (ICB) [10].

- *UNet*: The UNet [72] model represents a popular CNN architecture particularly suited for data-scarce image translation tasks such as sclera segmentation. Similarly to SegNet, the model uses an encoder–decoder architecture but ensures information flow from the encoder to the decoder by concatenating feature maps from the encoder with the corresponding outputs of the decoder. We train the models on the same data and with the same protocol as SegNet. For the experiments we use our own Keras (with TensorFlow backend) implementation of UNet and make it publicly available to the research community.[4]
- *Unsupervised Sclera Segmentation (USS)* [73]: Different from the models above, USS represents an unsupervised segmentation technique, which does not rely on any prior knowledge. The technique operates on greyscale images and is based on an adaptive histogram normalisation procedure followed by clustering and adaptive thresholding. Details on the method can be found in [73]. The technique was ranked second in the 2017 sclera segmentation competition. Code provided by the author of USS was used for the experiments to ensure a fair comparison with our segmentation models.

Note that the three CNN-based models produce probability maps for the sclera vasculature, whereas the USS approach returns only binary masks. In accordance with these characteristics we report precision, recall and F1-scores for all tested methods (the CNN models are thresholded with a value of Δ that ensures the highest possible F1-score) in Table 13.4 and complete precision–recall curves only for the CNN-based methods in Fig. 13.7. For both the quantitative results and the performance graphs, we also report standard deviations to have a measure of dispersion across the test set.

The results show that the CNN-based models perform very similarly (there is no statistical difference in performance between the models). The unsupervised approach USS, on the other hand, performs somewhat worse, but the results are consistent with the ranking reported in [5]. Overall, the CNN models all achieve near-perfect performance and are able to ensure F1-scores of around 0.95. Note that such high results suggest that performance for this task is saturated and further improvements would likely be a consequence of over-fitting to the dataset and corresponding manual annotations.

The average processing time per image (calculated over a test set of 100 images) is 1.2s for UNet, 0.6s for RefineNet-50, 0.8s for RefineNet-101, 0.15s for SegNet and 0.34s for USS. In our experiments, SegNet is the fastest of the tested models.

We show some examples of the segmentation results produced by the tested segmentation models in Fig. 13.8. Here, the first column shows the original RGB ocular

[3] Available from https://github.com/guosheng/refinenet.

[4] Available from: http://sclera.fri.uni-lj.si/.

Table 13.4 Segmentation results generated based on binary segmentation masks. For the CNN-based models, the masks are produced by thresholding the generated probability maps with a value of Δ that ensures the highest possible F1-score, whereas the USS approach is designed to return a binary mask of the sclera region only. Note that all CNN perform very similarly with no statistical difference in segmentation performance, while the unsupervised USS approach performs somewhat worse. The reported performance scores are shown in the form $\mu \pm \sigma$, computed over all test images

Algorithm	Precision	Recall	F1-score
UNet [72] (ours)	0.936 ± 0.044	0.930 ± 0.037	0.933 ± 0.037
RefineNet-50 [70] (ours)	0.959 ± 0.020	0.959 ± 0.020	0.959 ± 0.018
RefineNet-101 [70] (ours)	0.953 ± 0.025	0.951 ± 0.023	0.952 ± 0.021
SegNet [5, 57] (ours, this chapter)	0.949 ± 0.024	0.949 ± 0.022	0.949 ± 0.021
USS [5, 73]	0.729 ± 0.041	0.718 ± 0.039	0.723 ± 0.036

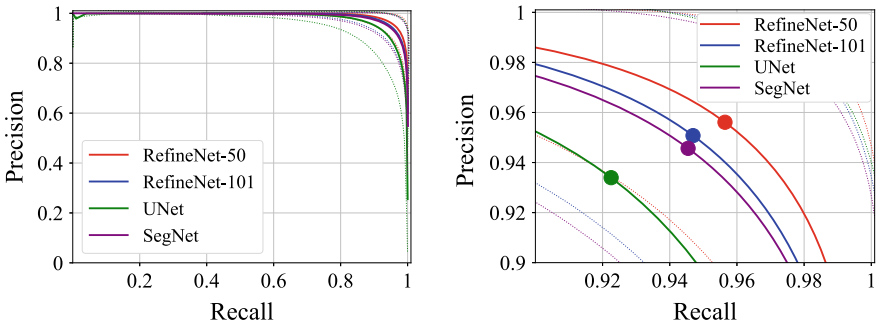


Fig. 13.7 Precision–recall curves for the tested CNN models. USS is not included here, as it returns only binary masks of the sclera region. The left graph shows the complete plot generated by varying the segmentation threshold Δ over all possible values, whereas the right graph shows a zoomed in region to highlight the minute differences between the techniques. The marked points stand for the operating points with the highest F1-Score. The dotted lines show the dispersion (σ) of the precision and recall scores over the test images

images, the second shows the manually annotated ground truth and the remaining columns show results generated by (from left to right): USS, RefineNet-50, RefineNet-101, SegNet and UNet. These results again confirm that all CNN-based models ensure similar segmentation performance. All models segment the sclera region well and differ only in some finer details, such as eyelashes, which are not really important for the second segmentation step, where the vasculature needs to be extracted from the ocular images.

Consequently, any of the tested CNN-based segmentation models could be used in our sclera recognition pipeline for the initial segmentation step, but we favour

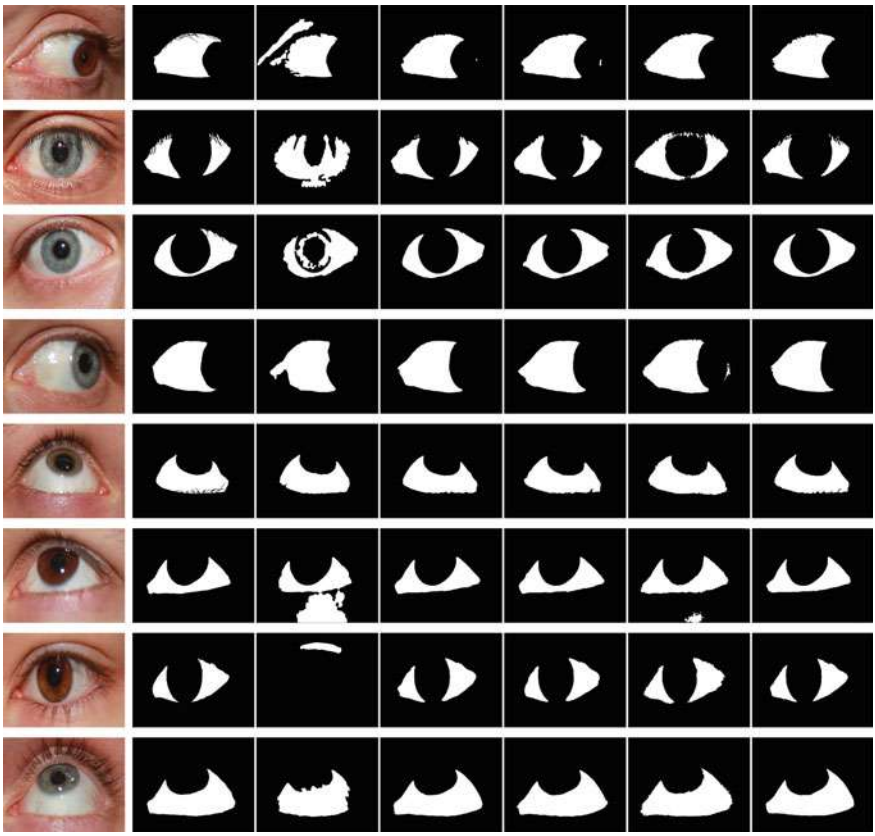


Fig. 13.8 Visual examples of the segmentation results produced by the tested segmentation models. The first column shows the input RGB ocular images, the second the manually annotated ground truth and the remaining columns show the results generated by (from left to right): USS, RefineNet-50, RefineNet-101, SegNet and UNet. Note that the CNN models (last four columns) produce visually similar segmentation results and differ only in certain fine details

SegNet because of the fast prediction time, which is 4 times faster the second fastest CNN model, i.e. RefineNet-50.

13.5.4 Evaluation of Vasculature Segmentation Models

In the next series of experiments, we evaluate the performance of the second segmentation step of our pipeline, which aims to locate and segment the vascular structure of the sclera from the input image. The input to this step is again an RGB ocular image (see Fig. 13.9), but masked with the segmentation output produced by the SegNet model evaluated in the previous section.

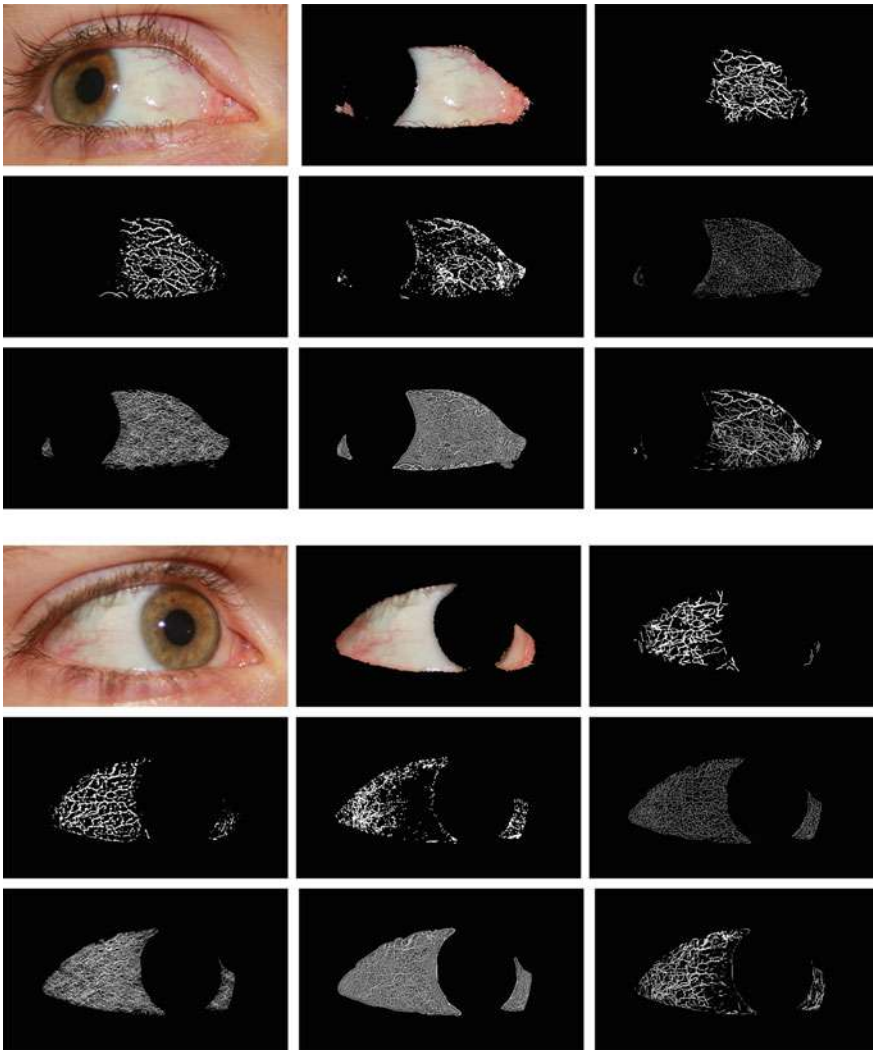


Fig. 13.9 Examples of vasculature segmentation results. Each of the two image blocks shows (from left to right and top to bottom): the input RGB ocular image, the input image masked with the sclera region produced by the initial segmentation step, the ground truth markup, results for the proposed cascaded SegNet assembly, and results for the Adaptive Gaussian Thresholding (AGT), and the NMC, NRLT, Coye and B-COSFIRE approaches. The results show the generated binary masks corresponding to the operating point used in Table 13.5. Note that the proposed approach most convincingly captures the characteristics of the manual vasculature markup. Best viewed electronically and zoomed in

As emphasised earlier, we conduct segmentation with our approach in a patch-wise manner to ensure that information about the finer details of the sclera vasculature is not lost. Because the second SegNet model of the cascaded assembly outputs probability maps, we use adaptive Gaussian thresholding [74] to generate binary masks to compare with the manually annotated ground truth. To assess performance, we compute results over the binary masks and again report fixed precision, recall and F1-score values in this series of experiments. The performance scores are computed for the operating point on the precision–recall curve that corresponds to the maximum possible F1-score. We again report standard deviations in addition to the average scores to have a measure of dispersion for the results of the test data.

For comparison purposes, we implement a number of competing techniques from the literature that are regularly used for vessel segmentation in the field of vascular biometrics, i.e. (i) Adaptive Gaussian Thresholding (AGT) [74], (ii) Normalized Maximum Curvature (NMC) [75], (iii) Normalized Repeated Line Tracking (NRLT) [76], (iv)) Coye filtering [77] and (v) the B-COSFIRE approach from [78, 79]. The NMC and NRLT approaches represent a modified version of the original segmentation techniques and are normalised to return continuous probability maps rather than binarized segmentation results. The hyper-parameters of all baseline techniques (if any) are selected to maximise performance. The techniques are implemented using publicly available source code.[5] We note again that no supervised approach to sclera vasculature segmentation has been presented in the literature so far. We focus, therefore, exclusively on unsupervised segmentation techniques in our comparative assessment.

The results of the experiments are presented in Table 13.5. As can be seen, SegNet ensures the best overall results by a large margin with an average F1-score of 0.727. The B-COSFIRE techniques, regularly used for vasculature segmentation in retina images, is the runner-up with an average F1-score of 0.393, followed closely by AGT thresholding with an F1-score of 0.306. The NMC, NRLT and Coye filter approaches result in worse performance with F1-scores below 0.25. While the performance difference between the SegNet model and the competing techniques is considerable, it is also expected, as SegNet is trained on the manually annotated vasculature, while the remaining approaches rely only on local image characteristics to identify the vascular structure of the sclera. As a result, the vasculature extracted by the unsupervised techniques (NMC, NRLT, Coye filter and B-COSFIRE) does not necessarily correspond to the markup generated by a human annotator. However, the low-performance scores of the unsupervised techniques do not indicate that the extracted vasculature is useless for recognition, but only that there is low correspondence with the manual markup. To investigate the usefulness of the extracted vascular patterns of these

[5]Code for the techniques is available from: AGT from OpenCV: https://opencv.org/, NMC and NRLT from Mathworks: https://www.mathworks.com/matlabcentral/fileexchange/35716-miura-et-al-vein-extraction-methods
Coye filter from Mathworks: https://www.mathworks.com/matlabcentral/fileexchange/50839-novel-retinal-vessel-segmentation-algorithm-fundus-images
B-COSFIRE from Mathworks: https://www.mathworks.com/matlabcentral/fileexchange/49172-trainable-cosfire-filters-for-curvilinear-structure-delineation-in-images.

Table 13.5 Comparison of vasculature segmentation techniques. Results are presented for the proposed cascaded SegNet assembly, as well as for five competing unsupervised segmentation approaches from the literature. The probability maps generated by the techniques have been thresholded to allow for comparisons with the annotated binary vasculature markup. Note that the proposed approach achieves the best overall performance by a large margin

Algorithm	Precision	Recall	F1-score
SegNet [15] + AGT (ours, this chapter)	0.806 ± 0.155	0.675 ± 0.131	0.727 ± 0.120
Adaptive Gaussian thresholding (AGT)	0.308 ± 0.119	0.372 ± 0.201	0.306 ± 0.120
Normalized maximum curvature (NMC)	0.240 ± 0.097	0.247 ± 0.044	0.232 ± 0.062
Normalized repeated line tracking (NRLT)	0.145 ± 0.055	0.314 ± 0.114	0.191 ± 0.066
Coye filter	0.143 ± 0.070	0.376 ± 0.085	0.198 ± 0.078
B-COSFIRE	0.351 ± 0.142	0.480 ± 0.083	0.393 ± 0.116

techniques for recognition, we conduct a series of recognition experiments in the next section.

To put the reported results into perspective and show what the scores mean visually, we present in Fig. 13.9 some qualitative segmentation results. Here, each of the two image blocks shows (from left to right and top to bottom): the input ocular image, the masked sclera region, the ground truth annotation and results for the proposed cascaded SegNet assembly, the Adaptive Gaussian Thresholding (AGT), and the NMC, NRLT, Coye and B-COSFIRE techniques. It is interesting to see what level of detail the SegNet-based model is able to recover from the input image. Despite the relatively poor contrast of some of the finer veins, the model still successfully segments the sclera vasculature from the input images. The B-COSFIRE results are also convincing when examined visually, but as emphasised earlier do not result in high-performance scores when compared to the manual markup. Other competing models are less successful and generate less precise segmentation results. However, as suggested above, the competing models use no supervision to learn to segment the vascular structures and therefore generate segmentation results that do not correspond well to the manual markup.

To further highlight the quality of the segmentation ensured by the SegNet-based model, we show a close up of the vascular structure of an eye and the corresponding segmentation output in Fig. 13.10. We see that the model successfully segments most of the vascular structure, but also picks up on the eyelashes, which very much resemble the vein patterns of the sclera even from a human perspective. In the area where reflections are visible, the model is not able to recover the vascular structure from the input image. Furthermore, despite the patch-wise processing used with the cascaded SegNet segmentation approach, we observe no visible artifacts caused

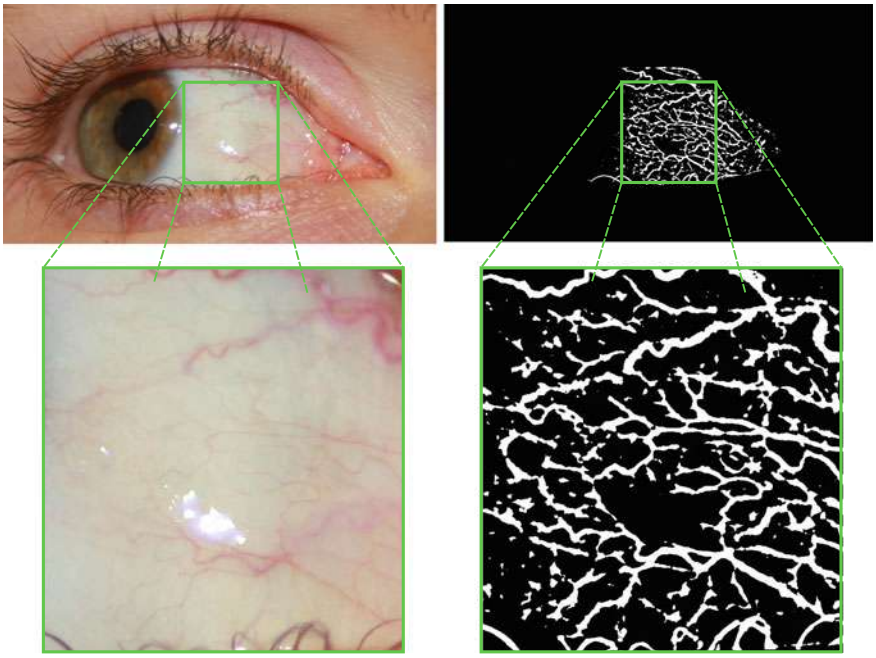


Fig. 13.10 Visualisation of the fine vascular structure recovered by our segmentation model. The image shows a zoomed in region of the vascular structure of the eye (on the left) and the corresponding binarized output of our model (on the right)

by the re-assembly procedure. We assume this is a consequence of the run-time augmentation step that smooths out such artifacts.

Because the segmentation is performed in a patch-wise manner, the average time needed to process one input image with the proposed model in this part is 5.6 seconds when using a single GPU (please note that this step can be parallelised using multiple GPUs, because patch predictions can be calculated independently). For comparison, the average processing time for AGT is 1.2 s, for NMC it is 32.5 s, for NRLT the processing time is 7.9 s, for Coye it is 1.2 s and for the B-COSFIRE the processing time is 13.9 s. However, note that different programming languages were used for the implementation of the segmentation methods, so the processing times need to be interpreted accordingly. For the proposed cascaded SegNet assembly, the entire region-of-interest extraction step (which comprises the initial sclera segmentation and vascular structure segmentation steps), takes around 6 s using a single GPU for one input image on average.

Overall, these results suggest that the trained segmentation model is able to produce good quality segmentation results that can be used for recognition purposes. We evaluate the performance of our recognition approach with the generated segmentation outputs next.

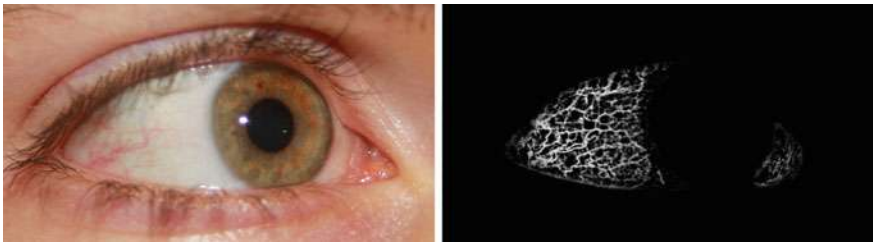


Fig. 13.11 Example of an input image and the corresponding probability map generated by the SegNet model. The probability mask on the left is used as input to the ScleraNET model

13.5.5 Recognition Experiments

In the last series of experiments, we assess the performance of the entire recognition pipeline and feed the segmented sclera vasculature into our ScleraNET model for feature extraction. Note again that we use the probability output of the segmentation models as input to ScleraNET (marked **y** in Fig. 13.2) and not the generated binary masks of the vasculature. An example of the probability map generated with the SegNet model is shown in Fig. 13.11. Once a feature representation is computed from the input image, it is used with the cosine similarity to compute similarity scores and to ultimately conduct identity inference. The feature computation procedure takes 0.1 s per image on average.

To evaluate the recognition performance of ScleraNET, we conduct verification experiments using the following experimental setup:

- We first generate user templates by randomly selecting four images of each subject in the test set. We sample the test set in a way that ensures that each template contains all four gaze directions (i.e. up, down, left and right). Since each subject has at least 4 images of each gaze direction, we are able to generate multiple templates for each subject in the test set.
- Next, we use all images in the test set and compare them to the generated user templates. The comparison is conducted by comparing (using the cosine similarity) the query vasculature descriptor to the descriptors of each image in the template. The highest similarity score is kept as the score for the query-to-template comparison. If the query image is also present in the template, we exclude the score from the evaluation.
- We repeat the entire process 5-times to estimate average performance scores as well as standard deviations. The outlined setup results in a total of 1228 legitimate and 121572 illegitimate verification attempts in each of the 5 repetitions.

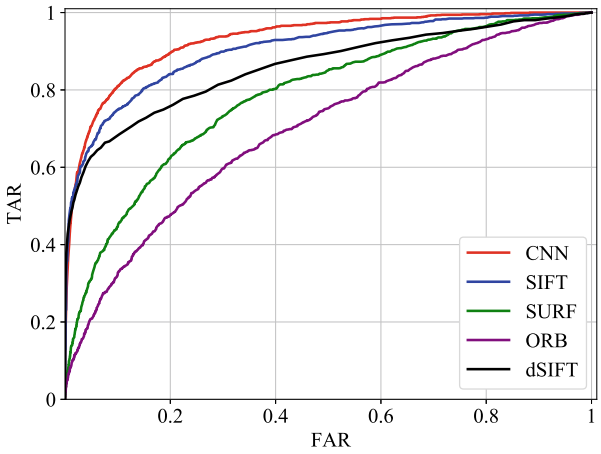
Becausethe ocular images are not aligned, we implement multiple descriptor-based approaches for comparison. Specifically, we implement the dense SIFT (dSIFT hereafter) approach from [8] and several keypoint based techniques. For the latter, we compute SIFT [80], SURF [81] and ORB [82] descriptors using their corresponding keypoint detectors. For each image-pair comparison, we use the average Euclidean

Table 13.6 Results of the recognition experiments. The table shows performance scores for five different descriptor-computation strategies and five approaches to vasculature segmentation. For each performance metric, the best overall result is coloured red and the best results for a given segmentation approach is coloured blue. The proposed ScleraNET model ensures competitive performance significantly outperforming the competing models when applied on the segmentation results generated by the proposed cascaded SegNet assembly

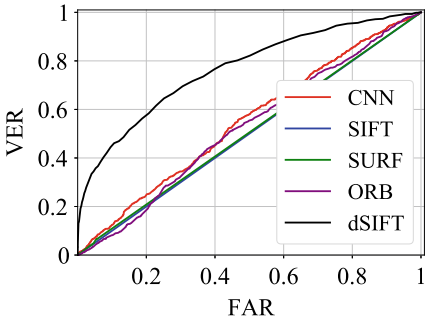
Segment.	Algorithm	VER@0.1FAR	VER@1FAR	EER	AUC
C. SegNet (ours)	ScleraNET (ours)	0.181 ± 0.009	0.459 ± 0.009	0.145 ± 0.002	0.933 ± 0.002
	SIFT	0.184 ± 0.076	0.452 ± 0.040	0.176 ± 0.005	0.903 ± 0.005
	SURF	0.023 ± 0.007	0.126 ± 0.010	0.286 ± 0.004	0.782 ± 0.005
	ORB	0.017 ± 0.004	0.080 ± 0.011	0.351 ± 0.003	0.704 ± 0.005
	Dense SIFT	0.326 ± 0.016	0.507 ± 0.010	0.221 ± 0.004	0.865 ± 0.002
NMC	ScleraNET	0.002 ± 0.001	0.023 ± 0.003	0.425 ± 0.004	0.596 ± 0.004
	SIFT	0.000 ± 0.000	0.000 ± 0.000	0.500 ± 0.000	0.500 ± 0.000
	SURF	0.017 ± 0.024	0.031 ± 0.016	0.488 ± 0.013	0.535 ± 0.010
	ORB	0.000 ± 0.000	0.005 ± 0.005	0.504 ± 0.006	0.497 ± 0.006
	Dense SIFT	0.063 ± 0.014	0.184 ± 0.028	0.371 ± 0.012	0.683 ± 0.010
NRLT	ScleraNET	0.112 ± 0.011	0.311 ± 0.006	0.196 ± 0.008	0.888 ± 0.004
	SIFT	0.000 ± 0.000	0.014 ± 0.005	0.500 ± 0.001	0.500 ± 0.002
	SURF	0.000 ± 0.000	0.021 ± 0.013	0.492 ± 0.008	0.509 ± 0.005
	ORB	0.000 ± 0.000	0.021 ± 0.010	0.502 ± 0.005	0.499 ± 0.007
	Dense SIFT	0.047 ± 0.004	0.153 ± 0.010	0.362 ± 0.008	0.701 ± 0.004
Coye	ScleraNET	0.067 ± 0.008	0.215 ± 0.007	0.267 ± 0.006	0.812 ± 0.004
	SIFT	0.000 ± 0.000	0.036 ± 0.014	0.496 ± 0.001	0.507 ± 0.002
	SURF	0.000 ± 0.000	0.000 ± 0.000	0.500 ± 0.005	0.497 ± 0.005
	ORB	0.002 ± 0.001	0.023 ± 0.005	0.451 ± 0.005	0.568 ± 0.006
	Dense SIFT	0.091 ± 0.005	0.234 ± 0.018	0.300 ± 0.004	0.772 ± 0.004
B-COSFIRE	ScleraNET	0.042 ± 0.004	0.140 ± 0.008	0.337 ± 0.005	0.723 ± 0.006
	SIFT	0.000 ± 0.000	0.012 ± 0.005	0.488 ± 0.002	0.522 ± 0.003
	SURF	0.000 ± 0.000	0.000 ± 0.000	0.494 ± 0.005	0.513 ± 0.003
	ORB	0.000 ± 0.000	0.008 ± 0.002	0.467 ± 0.003	0.539 ± 0.004
	Dense SIFT	0.110 ± 0.011	0.242 ± 0.011	0.325 ± 0.006	0.748 ± 0.005

distance between matching descriptors as the similarity score for recognition. Since the descriptor-based approaches are local and rely on keypoint correspondences, they are particularly suitable for problems such as sclera recognition, where (partially visible) unaligned vascular structures under different views need to be matched against each other. We conduct experiments with the vasculature extracted with the proposed cascaded SegNet assembly, so we are able to evaluate our complete processing pipeline, but also with the segmentation results produced by the competing segmentation approaches evaluated in the previous section, i.e. NMC, NRLT, Coye and B-COSFIRE.

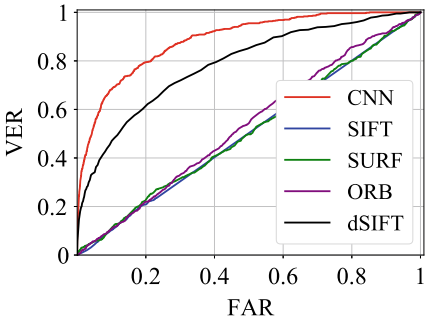
From the results in Table 13.6 and Fig. 13.12 (results for ScleraNET in the figures are marked as CNN), we see that the proposed pipeline (cascaded SegNet assembly +



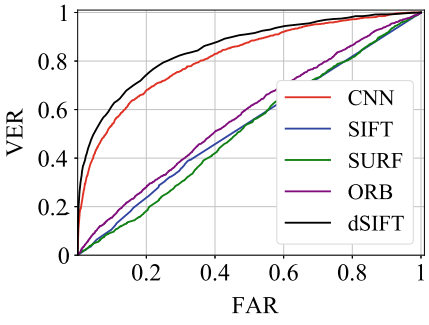
(a) Recognition results based on vasculature extracted with the SegNet assembly.



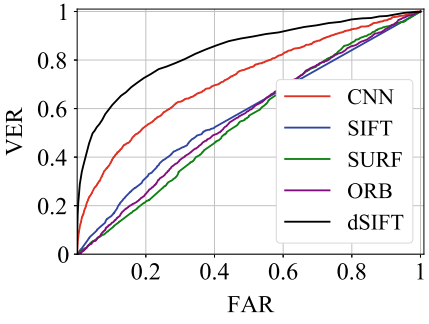
(b) NMC-based ROC curves.



(c) NRLT-based ROC curves.



(d) Coye-filter-based ROC curves.



(e) B-COSFIRE-based ROC curves.

Fig. 13.12 Results of the verification experiments. The graphs show recognition results for several feature extraction techniques and multiple approaches to vasculature segmentation. The pipeline proposed in this chapter results in the best overall performance

ScleraNET) ensures an average AUC of 0.933 for the verification experiments compared to the average AUC of 0.903 for the runner-up, the SIFT-based approach. Interestingly, the dSIFT approach is very competitive at the lower FAR values, but becomes less competitive at the higher values of FAR—see Fig. 13.12a. This behaviour can likely be ascribed to the dense nature of the descriptor, which makes it difficult to reliably compare images when there is scale and position variability present in the samples. The remaining three descriptors, SIFT, SURF and ORB, are less competitive and result in lower performance scores.

The segmentation results generated by the proposed cascaded SegNet assembly appear to be the most suitable for recognition purposes, as can be seen by comparing the ROC curves from Fig. 13.12b–e, to the results in Fig. 13.12a, or examining the lower part of Table 13.6. While the NMC, NRLT, Coye and B-COSFIRE segmentation results (in the form of probability maps) result in above-random verification performance with the ScleraNET and dSIFT descriptors, the performance is at chance for the keypoint-descriptor-based methods—SIFT, SURF and ORB. The reason for this is the difficulty of finding matching descriptors in the images, which leads to poor performance. The ScleraNET model, on the other hand, seems to generalise reasonably well to segmentation outputs with characteristics different from those produced by the cascaded SegNet assembly. It achieves the best performance with the NRLT and Coye segmentation techniques, it is comparable in performance to dSIFT on B-COSFIRE segmented vasculature and is second only to dSIFT with the NMC approach. This is surprising, as it was not trained on vascular images produced by these methods. Nonetheless, it seems to be able to extract useful descriptors for recognition from these images as well.

Overall, the results achieved with the proposed pipeline are very encouraging and present a good foundation for further research, also in the context of multi-modal biometric systems built around (peri-)ocular information.

13.6 Conclusion

We have presented a novel approach to sclera recognition built around convolutional neural networks. Our approach uses a two-step procedure that first locates the vascular structure of the sclera from the input image and then extracts a discriminative representation from the segmented vasculature that can be used for image comparisons and ultimately recognition. The two-step segmentation procedure is based on cascaded SegNet assembly, the first supervised approach to sclera vasculature segmentation presented in the literature, while the descriptor-computation procedure is based on a novel CNN-based model, called ScleraNET, trained in a multi-task manner. We evaluated our approach on a newly introduced and publicly available dataset of annotated sclera images and presented encouraging comparative results with competing methods. As part of our future work, we plan to integrate the presented pipeline with other ocular traits into a multi-modal recognition system.

Acknowledgements This research was supported in parts by ARRS (Slovenian Research Agency) Research Program P2-0250 (B) Metrology and Biometric Systems, ARRS Research Program P2-0214 (A) Computer Vision, and the RS-MIZŠ and EU-ESRR funded GOSTOP. The GPU used for this research was donated by the NVIDIA Corporation.

References

1. Alkassar S, Woo WL, Dlay SS, Chambers JA (2017) Robust sclera recognition system with novel sclera segmentation and validation techniques. IEEE Trans Syst Man Cybern Syst (TSMC) 47(3):474–486
2. Ali Z, Park U, Nang J, Park JS, Hong T, Park S (2017) Periocular recognition using uMLBP and attribute features. KSII Trans Internet Inf Syst (TIIS) 11(12):6133–6151
3. Jain AK, Ross A, Prabhakar S et al (2004) An introduction to biometric recognition. IEEE Trans Circuits Syst Video Technol 14(1)
4. Patil V, Patil AM (2017) Human identification method: sclera recognition. Int J Comput Sci Netw (IJCSN) 6(1):24–29
5. Das A, Pal U, Ferrer MA, Blumenstein M, Štepec D, Rot P, Emeršič Ž, Peer P, Štruc V, Kumar SA et al (2017) SSERBC 2017: sclera segmentation and eye recognition benchmarking competition. In: International joint conference on biometrics (IJCB), pp 742–747
6. Derakhshani R, Ross A (2007) A texture-based neural network classifier for biometric identification using ocular surface vasculature. In: International joint conference on neural networks 2007 (IJCNN 2007), pp 2982–2987. IEEE
7. Zhou Z, Du EY, Thomas NL, Delp EJ (2012) A new human identification method: sclera recognition. IEEE Trans Syst Man Cybern (TSMC) Part A Syst Hum 42(3):571–583
8. Das A, Pal U, Ferrer Ballester MA, Blumenstein M (2013) Sclera recognition using dense-sift. In: 2013 13th international conference on intelligent systems design and applications (ISDA), pp 74–79. IEEE
9. Zhou Z, Du EY, Thomas NL, Delp EJ (2011) Multi-angle sclera recognition system. In: 2011 IEEE workshop on computational intelligence in biometrics and identity management (CIBIM), pp 103–108. IEEE
10. Das A, Pal U, Ferrer MA, Blumenstein M, Štepec D, Rot P, Peer P, Štruc V (2018) SSBC 2018: sclera segmentation benchmarking competition. In: International conference on biometrics (ICB), pp 303–308
11. Emeršič Ž, Štepec D, Štruc V, Peer P (2017) Training convolutional neural networks with limited training data for ear recognition in the wild. arXiv:1711.09952
12. Emeršič Ž, Meden B, Peer P, Štruc V (2018) Evaluation and analysis of ear recognition models: performance, complexity and resource requirements. Neural computing and applications, pp 1–16
13. Emeršič Ž, Štepec D, Štruc V, Peer P, George A, Ahmad A, Omar E, Boult TE, Safdaii R, Zhou Y et al (2017) The unconstrained ear recognition challenge. In: 2017 IEEE international joint conference on biometrics (IJCB), pp 715–724. IEEE
14. Grm K, Štruc V, Artiges A, Caron M, Ekenel HK (2017) Strengths and weaknesses of deep learning models for face recognition against image degradations. IET Biom 7(1):81–89
15. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell (TPAMI) 39(12):2481–2495
16. Nigam I, Vatsa M, Singh R (2015) Ocular biometrics: a survey of modalities and fusion approaches. Inf Fusion 26:1–35
17. De Marsico M, Petrosino A, Ricciardi S (2016) Iris recognition through machine learning techniques: a survey. PRL 82:106–115

18. Nguyen K, Fookes C, Jillela R, Sridharan S, Ross A (2017) Long range iris recognition: a survey. PR 72:123–143
19. Daugman J (1993) High confidence visual recognition of persons by a test of statistical independence. IEEE Trans Pattern Anal Mach Intell (TPAMI) 15(11):1148–1161
20. Daugman J (2009) How iris recognition works. In: The essential guide to image processing, pp 715–739. Elsevier
21. Daugman J (2007) New methods in iris recognition. IEEE Trans Syst Man Cybern (TSMC) Part B 37(5):1167–1175
22. Monro DM, Rakshit S, Zhang D (2007) DCT-based iris recognition. IEEE Trans Pattern Anal Mach Intell (TPAMI) 29(4):586–595
23. Miyazawa K, Ito K, Aoki T, Kobayashi K, Nakajima H (2008) An effective approach for iris recognition using phase-based image matching. IEEE Trans Pattern Anal Mach Intell (TPAMI) 30(10):1741–1756
24. Sun Z, Tan T (2009) Ordinal measures for iris recognition. IEEE Trans Pattern Anal Mach Intell (TPAMI) 31(12):2211–2226
25. Sun Z, Zhang H, Tan T, Wang J (2014) Iris image classification based on hierarchical visual codebook. IEEE Trans Pattern Anal Mach Intell (TPAMI) 36(6):1120–1133
26. Gangwar A, Joshi A (2016) Deepirisnet: deep iris representation with applications in iris recognition and cross-sensor iris recognition. In: 2016 IEEE international conference on image processing (ICIP), pp 2301–2305, Sept 2016
27. Liu N, Zhang M, Li H, Sun Z, Tan T (2016) Deepiris: learning pairwise filter bank for heterogeneous iris verification. Pattern Recogn Lett 82:154–161
28. Xingqiang T, Jiangtao X, Peihu L (2017) Deep convolutional features for iris recognition. In: Proceedings of the 12th Chinese conference biometric recognition CCBR 2017, pp 391–400, Cham, 2017. Springer International Publishing
29. Zhao Z, Kumar A (2017) Towards more accurate iris recognition using deeply learned spatially corresponding features. In: International conference on computer vision, ICCV 2017, pp 1–10
30. Nguyen K, Fookes C, Ross A, Sridharan S (2018) Iris recognition with off-the-shelf CNN features: a deep learning perspective. IEEE Access 6:18848–18855
31. Park U, Ross A, Jain AK (2009) Periocular biometrics in the visible spectrum: a feasibility study. In: 2009 IEEE 3rd international conference on biometrics: theory, applications, and systems, pp 1–6. IEEE
32. Uzair M, Mahmood A, Mian A, McDonald C (2013) Periocular biometric recognition using image sets. In: IEEE winter conference on applications of computer vision (WACV), pp 246–251
33. Sequeira A, Chen L, Ferryman J, Wild P, Alonso-Fernandez F, Bigun J, Raja K, Raghavendra R, Busch C, Freitas Pereira T et al (2017) Cross-eyed 2017: cross-spectral iris/periocular recognition database and competition. In: IEEE international joint conference on biometrics (IJCB)
34. Proença H, Neves J (2018) Deep-prwis: periocular recognition without the iris and sclera using deep learning frameworks. IEEE Trans Inf Forensics Secur (TIFS) 13(4):888–896
35. Zhao Z, Kumar A (2017) Accurate periocular recognition under less constrained environment using semantics-assisted convolutional neural network. IEEE Trans Inf Forensics Secur (TIFS) 12(5):1017–1030
36. Alonso-Fernandez F, Bigun J (2015) Near-infrared and visible-light periocular recognition with gabor features using frequency-adaptive automatic eye detection. IET Biom 4(2):74–89
37. Das A, Mondal P, Pal U, Blumenstein M, Ferrer M (2017) Sclera vessel pattern synthesis based on a non-parametric texture synthesis technique. In: Proceedings of computer vision and image processing (CVIP), pp 241–250. Springer
38. Alkassar S, Woo WL, Dlay S, Chambers J (2016) Sclera recognition: on the quality measure and segmentation of degraded images captured under relaxed imaging conditions. IET Biom 6(4):266–275
39. Yadav D, Kohli N, Doyle J, Singh R, Vatsa M, Bowyer K (2014) Unraveling the effect of textured contact lenses on iris recognition. IEEE Trans Inf Forensics Secur (TIFS) 9(5):851–862

40. Raja KB, Raghavendra R, Vemuri VK, Busch C (2015) Smartphone based visible iris recognition using deep sparse filtering. PRL 57:33–42
41. Das A, Pal U, Ferrer MA, Blumenstein M (2015) SSBC 2015: sclera segmentation benchmarking competition. In: International conference on biometrics: theory, applications, and systems (BTAS), pp 1–6
42. Das A, Pal U, Ferrer MA, Blumenstein M (2013) SSRBC 2016: sclera segmentation and recognition benchmarking competition. In: International conference on biometrics (ICB), pp 1–6
43. Maxwell EG, Tripti C (2013) A comparison between contrast limited adaptive histogram equalization and gabor filter sclera blood vessel enhancement techniques. Int J Soft Comput Eng (IJSCE) 3(4):22–5
44. Tankasala SP, Doynov P, Derakhshani RR, Ross A, Crihalmeanu S (2011) Biometric recognition of conjunctival vasculature using GLCM features. In: 2011 international conference on image information processing (ICIIP), pp 1–6, Nov 2011
45. Das A, Pal U, Ferrer Ballester MA, Blumenstein M (2014) Fuzzy logic based selera recognition. In: 2014 IEEE international conference on fuzzy systems, pp 561–568
46. Das A, Pal U, Ferrer Ballester MA, Blumenstein M (2013) A new method for sclera vessel recognition using OLBP. In: Sun Z, Shan S, Yang G, Zhou J, Wang Y, Yin Y (eds) Biometric Recognition. Springer International Publishing, Cham, pp 370–377
47. CASIA Iris Image Database. http://biometrics.idealtest.org/
48. Proença H, Alexandre L (2005) Ubiris: a noisy iris image database. In: International conference on image analysis and processing (ICIAP), pp 970–977. Springer
49. Bowyer KW, Flynn PJ (2016) The ND-IRIS-0405 iris image dataset. Technical report, Notre Dame University
50. Hosseini MS, Araabi BN, Soltanian-Zadeh H (2010) Pigment melanin: pattern for iris recognition. IEEE Trans Instrum Meas 59(4):792–804
51. Kumar A, Passi A (2010) Comparison and combination of iris matchers for reliable personal authentication. Pattern Recogn 43(3):1016–1026
52. Proença H, Filipe S, Santos R, Oliveira J, Alexandre LA (2010) The UBIRIS.v2: a database of visible wavelength iris images captured on-the-move and at-a-distance. IEEE Trans Pattern Anal Mach Intell (TPAMI) 32(8):1529–1535
53. De Marsico M, Nappi M, Riccio D, Wechsler H (2015) Mobile iris challenge evaluation (MICHE)-I, biometric iris dataset and protocols. Pattern Recogn Letters 57:17–23
54. Padole CN, Proenca H (2012) Periocular recognition: analysis of performance degradation factors. In: 5th IAPR international conference on biometrics (ICB), pp 439–445
55. Sharma A, Verma S, Vatsa M, Singh R (2014) On cross spectral periocular recognition. In: IEEE international conference on image processing (ICIP), pp 5007–5011. IEEE
56. Hosseini MS, Araabi BN, Soltanian-Zadeh H (2010) Pigment melanin: pattern for iris recognition. IEEE Trans Instrum Meas (TIM) 59(4):792–804
57. Rot P, Štruc V, Peer P (2018) Deep multi-class eye segmentation for ocular biometrics. In: IEEE international work conference on bioinspired intelligence (IWOBI)
58. Emeršič Ž, Gabriel L, Štruc V, Peer P (2018) Convolutional encoder-decoder networks for pixel-wise ear detection and segmentation. IET Biom 7(3):175–184
59. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition
60. Kendall A, Gal Y, Cipolla R (2018) Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7482–7491
61. Krizhevsky A, Sutskever I, Hinton G (2012) Imagenet classification with deep convolutional neural networks. In: Conference on neural information processing systems (NIPS), pp 1097–1105
62. Yin X, Liu X (2017) Multi-task convolutional neural network for pose-invariant face recognition. IEEE Trans Image Process (TIP)

63. Lozej J, Meden B, Štruc V, Peer P (2018) End-to-end iris segmentation using u-net. In: IEEE international work conference on bioinspired intelligence (IWOBI)
64. Križaj J, Štruc V, Dobrišek S (2013) Combining 3d face representations using region covariance descriptors and statistical models. In: 2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG), pp 1–7. IEEE
65. Štruc V (2012) The PhD face recognition toolbox: toolbox description and user manual. Faculty of Electrical Engineering Ljubljana
66. Vesnicer B, Žganec Gros J, Pavešić N, Štruc V (2012) Face recognition using simplified probabilistic linear discriminant analysis. Int J Adv Robot Syst (IJARS) 9(5):180
67. Glorot X, Bengio Y (2012) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics (AISTATS), pp 249–256
68. Štruc V, Pavešić N (2009) Phase congruency features for palm-print verification. IET Signal Process 3(4):258–268
69. Savič T, Pavešić N (2007) Personal recognition based on an image of the palmar surface of the hand. Pattern Recogn 40(11):3152–3163
70. Lin G, Milan A, Shen C, Reid I (2017) Refinenet: multi-path refinement networks for high-resolution semantic segmentation. In: IEEE conference on computer vision and pattern recognition (CVPR)
71. Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 6450–6458
72. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: 18th international conference, proceedings, part III, medical image computing and computer-assisted intervention (MICCAI), pp 234–241. Springer International Publishing, Cham
73. Riccio D, Brancati N, Frucci M, Gragnaniello D (2017) An unsupervised approach for eye sclera segmentation. In: Iberoamerican congress on pattern recognition (CIARP), pp 550–557. Springer
74. Gonzalez RC, Woods RE (2007) Image processing. Digit Image Process 2
75. Miura N, Nagasaka A, Miyatake T (2007) Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans Inf Syst 90(8):1185–1194
76. Miura N, Nagasaka A, Miyatake T (2004) Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach Vis Appl 15(4):194–203
77. Coye T (2015) A novel retinal blood vessel segmentation algorithm for fundus images. http://www.mathworks.com/matlabcentral/fileexchange/50839. matlab central file exchange. Accessed 4 Mar 2019
78. Azzopardi George, Strisciuglio Nicola, Vento Mario, Petkov Nicolai (2015) Trainable cosfire filters for vessel delineation with application to retinal images. Med Image Anal 19(1):46–57
79. Strisciuglio Nicola, Azzopardi George, Vento Mario, Petkov Nicolai (2016) Supervised vessel delineation in retinal fundus images with the automatic selection of B-COSFIRE filters. Mach Vis Appl 27(8):1137–1149
80. Lowe DG (1999) Object recognition from local scale-invariant features. Int J Comput Vis (ICCV) 2:1150–1157. IEEE
81. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: European conference on computer vision, pp 404–417. Springer
82. Rublee E, Rabaud V, Konolige K, Bradski G (2011) ORB: an efficient alternative to sift or surf

Part IV
Security and Privacy in Vascular Biometrics

Chapter 14
Presentation Attack Detection for Finger Recognition

Jascha Kolberg, Marta Gomez-Barrero, Sushma Venkatesh, Raghavendra Ramachandra and Christoph Busch

Abstract Whereas other biometric characteristics, such as the face, are readily available for an eventual attacker through social media or easy to capture with a conventional smartphone, vein patterns can only be acquired with dedicated sensors. This fact makes them relevant not only for recognition purposes but especially for Presentation Attack Detection (PAD), for instance, in combination with fingerprint recognition. In this chapter, we make use of this combination and present a finger vein-based PAD algorithm to detect presentation attacks targeting fingerprint recognition. The experiments are carried out on a newly collected database, comprising 32 species of Presentation Attack Instruments ranging from printed artefacts to more sophisticated fingerprint overlays. The results show that our method preserves a convenient usage while detecting around 90% of the attacks. However, thin and transparent fingerprint overlays remain very challenging.

Keywords Presentation attack detection · Fingerprint recognition

14.1 Introduction

In spite of the many advantages offered by biometric recognition with respect to other traditional authentication methods (the well-known Lema "forget about PINs or

J. Kolberg (✉) · M. Gomez-Barrero · C. Busch
da/sec - Biometrics and Internet Security Research Group, Hochschule Darmstadt, Darmstadt, Germany
e-mail: jascha.kolberg@h-da.de

M. Gomez-Barrero
e-mail: marta.gomez-barrero@h-da.de

C. Busch
e-mail: christoph.busch@h-da.de

S. Venkatesh · R. Ramachandra
Norwegian Information Security Laboratory, Norwegian University of Science and Technology, NTNU, Gjøvik, Norway
e-mail: sushma.venkatesh@ntnu.no

R. Ramachandra
e-mail: raghavendra.ramachandra@ntnu.no

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_14

passwords, you are your own key"), biometric systems are also vulnerable to external attacks. As a consequence, the security and privacy offered by biometric recognition systems can be undermined. Given its serious implications, the vulnerabilities of biometric systems to different types of attacks have been the subject of numerous studies in the last decades for different characteristics, including fingerprint [9, 18, 64], face [1], iris [23, 26, 27], voice [3] or multimodal systems [2, 10, 28].

Among other possible points of attack [64], the biometric capture device is probably the most exposed one: the attacker does not need to know any details about the inner modules of the biometric system in order to attack the sensor. To fool the biometric system, he can present the capture device with a *Presentation Attack Instrument* (PAI), such as a 3D mask [16], a printed finger vein image [76] or a fingerprint overlay [18]. These attacks are known in the literature as *Presentation Attacks* (PA) [38].

In order to prevent such attacks, *Presentation Attack Detection* (PAD) methods have been recently developed to automatically distinguish between bona fide (i.e. real, live or genuine) presentations and access attempts carried out by means of PAIs [49]. Incorporating such countermeasures in biometric systems are crucial, especially in unattended scenarios. Given the importance of increasing the robustness of biometric systems to these attacks, and hence the systems' security, this area of research has attracted a considerable attention within the biometric community in the last decade. In fact, several international projects like the European Tabula Rasa [70] and BEAT [48], or the more recent US Odin research program [55], deal with these security concerns. In addition, the LivDet liveness detection competition series on iris [79] and fingerprint [80] have been running since 2009. In turn, these initiatives have led to a wide number of publications on PAD methodologies for several biometric characteristics, including iris [19], fingerprint [47, 67], or face [20].

Compared to other biometric characteristics, such as fingerprint or handwritten signature, the use of finger vein for recognition purposes are relatively new: the first commercial applications date back to 2005 by Hitachi Ltd [45]. The first studies on the vulnerability of finger vein recognition systems to presentation attacks were carried out only in 2014 [76]. In this work, Tome et al. showed how a simple print out of a finger vein image could successfully fool the system in up to 86% of the attempts. A similar evaluation was carried out by Tome and Marcel [74] in 2015 for palm vein images, where the success rate of the attacks reached figures as high as 75%. It is hence crucial to protect vein-based systems from these presentation attacks, which, given their simplicity, can be carried out by potentially any individual. This is especially relevant for finger vein, due to the extended use of the corresponding sensors in ATMs (i.e. unsupervised scenario) in countries as diverse as China,[1] Turkey,[2] Taiwan,[3] or Poland.[4]

These facts call for a joint effort within the biometrics community to develop PAD techniques for vein-based systems. In this context, the first approach based on Fourier

[1] https://findbiometrics.com/finger-vein-authentication-atms-china-502087/.

[2] http://www.hitachi.com/New/cnews/120206b.pdf.

[3] http://www.hitachi-omron-ts.com/news/201607-001.html.

[4] http://edition.cnn.com/2010/WORLD/europe/07/05/first.biometric.atm.europe/index.html.

and wavelet transforms was proposed in 2013 by Nguyen et al. [51]. Two years later, the first competition on finger vein PAD was organised [75], where three different teams participated. Since then, different PAD approaches have been presented, based on either a video sequence and motion magnification [60], texture analysis [44, 61, 71], image quality metrics [7], or more recently, neural networks [52, 59, 63] and image decomposition [58].

All the aforementioned works are focused on the detection of printed finger vein images, or, in some cases, of replay attacks carried out with digital displays [61]. In all cases, almost perfect error rates are achieved, thereby indicating that such PAIs can be easily detected with the current techniques. However, the applications of finger vein-based PAD are not limited to finger vein recognition. In fact, the development of multimodal capture devices which are able to acquire both finger vein images or videos, and finger photos, opens new lines of research [62]: biometric recognition can be based on fingerprints extracted from the photos, and PAD techniques can be developed for the finger vein data. This approach is being currently followed in the BATL project [6] within the US Odin research program [55]: among other sensors, finger vein images are used to detect fingerprint presentation attacks. As with the aforementioned finger vein print outs, it has already been shown that fingerprints can be recovered even from the stored ISO templates [18], and then be transformed into a PAI, which is recognised as a fingerprint. However, most fingerprint PAIs do not take into account the blood flow, which is also harder to simulate. On the other hand, the finger vein printed images analysed in the finger vein PAD literature will not be able to fool the fingerprint scanner, as it contains no fingerprint. We can therefore also include a finger vein PAD module in multimodal finger sensors designed for fingerprint recognition, thereby making it harder for an eventual attacker to design a PAI which is able to bypass both sensors.

In this chapter, we will first summarise in Sect. 14.2 the main concepts and evaluation metrics for biometric PAD defined in the recent ISO/IEC 30107 standard [38, 39]. The state of the art in fingervein and fingerprint PAD is subsequently reviewed in Sect. 14.3. We will then describe the multimodal sensor developed in the BATL project and the proposed approach to finger vein-based PAD to detect fingerprint PAIs (Sect. 14.4). The proposed method is evaluated according to the ISO/IEC 30107 standard [39] in Sect. 14.5. The chapter ends with the final discussion and conclusions in Sect. 14.6.

14.2 Presentation Attack Detection

Presentation attacks are defined within the ISO/IEC 30107 standard on biometric presentation attack detection [38] as the "*presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system*". The attacker may aim at impersonating someone else (i.e. impostor) or avoiding being recognised due to black-listing (i.e. identity concealer).

In the following, we include the main definitions presented within the ISO/IEC 30107-3 standard on biometric presentation attack detection—part 3: testing and reporting [39], which will be used throughout the chapter:

- Bona fide presentation: "*interaction of the biometric capture subject and the biometric data capture subsystem in the fashion intended by the policy of the biometric system*". That is, a normal or genuine presentation.
- Attack presentation/presentation attack: "*presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system*". That is, an attack carried out on the capture device to either conceal your identity or impersonate someone else.
- Presentation Attack Instrument (PAI): "*biometric characteristic or object used in a presentation attack*". For instance, a silicone 3D mask or an ecoflex fingerprint overlay.
- PAI species: "*class of presentation attack instruments created using a common production method and based on different biometric characteristics*".

In order to evaluate the vulnerabilities of biometric systems to PAs, the following metrics should be used:

- Impostor Attack Presentation Match Rate (IAPMR): "*proportion of impostor attack presentations using the same PAI species in which the target reference is matched*".
- Attack Presentation Classification Error Rate (APCER): " *proportion of attack presentations using the same PAI species incorrectly classified as bona fide presentations in a specific scenario*".
- Bona Fide Presentation Classification Error Rate (BPCER): " *proportion of bona fide presentations incorrectly classified as presentation attacks in a specific scenario*".

Derived from the aforementioned metrics, a global measure can be computed for an easier benchmark across different systems: the Detection Equal Error Rate (D-EER). It is defined as the error rate at the operating point where APCER = BPCER.

14.3 Related Works

In addition to the initial review of the existing works on finger vein PAD presented in the introductory chapter, we first survey those works in detail, further discussing the PAI species analysed and the detection performance achieved (see Sect. 14.3.1). We subsequently summarise in Sect. 14.3.2 the most relevant works on fingerprint PAD, since our main aim is to detect fingerprint PAIs with finger vein images. For more details and a more extensive survey on fingerprint PAD, the reader is referred to [47, 67].

14.3.1 Finger Vein Presentation Attack Detection

A summary of the most relevant works in finger vein PAD is presented in Table 14.1, classified according to the feature types extracted (handcrafted versus deep learning) and the publication year. In addition, the main performance metrics over the selected database is reported.

As mentioned in Sect. 14.1, research on finger vein recognition is relatively new. As a direct consequence, the pioneering work on finger vein PAD was published as recent as in 2013 [51]. Nguyen et al. proposed the combination of features in both spatial and frequency domains through the Fourier and two different wavelet transforms (i.e. Haar and Daubechies). They achieved a D-EER as low as 1.5% in their experiments on a self-acquired database comprising both bona fides and a single PAI species: printed finger vein images.

One year later, in 2014, Tome et al. analysed in-depth the vulnerabilities of finger vein recognition systems to PAs, revealing an alarming IAPMR up to 86% for simple print outs of vein images [76]. This study motivated Tome et al. to organise the first competition on finger vein PAD in 2015 [75]. In addition to the baseline system developed at Idiap,[5] three teams participated, proposing different approaches to detect the PAs, namely: (i) Binarised Statistical Image Features (BSIF), (ii) a monogenic global descriptor to capture local energy and local orientation at coarse level and (iii) a set of local descriptors including Local Binary Patterns (LBP), Local Phase Quantisation (LPQ), a patch-wise Short-time Fourier transform (STFT) and a Weber Local Descriptor (WLD). In all cases, the final classification was carried out with Support Vector Machines (SVMs), achieving remarkable detection rates with a low complexity. Another byproduct of the competition was the establishment of the Idiap Research Institute VERA Fingervein Database [77] as a benchmark for finger vein PAD (see Table 14.1) with a single PAI species: printed images. This, in turn, motivated the biometrics community to pursue the development of more efficient PAD techniques.

Also in 2015, Raghavendra et al. [60] analysed short video sequences with the aid of Eulerian video magnification [78]. The goal was to amplify the blood flow and thus detect the printed artefacts. They compared the newly proposed method with reimplementations of the algorithms presented in [75] over a self-acquired database: the ACER was reduced 5 to 23 times, thus proving the soundness of the proposed approach. In the same year, Tirunagari et al. proposed the use of Dynamic Mode Decomposition (DMD), which is a mathematical method developed to extract information from non-linear complex fluid flows [71]. They designed a windowed DMD technique in order to extract micro-texture information from a single image, which is decomposed into its maximum variance at column level, and the corresponding residual or noise image. Using SVMs for classification over the VERA DB, they achieved D-EERs outperforming other texture descriptors.

As for other biometric characteristics, texture patterns have been extensively analysed for finger vein PAD. In addition to the approaches presented in [71, 75],

[5] http://www.idiap.ch/en/scientific-research/biometrics-security-and-privacy.

Table 14.1 Summary of the most relevant methodologies for **finger vein** presentation attack detection. For performance evaluation, the metrics are the ones reported in the articles, where "Acc." stands for detection accuracy

Category	Year	References	Description	Performance	Database
Hand-crafted	2013	[51]	FFT, Haar and Daubechies wavelets + Fusion	D-EER $\geq$ 1.5%	Own DB
		[60]	Video analysis with Eulerian	APCER = 5.20%	Own DB
			Video Magnification (EVM)	BPCER = 2.00%	
		[61]	Steerable pyramids	APCER = 2.4%	Own DB
				BPCER = 0.4%	
	2015	[75]	BSIF	APCER = 0%	VERA (full)
				BPCER = 8.00%	
			Monogenic scale space based texture descriptors	APCER = 0%	
				BPCER = 0%	
			LBP, LPQ, STFT, WLD	APCER = 0%	
				BPCER = 0%	
		[71]	Windowed DMD as micro-texture descriptor	D-EER = 0.08%	
	2016	[44]	LBP Extensions	Acc $\geq$ 95%	
	2017	[7]	Image Quality Assessment + Fusion	Acc $\approx$ 99.8%	
	2018	[58]	Total Variation Decomposition + LBP	APCER = 0%	
				BPCER = 0%	
Deep learning	2017	[59]	FPNet (ad-hoc CNN)	APCER = 0%	
				BPCER = 0%	
		[52]	D-CNN (AlexNet or VGG-16) + PCA + SVM	APCER = 0%	
				BPCER = 0%	
	2018	[63]	D-CNN (AlexNet) + LDA or SVM	APCER = 1.82% / 0%	Own DB
				BPCER = 0%	

Raghavendra and Busch included a new PAI species in a subsequent work [61]: a smartphone display. In this case, they considered the residual high frequency band extracted from steerable pyramids and a SVM, achieving again ACERs around 3%. The following year, Kocher et al. thoroughly analysed different LBP extensions in [44], to finally conclude that the baseline LBP technique performs as good as its "improvements". Finally, in a combined approach, Qiu et al. used total variation decomposition to divide the finger vein sample into its structural and noise components [58]. Using again LBP descriptors and SVMs, they achieved a perfect detection accuracy with APCER = BPCER = 0% over the VERA DB.

Another approach followed for PAD, in general, is based on the use of image quality assessment [21]. This technique was also analysed by Bhogal et al. in [7] for finger vein. In particular, they considered six different measures and their combinations, achieving a detection accuracy over 99%.

Finally, in the last years, Deep Learning (DL) has become a thriving topic [33], allowing computers to learn from experience and understand the world in terms of a hierarchy of simpler units. This way, DL has enabled significant advances in complex domains such as natural language processing [69], computer vision [81], biometric recognition in general, and finger vein PAD in particular. In this context, in 2017, Qiu et al. designed a new Convolutional Neural Network (CNN) for finger vein PAD, which they named FPNet [59]. This network achieved a perfect detection accuracy over the VERA DB. In the same year, Nguyen et al. used two different pre-trained models (i.e. AlexNet [46] and VGG-16 [66]) for the same task. After extracting the features with these nets, Nguyen et al. reduced their dimensionality with Principal Component Analysis (PCA) and used SVMs for final classification. Again, a perfect detection rate over the VERA DB was reported. In a similar fashion, Raghavendra et al. analysed in [63] the use of AlexNet with Linear Discriminant Analysis (LDA) and SVMs for classification purposes, also achieving perfect error rates over a self-acquired database.

14.3.2 Fingerprint Presentation Attack Detection

The excellent performance of the finger vein PAD methods described above has motivated us to also use finger vein images to detect fingerprint PAIs. However, let us first review the state of the art in fingerprint PAD. Given the vast number of articles studying this problem, we will summarise the most relevant ones for the present study and refer the reader to [47, 67, 72] for more comprehensive reviews.

In general, PAD approaches can be broadly classified into two categories: *software-based* methods perform a deeper analysis of the captured data to distinguish between bona fide and attack presentations, *hardware-based* setups make use of information captured by additional sensors. In contrast to the younger finger vein PAD research field, where only the former have been studied so far, for fingerprint PAD both approaches have been followed. Tables 14.2 and 14.3 provide a summary of the reviewed works, classified into soft- and hardware-based approaches. In addi-

Table 14.2 Summary of the most relevant methodologies for **software-based fingerprint** presentation attack detection. For performance evaluation, the metrics are the ones reported in the articles, where CCR stands for correct classification rate and ACER for average classification error rate

Year	References	Description	Performance	#PAI	Database
2007	[11]	Score fusion of pore spacing, noise, and statistical properties	CCR = 85.2%	1	Own DB
2008	[53]	LBP texture and wavelet energy fusion	CCR = 97.4%	2	Own DB
2011	[17]	Closed sweat pore extraction	APCER = 21.2%	4	Own DB
			BPCER = 8.3%		
	[50]	Active sweat pore localisation	N/A	0	BFBIG-DB1
2014	[22]	25 image quality metrics	APCER $<$ 13%	3	LivDet 2009
			BPCER $\leq$ 14%		
	[40]	Multiscale LBP	D-EER = 7.52%	7	LivDet 2011
2016	[54]	Pre-trained CNNs (Best: VGG)	ACER = 2.90%	8	LivDet 2009-13
2017	[32]	Bag of Words and SIFT	APCER = 5%	7	LivDet 2011
			BPCER = 4.3%		
2018	[41]	LBP extracted from Gaussian pyramids (PLBP)	ACER = 21.21%	7	LivDet 2013
	[12]	Minutiae-centred CNN several different scenarios	APCER $<$ 7.3%	12	LivDet 2011-15, MSU-FPAD, PBSKD
			BPCER = 1%		
	[13]	Minutiae-centred CNN generalisation	APCER = 4.7%	12	MSU-FPAD, PBSKD
			BPCER = 0.2%		

tion, the number of PAI species and the main performance metrics over the selected databases are reported.

A typical example of software-based approaches is the detection of sweat pores in high-resolution fingerprint images [11, 17, 50]. Sweat pores are not visible in latent fingerprints and, because of their tiny size, it is challenging to include them in artefacts. Therefore, the existence of sweat pores can be utilised as an indicator of a bona fide sample.

Another classical approach, widely applied not only to fingerprint but to other biometric characteristics, is the extraction of textural information. Nikam and Agarwal [53] were among the first ones in 2008 to analyse this kind of approaches. On the one hand, they extracted Local Binary Pattern (LBP) histograms to capture textural details. On the other hand, the ridge frequency and orientation information were characterised using wavelet energy features. Both feature sets were fused and the dimensionality reduced with the Sequential Forward Floating Selection (SFFS) algorithm. For classification, the authors utilised a hybrid classifier, formed by fusing three classifiers: a neural network, SVMs and K-nearest neighbours. Over a self-

Table 14.3 Summary of the most relevant methodologies for **hardware-based fingerprint** presentation attack detection. For performance evaluation, the metrics are the ones reported in the articles

Year	References	Description	Performance	#PAI	Database
2011	[34]	Multi-spectral blanching effect, pulse	APCER = 0%	4	Own DB
			BPCER = 0%		
2013	[15]	Optical methods pulse, pressure, skin reflections	APCER = 10%	N/A	Own DB
			BPCER $< 2\%$		
2018	[29]	SWIR spectral signatures + SVM	APCER = 5.7%	12	Own DB
			BPCER = 0%		
	[73]	SWIR + CNN	APCER = 0%	12	Own DB
			BPCER = 0%		
	[43]	LSCI + SVM BSIF, LBP, HOG, histogram	APCER = 15.5%	32	Own DB
			BPCER = 0.2%		
	[37]	SWIR, LSCI + patch-based CNN	APCER = 0%	17	Own DB
			BPCER = 0%		
	[30]	Weighted score fusion + SVM SWIR, LSCI, vein	APCER = 6.6%	35	Own DB
			BPCER = 0.2%		
2019	[72]	SWIR + CNN fusion (pre-trained and from scratch)	APCER $\approx 7\%$	35	Own DB
			BPCER = 0.1%		
	[31]	Fusion of: SWIR + CNN and LSCI + hand-crafted features	APCER $\leq 3\%$	35	Own DB
			BPCER $\leq 0.1\%$		

acquired database comprising two different PAI fabrication materials and several mould materials, an overall classification rate up to 97.4% is reported.

In 2009, the LivDet competition series on fingerprint and iris started in a bi-annual basis [25]. The datasets provided quickly became the de facto standard for fingerprint PAD evaluations. For instance, Jia et al. [40] continued the research line based on texture information and proposed the use of two different variants of multi-scale LBP in combination with SVMs. Over the LivDet 2011 dataset, their method achieved a D-EER of 7.52%. More recently, Jiang et al. presented another approach to extract LBP features from multiple scales in [41]. In particular, a Gaussian pyramid was constructed from the input samples and the corresponding LBP histograms, extracted from three different levels, were classified using an SVM. Achieving an

ACER of 21% over the LivDet 2013 dataset, this method outperformed the algorithms presented in the competition.

In a more general approach, Galbally et al. [22] use 25 complementary image quality features to detect presentation attacks for face, iris and fingerprint on legacy data. Regarding fingerprint, they compare their approach with other state-of-the-art methods on the LivDet 2009 fingerprint database, which includes three different PAI species. Their results are competitive for 2014 and even outperform some previously published PAD algorithms on the same dataset. Their main advantage is its independency of the modality, and, additionally, the method is "simple, fast, non-intrusive, user-friendly, and cheap".

All the aforementioned approaches focus on the basic scenario where all PAI species in the test set are also included in the training test. However, a more realistic, and challenging, scenario should include additional "unknown attacks", or PAI species only used for testing purposes. In such a case, the detection performance usually decreases. To tackle this issue, Gonzalez-Soler et al. analysed in [32] the use of the Bag of Words feature encoding approach applied to local keypoint-based descriptors (dense Scale Invariant Feature Transform, SIFT). They compare their detection performance with other existing methods using feature descriptors, with no encoding schemes, and show a relative 25% improvement on the average Average Classification Error Rate (ACER, the performance metric used in the LivDet competitions) over the LivDet 2011 with respect to the state of the art. In addition, they present a fully compliant ISO evaluation in terms of APCER and BPCER for the first time for the LivDet datasets.

In contrast to the handcrafted approaches mentioned above, most of the newest approaches rely on deep learning. One of the first works directly related to fingerprint PAD based on conventional capture devices (i.e. a software-based method), was carried out by Nogueira et al. [54]. In more details, the following three CNNs were tested: (i) the pre-trained VGG [66], (ii) the pre-trained Alexnet [46] and (iii) a CNN with randomly initialised weights and trained from scratch. The authors benchmarked the ACER obtained with the networks over the LivDet 2009, 2011 and 2013 databases to a classical state of the art algorithm based on LBP. The best detection performance is achieved using a VGG pre-trained model and data augmentation (average ACER = 2.9%), with a clear improvement with respect to LBP (average ACER = 9.6%). It should be also noted that the ACER decreased between 25% and 50% (relative decrease) for all three networks tested when data augmentation was used.

More recently, Chugh et al. presented the current state of the art for the LivDet datasets in [12], and they evaluated it on multiple publicly available datasets including three LivDet datasets (2011, 2013, 2015), as well as their own collected and published MSU-FPAD and Precise Biometric Spoof-Kit datasets (PBSKD), which include in total 12 PAI species and more than 20000 samples. The so-called *Fingerprint Spoof Buster* [12] is a convolutional neural network (CNN) based on MobileNet [35], which is applied to minutiae-centred patches. Splitting the CNN input into patches allows them to train the network from scratch without over-fitting. They evaluate several different test scenarios and outperform other state-of-the-art approaches on the LivDet datasets. In a subsequent work [13], the *Fingerprint Spoof Buster's* gen-

eralisation capability is analysed by applying a leave-one-out protocol on all 12 PAI species from the MSU-FPAD and PBSKD datasets. They observe that some materials are harder to detect when not included during training and specify an optimised training set comprising six of twelve PAIs. The testing results in an APCER of 4.7% at a BPCER of 0.2%.

Even if the aforementioned works manage to achieve remarkably low error rates, PAD can also benefit from information captured by additional sensors, as any other pattern recognition task. To that end, some hardware-based approaches utilise different illumination techniques or capture the pulse frequencies. Hengfoss et al. [34] analysed in 2011 the reflections for all wavelengths between 400 and 1650 nm on the blanching effect. This effect appears when the finger is pressed against a surface and the blood is squeezed out due to the compression of the tissue. Furthermore, they utilise pulse oximetry but admit that this approach takes more time and thus is less desirable for PAD. They manage to correctly distinguish living fingers, cadaver fingers and three PAIs for both methods, and conclude that those dynamic effects (i.e. blanching and pulse) only occur for living fingers. Two years later, Drahansky et al. [15] proposed new optical handcrafted PAD methods for pulse, colour change under pressure and skin reflection for different wavelengths (470, 550 and 700 nm). These methods are evaluated on a database comprising 150 fingerprints, achieving the best results for the wavelength approach. Additionally, they analyse 11 different skin diseases that could occur on the fingertip. However, the influence on the detection performance was not tested.

Over the last five years, it has been shown that the skin reflection within the Short-wave Infrared (SWIR) spectrum of 900–1700 nm are independent from the skin tone. This fact was first analysed by NIST [14] and later on confirmed by Steiner et al. [68] for face PAD. Building upon the work of [68], Gomez-Barrero et al. [29] apply the spectral signature concept first developed for facial images to fingerprint PAD. Their preliminary experiments, over a rather small database, show that most materials, except for orange play doh, respond different than human skin in the SWIR wavelengths of 1200, 1300, 1450 and 1550 nm. However, with the use of fine-tuned CNNs, also the orange play doh is correctly classified in a subsequent work [73]. In a follow-up study [72], Tolosana et al. benchmark both pre-trained CNN models, and design and train a new residual CNN from scratch for PAD purposes for the same SWIR data. Over a larger dataset including 35 different PAI species and more than 4700 samples, they show that a combination of two different CNNs can achieve a remarkable performance: an APCER around 7% for a BPCER of 0.1%. In addition, the evaluation protocol includes 5 PAI species considered only for testing, thereby proving the soundness of their approach even in the presence of unknown attacks.

Additionally, it has been shown that Laser Speckle Contrast Imaging (LSCI) can be used for PAD purposes [43]. The LSCI technique comes from biomedical applications, where it has been applied to visualise and monitor microvascular blood flow in biological tissues, such as skin and retina [65]. Keilbach et al. capture the blood movement beneath the skin to differentiate living fingers from presentation attacks in [43]. However, the utilised laser also penetrates thin transparent fingerprint overlays, thereby detecting the underlying blood flow and falsely classifying the presentation

as a bona fide one. Therefore, for a BPCER of 0.2% (system focused on the user convenience), the APCER increases to 15.5%.

Combining SWIR and LSCI, Hussein et al. [37] use a patch-based CNN to classify multi-spectral samples from both domains. For both techniques, low error rates are reported and a combined fusion achieves a perfect detection performance over a database compromising 551 bona fides and 227 PAs, including 17 different PAI species.

Further research by Gomez-Barrero et al. [30] applies a score-level fusion method based on handcrafted features to benefit from different domains, including SWIR, LSCI and vein images. Their training set comprises only 136 samples in order to evaluate the approach on 4531 samples in the test set containing 35 different PAI species. The weights for the fusion are computed on 64 samples of the development set. An APCER $< 10\%$ for a BPCER $= 0.1\%$ is reported, as well as an APCER of 6.6% for a BPCER $= 0.2\%$, thus yielding secure systems even for very low BPCERs.

Lastly, in a subsequent work by Gomez-Barrero et al. [31], the SWIR CNN approaches proposed in [72] are combined with an enhancement of the handcrafted features extracted from the LSCI data in [43]. This combined approach, tested on the same database comprising 35 different PAI species, shows a clear improvement on the detection capabilities of the proposed method, even if only 2 sets of images are used (i.e. reduced capture device cost): the D-EER is reduced from 2.7 to 0.5%.

14.4 Proposed Finger Vein Presentation Attack Detection

As indicated in Sect. 14.1, we will now focus on the development of PAD techniques based on finger vein data, in order to detect fingerprint PAIs. It should be noted that the PAD algorithm can process data that is captured simultaneously with a single capture device from both the finger vein and the fingerprint. Otherwise, if the capture with both sensors was done sequentially, the attacker might exchange the PAI used for fingerprint verification with his bona fide finger for the PAD capture process. Therefore, in this section, we first describe a multimodal capture device which is able to acquire both fingerprint and finger vein images (Sect. 14.4.1). We subsequently present an efficient PAD method applied to the finger vein data in Sect. 14.4.2. Given that some fingerprint overlays may still reveal part of the vein structure, we will focus on texture analysis to detect PAs in a real-time fashion using a single image.

14.4.1 Multimodal Finger Capture Device

Given the requirement to capture both fingerprint and finger veins, a contact-less multimodal capture device is used to acquire photos of fingerprints as well as finger veins. A diagram of the inner components of the capture device is depicted in Fig. 14.1. As it may be observed, the camera and illumination boards are placed inside a closed

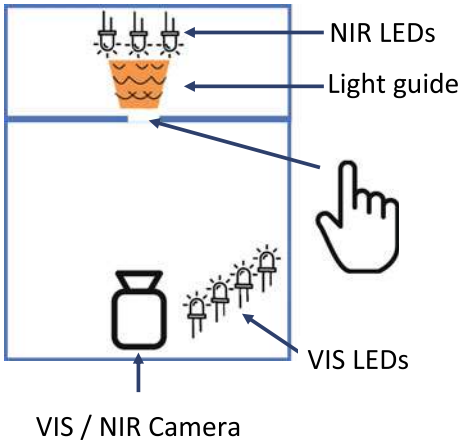
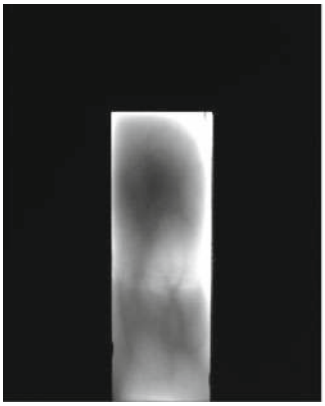
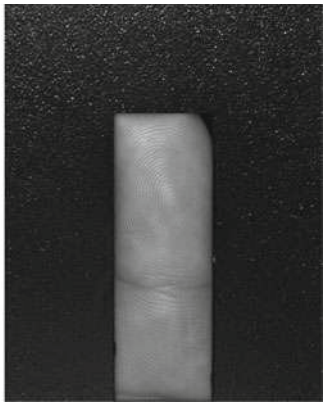


Fig. 14.1 Sensor diagram: a box, with a slot in the middle to place the finger, encloses all the components: a single camera, two sets of LEDs for visible (VIS) and NIR illumination and the light guide necessary for the finger vein capture (more details in Sect. 14.4.1.2)



(a) Finger vein (NIR) sample.



(b) Finger photo (VIS) sample.

Fig. 14.2 Full bona fide samples as they are captured by the camera

box, which includes an open slot in the middle. When the finger is placed there, all ambient light is blocked and therefore only the desired wavelengths are used for the acquisition of the images. In particular, we have used a Basler acA1300-60gm Near-infrared (NIR) camera, which captures 1280 × 1024 px. images, with an Edmunds Optics 35mm C Series VIS-NIR Lens. This camera is used for both frontal visible (VIS) light images and NIR finger vein samples (see the following subsections for more details on each individual sensor).

An example finger photo as it is captured by the camera is shown in Fig. 14.2, for both the finger vein and the finger photo acquisition. As it can be seen, the

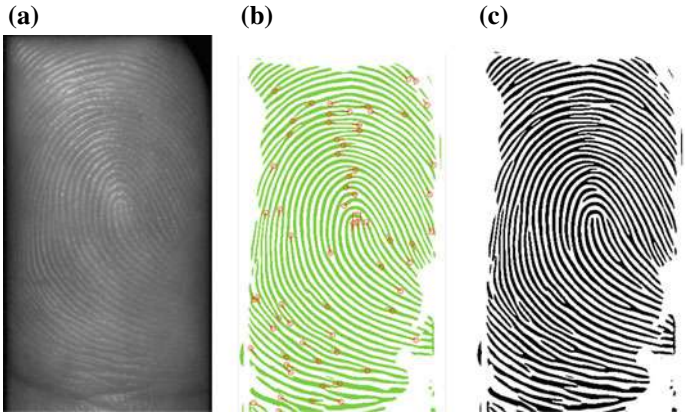


Fig. 14.3 Bona fide finger photos: **a** visible (VIS) light image, **b** minutiae extracted with Verifinger and **c** fingerprint enrolled with Verifinger

central Region of Interest (ROI) corresponding to the open slot where the finger is placed needs to be extracted from the background before the images can be further processed. Given that the finger is always placed over the open slot, and the camera does not move, a simple fixed size cropping can be applied.

14.4.1.1 Finger Photo Sensor

The most important requirement for the design of the finger photo sensor is its compatibility with legacy (optical) sensors. In other words, we need to make sure that fingerprints can be extracted from the finger photos captured within the visible wavelengths and be subsequently used for verification with Commercial off-the-shelf (COTS) systems. In order to fulfil this requirement, the resolution and focus of the selected camera and lens combination need to be high enough to yield fingerprints with at least the equivalence to 500 dpi resolution. We have therefore chosen the aforementioned Basler and Edmunds Optics components.

To illustrate how the finger photos can be used for fingerprint recognition, Fig. 14.3 shows the captured bona fide sample (Fig. 14.3a). Next to it, the minutiae extracted with Neurotechnology VeriFinger SDK[6] (Fig. 14.3b), which has been defined as the standard fingerprint recognition SDK within the Odin program, and the corresponding enrolled fingerprint (Fig, 14.3c) are depicted. As it may be observed, the minutiae are correctly detected within the fingerprint area. It should be noted that, if this system should be used in combination with optical sensors, the finger photo needs to be flipped (left-to-right) before enrolment or comparison.

[6] https://www.neurotechnology.com/verifinger.html.

Fig. 14.4 Bona fide finger vein ROI, of size 830 × 240 px

14.4.1.2 Finger Vein Sensor

The finger vein capture device comprises three main components, namely: (i) a NIR light source behind the finger with 20 LEDs of 940 nm, (ii) the corresponding NIR camera and lens and (iii) an elevated physical structure to obtain the adequate amount of light.

It should be noted that, in order to capture high-quality finger vein samples, it is vital to let only the right amount of light intensity penetrate through the finger. To achieve the correct amount of light transmission, a physical structure with elevation is placed to concentrate the light intensity to the specified area, referred to in Fig. 14.1 as "light guide". The subject interacts with the sensor by placing a finger on the small gap provided between the NIR light source and the camera. The NIR spectral light is placed facing the camera in a unique way, so that the light emitting from the NIR spectrum penetrates through the finger. Since the haemoglobin blocks the NIR illumination, the veins appear as darker areas in the captured image. A sample image is depicted in Fig. 14.4, where the veins are clearly visible even before preprocessing the sample.

14.4.2 Presentation Attack Detection Algorithm

As mentioned at the beginning of this Section, we will focus on texture analysis of the finger vein samples in order to discriminate bona fide samples from presentation attacks. To that end, we have chosen a combination of Gaussian pyramids and Local Binary Patterns (LBP), referred to as PLBP, which was proposed in [57] as a general descriptor. The main advantage of this texture descriptor lies on the fact that, by extracting the LBP features from the hierarchical spatial pyramids, texture information at different resolution levels can be considered. In fact, the PLBP approach was used in [41] for fingerprint PAD over the LivDet 2013 DB [24], achieving results within the state of the art for only three pyramid levels. In order to analyse the influence of the different pyramid levels, we compare the results using up to 16 pyramid levels.

The flowchart of the proposed method is shown in Fig. 14.5. First, the Gaussian pyramids are computed from the original cropped image or ROI (see Fig. 14.4). Subsequently, LBP images are generated for every pyramid level, resulting in the PLBP images. Then, histograms are computed from the PLBP images and classified

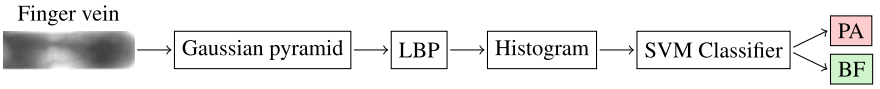


Fig. 14.5 General diagram of the proposed PAD algorithm. From the finger vein photo, the Gaussian pyramid is computed first, then LBP is applied and the corresponding histogram serves as input to the SVM classifier

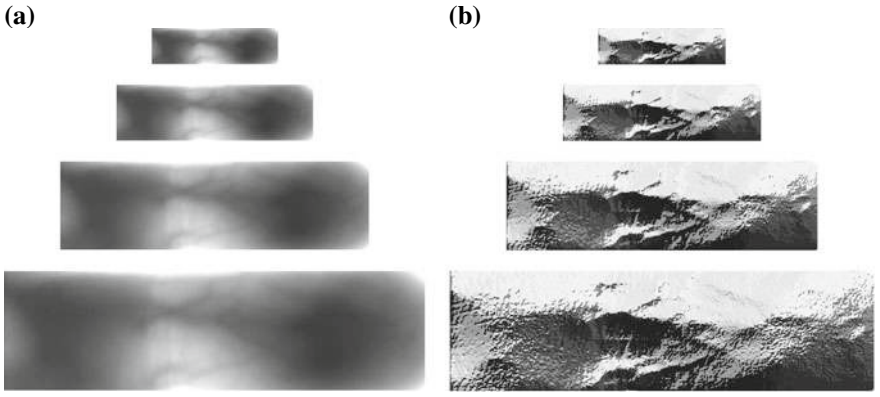


Fig. 14.6 Illustration of example pyramids for: **a** Gaussian pyramid of vein images and **b** LBP images of this Gaussian pyramid

with a Support Vector Machine (SVM). Each step is described in more detail in the following paragraphs.

Gaussian pyramids. For multi-resolution analysis, lowpass pyramid transforms are widely used [8]. In particular, the Gaussian blur lowpass filter can be used to down-sample the original image. This step can be repeated to get continuously smaller images, resembling a pyramid, as depicted in Fig. 14.6. In practice, one pixel of the down-sampled image corresponds to a fixed size area of the previous pyramid level, thereby losing information the further up we go into the pyramid. However, in our implementation, all levels of the pyramid have the same size, which is obtained by up-sampling the output image in each iteration. As a consequence, the higher level images appear blurrier.

It should be highlighted that, in our implementation, different pyramids with up to 16 levels are created. This allows us to determine how the PAD performance change when more levels of the pyramid are used.

Local Binary Patterns (LBP). Local binary patterns were introduced in [56] as a simple but efficient texture descriptor. Its computational simplicity and greyscale invariance are the most important properties of LBP. The algorithm compares neighbouring pixels and returns the result as a binary number, which is in turn stored as a decimal value. The process is illustrated in Fig. 14.7 for a radius of 1 pixel (3×3 block). It should be noted that the binary representation can also be flipped and the direction and starting point of reading the binary number does not matter as long

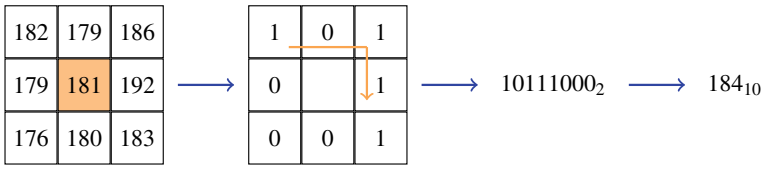
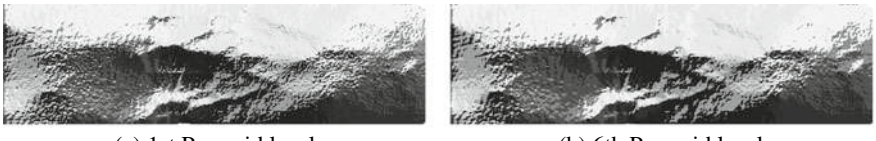
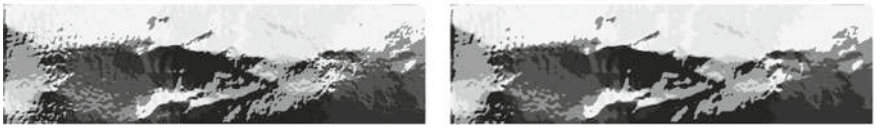


Fig. 14.7 LBP computation: Comparing the central pixel (orange) to each neighbouring pixel results in a binary representation. The binary values are converted to a decimal number, which is stored in the resulting LBP image instead of the original central pixel



(a) 1st Pyramid level (b) 6th Pyramid level



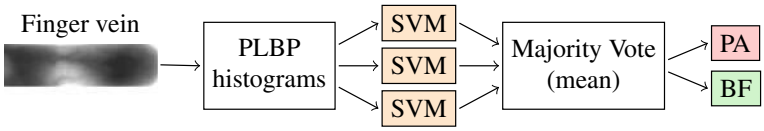
(c) 11th Pyramid level (d) 16th Pyramid level

Fig. 14.8 Resulting bona fide LBP images of different Gaussian pyramid levels (i.e. PLBP images)

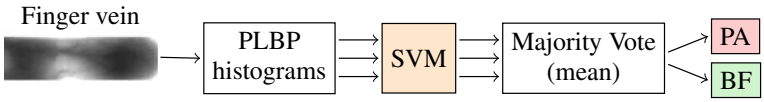
as it is fixed for the whole system (otherwise, the extracted feature would not be comparable). An example of the four selected PLBP images of the bona fide sample shown in Fig. 14.4 is presented in Fig. 14.8.

Classification. In order to reduce the dimensionality of the feature vector, a greyscale histogram is computed from the resulting LBP images. Subsequently, linear SVMs are used to classify the extracted histograms. These SVMs rely on a main parameter, C, which can be tuned for an optimal performance. Intuitively, the C parameter trades off misclassification of training examples against simplicity of the decision surface. A low C makes the decision surface smooth, while a high C aims at classifying all training examples correctly by giving the model freedom to select more samples as support vectors.

In addition, we benchmark two SVM approaches, as shown in Fig. 14.9 for the simple case of three pyramid levels. On the one hand, we use separate SVMs for each pyramid level (Fig. 14.9a). On the other hand, we utilise a single SVM for all pyramid levels (Fig. 14.9b). Both setups produce one label per pyramid level and then apply a majority vote on the corresponding SVM outputs in order to reach a final decision.



(a) Separate SVMs for 3 pyramid levels.



(b) Single SVM for 3 pyramid levels.

Fig. 14.9 Diagram of the two SVM approaches on the example of 3 pyramid levels

14.5 Experimental Evaluation

With the aim of analysing the suitability of the proposed method for finger vein-based PAD, several experiments were carried out using an identical experimental protocol. Our training and test sets are completely disjoint in order to avoid biased results. Furthermore, in order to allow reproducibility of the experiments, preprocessing and feature extraction are based on the bob toolkit [4, 5].

14.5.1 Experimental Set-Up

The captured dataset comprises 766 samples including 542 bona fides and 224 presentation attacks, stemming from 32 different PAI species. The PAs can be classified into three categories, namely: (i) 2D printouts, (ii) full fingers and (iii) overlays, whereby 2D printouts can also be used as an overlay during the presentation. A detailed listing of all PAIs from the database is presented in Table 14.4.

All samples were captured within the BATL project with our project partners at the University of Southern California. Note that the project sponsor has indicated that they will make the complete dataset available in the near future such that research results presented in this work can be reproduced.

We have additionally considered two test scenarios (see Table 14.5). The first one uses the same number of bona fides and PAs in the training set (69 samples each). To increase the robustness on the detection of bona fide presentations (i.e. minimise the BPCER), the second scenario adds additional 35 bona fide samples to the training set, thus reducing the test set. The partitioning for both scenarios is shown in Table 14.5. Both approaches, using a single SVM or separated SVMs, are compared using the same training and test sets for each scenario.

In more details, the training set comprises all different PAIs except from dragonskin overlays, since this thin and transparent material does not block NIR illumination as known from previous experiments [30]. As a consequence, all veins are visible

Table 14.4 Listing of all PAI species and the number of samples in parenthesis

2D printouts	Matte paper (10) Transparent (8)
Full fingers	3D printed (24) 3D printed + silver coating (9) dragon-skin (6) dragon-skin + conductive paint (9) dragon-skin + conductive paint + nanotips (9) dragon-skin + graphite coating (9) latex + gold coating (8) play doh (28) *in black, blue, green, orange, pink, purple, red, teal (3 each) and yellow (4)* silicone (7) silicone + bare paint (13) silicone + graphite coating (9) silicone + nanotips (6) silly putty (3) silly putty metallic (6) silly putty "glowing in the dark" (6) wax (6)
Overlays	dragon-skin (9) monster latex (10) school glue (6) silicone (13) urethane (6) wax (4)

Table 14.5 Partitioning of training and test data

		# Samples	# PA samples	# Bona fide samples
Scenario 1	Train set	138	69 (50%)	69 (50%)
	Test set	628	155 (25%)	473 (75%)
Scenario 2	Train set	173	69 (40%)	104 (60%)
	Test set	593	155 (26%)	438 (74%)

and the sample has the same appearance as a bona fide. Using such samples to train the SVM would thus have a negative impact on its detection accuracy, increasing the BPCER. These PAIs are therefore used only for testing purposes.

In the first scenario, cross-validation is used during the training to automatically select a best-fitting C value as SVM parameter. As suggested by Hsu et al. [36], expo-

nential growing sequences for C (2^x) were tested within the range $x = \{-20, ..., 20\}$. However, due to the increased number of training samples for the second scenario, and consequently, the training time required, only the range $x = \{-20, ..., 8\}$ has been used to cross-validate scenario 2.

Finally, all results are reported in terms of the APCER and BPCER over the test set (see Sect. 14.2), in compliance with the ISO/IEC 30107-3 standard on biometric presentation attack detection - part 3: testing and reporting [39].

It should be noted that establishing a fair benchmark with previous works in the state of the art are difficult since this is the first approach to carry out fingerprint PAD based on finger vein samples.

14.5.2 Results

The results in terms of APCER (dashed) and BPCER (solid) for scenario 1 are plotted in Fig. 14.10, in order to facilitate the visualisation and comparison across different pyramid levels. On the x-axis, the range of pyramid levels are given while the y-axis shows the error rates (in %). For the single SVM approach (Fig. 14.10a), both error rates reach a minimum when using 6 pyramid levels, namely, BPCER = 3.38% and APCER = 5.81%. On the other hand, for the separate SVM approach (Fig. 14.10b), the minimum of both error rates is reached at different levels, namely, BPCER = 2.54% for the fifth level and APCER = 6.45% for the fourth level. This means that, depending on the application at hand (i.e. which error rate should be optimised), different levels may be selected. As it may be observed from Fig. 14.10, the error rates of the separate SVMs somewhat stabilise for using five or more pyramid levels, whereas the single SVMs show much more peaks and no stabilisation.

Regarding the aforementioned decision of prioritising one error rate over the other one, it should be taken into account that a low BPCER results in user convenience (i.e. a low number of bona fide presentation will be wrongly rejected). On the other hand, a low APCER will grant a more secure system (i.e. the number of non-detected attacks will be minimised). One of the aims of the Odin program is achieving a low BPCER. To that end, we analyse the second scenario, for which more training samples for the bona fide class are utilised in order to make the classifier more robust. The corresponding plots with the APCER and BPCER for every pyramid level are presented in Fig. 14.11.

We can observe that the BPCER is significantly lower for all pyramid levels when compared to scenario 1, reaching minimum values of 0.68% for the single SVM and 2.28% for the separate SVMs. At the same time, the APCER stays similar to that of scenario 1, thereby showing the soundness of increasing the number of bona fide samples for training. Additionally, we can see that using only the first four levels produces higher peaks and higher error rates, thus making it unsuitable for PAD purposes. In turn, increasing the number of levels results in a decreasing BPCER, as can be seen for the levels greater than four. Taking into account the pyramid levels five to sixteen, the average APCER is slightly lower for the single SVM approach

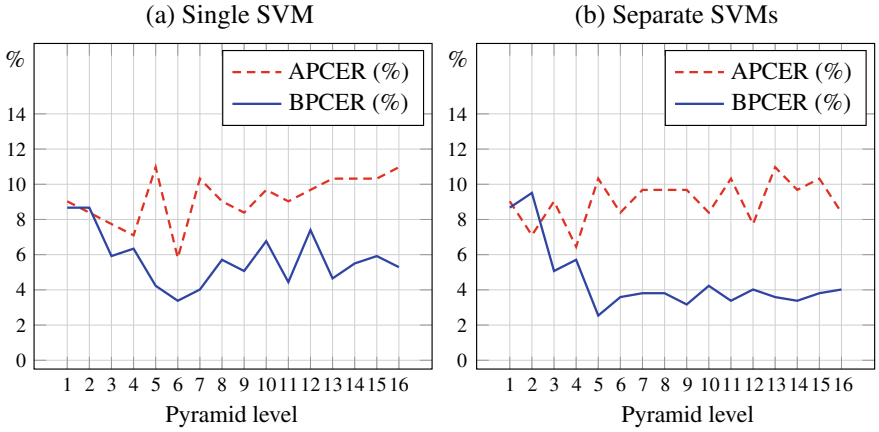


Fig. 14.10 Percentage of APCER and BPCER of **scenario 1** for both SVM classifiers

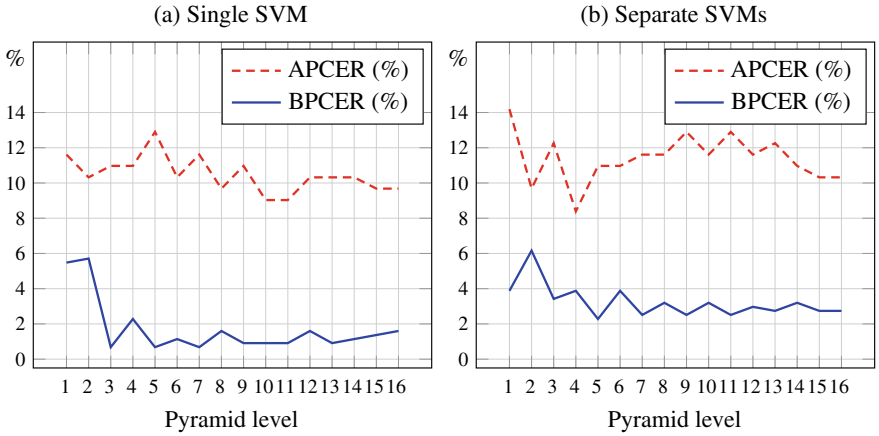


Fig. 14.11 Percentage of APCER and BPCER of **scenario 2** for both SVM classifiers

(10.32–11.50%), while the average BPCER improves significantly for the single SVM (1.12–2.87%). Therefore, we may conclude that the single SVM approach achieves a better PAD performance than the separate SVMs since the training set of the latter is not big enough to train one pyramid level independently of the others. The single SVM gets complimentary information when seeing all levels together and is thus able to reach a higher detection performance.

A comparison for both scenarios of the single SVM approach (level 7) to other handcrafted state-of-the-art implementations is given in Table 14.6. The *Luminosity* and *MC mean* algorithms operate on a very convenient threshold but classify only a fraction of presentation attacks correctly (APCER = 68.39% and APCER = 43.87%, respectively). The other algorithms use a support vector machine for classification

Table 14.6 Comparison of the proposed method to state-of-the-art implementations

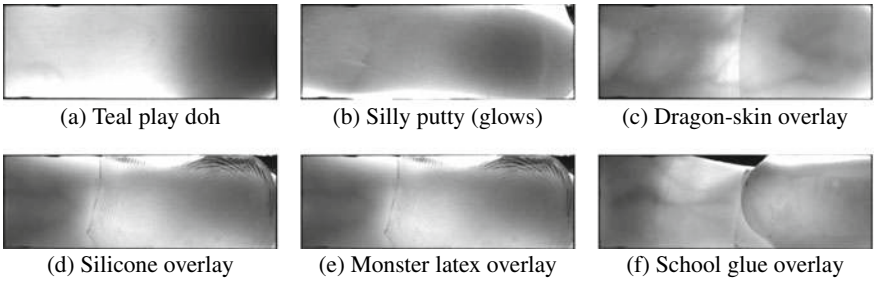
Algorithm	Scenario 1		Scenario 2	
	APCER	BPCER	APCER	BPCER
Luminosity [30]	68.39	0.00	68.93	0.00
MC mean [30]	43.87	0.21	43.87	0.23
MC histogram [30]	13.55	9.51	12.90	8.22
BSIF [42]	28.39	5.71	26.45	4.57
LBP [56]	10.32	1.90	11.61	1.14
Proposed PLBP (lvl 7)	10.32	4.02	11.61	0.68

and present lower APCERs. However, in some cases, the BPCER raises to nearly 10%. In particular, the *MC histogram* achieves an APCER between 12 and 14%while the BPCER is between 8 and 10%. In contrast, the *BSIF* implementation results in a BPCER of around 5% at the cost of a higher APCER (26–29%). The results of the plain *LBP* implementation and the *proposed PLBP* implementation are identical regarding APCER but differ in the BPCER. Whereas for scenario 1 *LBP* provides a better BPCER of 1.9% compared to 4.02%, the *proposed PLBP* approach reduces its BPCER in scenario 2 to 0.68% in contrast to 1.14% for *LBP*. Therefore, we can see that our PLBP algorithm achieves the best results for scenario 2 while it is outperformed by *LBP* in scenario 1. The score files from all tests in this chapter are freely available.[7]

Even if the results are promising, reaching an APCER $\approx$ 10% for BPCER $\approx$ 1%, where also unknown attacks (i.e. only used for testing and not seen by the classifier at training) are considered, there is still room for improvement. In particular, a deeper analysis of the results shows that a remarkable number of misclassified PAIs are transparent overlays made of dragon-skin, silicone, monster latex, school glue or wax. In addition, two types of full fake fingers also managed to deceive the PAD algorithm in some cases, namely, glow-in-the-dark silly putty, and one of the samples acquired from a teal play doh finger. Some samples that were not detected are shown in Fig. 14.12. As we may observe, especially for the dragon-skin (c) and the school glue (f) overlays, the samples are very similar to the bona fide sample shown in Fig. 14.4. In particular, the vein structure can be clearly seen.

Finally, Fig. 14.13 shows the 11th level of PLBP images for (a) a dragon-skin overlay, (b) a teal play doh finger, (c) a school glue overlay and (d) a 3D printed finger with silver coating. Comparing these samples with the bona fide one from Fig. 14.8, we can see the high similarities for the transparent overlays in (a) and (c). However, the teal play doh and the 3D printed finger have different patterns (i.e. the 3D printed finger does not block the NIR light at all, only the silver-coated part is

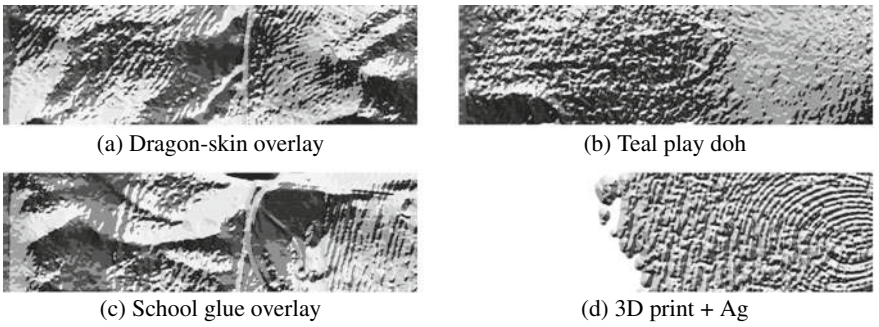
[7] https://dasec.h-da.de/research/biometrics/presentation-attack-detection-for-finger-recognition/.



(a) Teal play doh (b) Silly putty (glows) (c) Dragon-skin overlay

(d) Silicone overlay (e) Monster latex overlay (f) School glue overlay

Fig. 14.12 Examples of undetected PAI species



(a) Dragon-skin overlay (b) Teal play doh

(c) School glue overlay (d) 3D print + Ag

Fig. 14.13 Resulting LBP images of different PAIs for 11th Gaussian pyramid level (i.e. PLBP images)

visible). Hence, the SVMs always correctly classify the 3D printed PAIs, and only one error occurred for the teal play doh samples.

To sum up the findings in this section, we can state that the APCERs of around 10% show the limitations of vein-based still image PAD: thin transparent overlays cannot be detected since the extracted features look far too similar to the bona fide ones. However, this PAD technique already allows to successfully detect a wide range of PAIs, including full fake fingers and overlays fabricated from materials which block NIR light to a bigger extent than human flesh.

14.6 Summary and Conclusions

Although being relatively new in comparison with other biometric characteristics, such as fingerprints or handwritten signatures, finger vein recognition has enjoyed a considerable attention within the last decade. As with any other security-related technology, a wider deployment also implies an increase in security and privacy related concerns. This has, in turn, lead to the development of countermeasures to prevent, among others, presentation attacks.

In particular, the biometric community has focused on detecting finger vein images or videos presented to the capture device, in contrast to bona fide fingers. Highly accurate PAD methods have been developed in the literature, able to detect these PAIs with perfect error rates.

In parallel, multimodal capture devices able to acquire both finger vein and fingerprint images have been proposed and implemented. In contrast to the finger vein, which is harder to imitate, multiple recipes are available to an eventual attacker in order to carry out a PA and fool a fingerprint-based recognition system. These facts have motivated us to present in this chapter a novel approach to protect fingerprint sensors: finger vein PAD methods which are able to detect fingerprint PAIs.

In more details, due to the remarkable performance shown by LBP for different tasks, including PAD for several biometric characteristics, we chose this texture descriptor for our work. Even for some challenging PAIs, we can observe with the naked eye that the texture captured has a different appearance from the bona fide finger. In addition, different texture details were analysed utilising Gaussian pyramids and extracting the LBP features from each level of the pyramid. Subsequently, SVMs were utilised for classification purposes.

With a sensor developed for the Odin program, a database comprising 32 different PAIs was acquired and used for the present evaluation. After an extensive experimental evaluation, we found that using a single SVM for a concatenation of the features extracted from all the levels of the pyramid is the best performing approach. This scenario leads to operation points with BPCERs under 1% and an APCER around 10%. The latter shows the main limitation of vein-based still image PAD: thin transparent overlays cannot be detected. However, this PAD technique still allows to successfully detect a wide range of PAIs.

We thus believe that finger vein can be effectively used with fingerprint for both a more accurate recognition performance, as shown in previous works, and also for PAD purposes. In the end, an attacker who needs to deceive both the fingerprint and the vein sensors will face harder challenges in his path. In the forthcoming months, we will focus on improving the finger vein-based PAD, and on developing combined approaches with the finger photos captured with the sensor.

Acknowledgements This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA) under contract number 2017-17020200005. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorised to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.
We would also like to thank our colleagues at USC for the data collection efforts.

References

1. Adler A (2004) Images can be regenerated from quantized biometric match score data. In: Proceedings of Canadian conference on electrical and computer engineering (CCECE), pp 469–472
2. Akhtar Z, Kale S, Alfarid N (2011) Spoof attacks in mutimodal biometric systems. In: Proceedings of international conference on information and network technology (IPCSIT), vol 4, pp 46–51. IACSIT Press
3. Alegre F, Vipperla R, Evans N, Fauve B (2012) On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals. In: Proceedings of European signal processing conference (EUSIPCO), pp 36–40
4. Anjos A, Günther M, de Freitas Pereira T, Korshunov P, Mohammadi A, Marcel S (2017) Continuously reproducing toolchains in pattern recognition and machine learning experiments. In: Proceedings of international conference on machine learning (ICML)
5. Anjos A, Shafey LE et al (2012) Bob: a free signal processing and machine learning toolbox for researchers. In: Proceedings ACM international conference on multimedia (ACM MM), pp 1449–1452
6. BATL: Biometric authentication with a timeless learner (2017)
7. Bhogal APS, Söllinger D, Trung P, Hämmerle-Uhl J, Uhl A (2017) Non-reference image quality assessment for fingervein presentation attack detection. In: Proceedings Scandinavian conference on image analysis (SCIA), pp 184–196
8. Burt PJ, Adelson EH (1987) The Laplacian pyramid as a compact image code. In: Readings in computer vision, pp 671–679. Elsevier
9. Cappelli R, Maio D, Lumini A, Maltoni D (2007) Fingerprint image reconstruction from standard templates. IEEE Trans Pattern Anal Mach Intell 29:1489–1503
10. Chetty G, Wagner M (2005) Audio-visual multimodal fusion for biometric person authentication and liveness verification. In: Proceedings of NICTA-HCSNet multimodal user interaction workshop (MMUI)
11. Choi H, Kang R, Choi K, Kim J (2007) Aliveness detection of fingerprints using multiple static features. In: Proceedings of world academy of science, engineering and technology, vol 22
12. Chugh T, Cao K, Jain AK (2018) Fingerprint spoof buster: use of minutiae-centered patches. IEEE Trans Inf Forensics Secur 13(9):2190–2202
13. Chugh T, Jain AK (2018) Fingerprint presentation attack detection: generalization and efficiency. arXiv:1812.11574
14. Cooksey C, Tsai B, Allen D (2014) A collection and statistical analysis of skin reflectance signatures for inherent variability over the 250 nm to 2500 nm spectral range. In: Active and passive signatures V, vol 9082, p 908206. International Society for Optics and Photonics
15. Drahansky M, Dolezel M, Michal J, Brezinova E, Yim J et al (2013) New optical methods for liveness detection on fingers. BioMed Res Int 2013:197,925
16. Erdogmus N, Marcel S (2014) Spoofing face recognition with 3D masks. IEEE Trans Inf Forensics Secur 9(7):1084–1097
17. Espinoza M, Champod C (2011) Using the number of pores on fingerprint images to detect spoofing attacks. In: International conference on hand-based biometrics (ICHB), 2011, pp 1–5. IEEE
18. Galbally J, Cappelli R, Lumini A, de Rivera GG, Maltoni D, Fierrez J, Ortega-Garcia J, Maio D (2010) An evaluation of direct and indirect attacks using fake fingers generated from ISO templates. Pattern Recogn Lett 31:725–732
19. Galbally J, Gomez-Barrero M (2017) Presentation attack detection in iris recognition. In: Busch C, Rathgeb C (eds) Iris and periocular biometrics. IET
20. Galbally J, Marcel S, Fierrez J (2014) Biometric antispoofing methods: a survey in face recognition. IEEE Access 2:1530–1552
21. Galbally J, Marcel S, Fierrez J (2014) Image quality assessment for fake biometric detection: application to iris, fingerprint and face recognition. IEEE Trans Image Process 23(2):710–724

22. Galbally J, Marcel S, Fierrez J (2014) Image quality assessment for fake biometric detection: application to iris, fingerprint, and face recognition. IEEE Trans Image Process 23(2):710–724
23. Galbally J, Ross A, Gomez-Barrero M, Fierrez J, Ortega-Garcia J (2013) Iris image reconstruction from binary templates: an efficient probabilistic approach based on genetic algorithms. Comput Vis Image Underst 117(10):1512–1525
24. Ghiani L, Yambay D, Mura V, Tocco S, Marcialis GL, Roli F, Schuckers S (2013) LivDet 2013 fingerprint liveness detection competition 2013. In: International conference on biometrics (ICB), 2013, pp 1–6. IEEE
25. Ghiani L, Yambay DA, Mura V, Marcialis GL et al (2017) Review of the fingerprint liveness detection (LivDet) competition series: 2009 to 2015. Image Vis Comput 58:110–128
26. Gomez-Barrero M, Galbally J (2017) Inverse biometrics and privacy. In: Vielhauer C (ed) User-centric privacy and security in biometrics. IET
27. Gomez-Barrero M, Galbally J (2017) Software attacks on iris recognition systems. In: Busch C, Rathgeb C (eds) Iris and periocular biometrics. IET
28. Gomez-Barrero M, Galbally J, Fierrez J (2014) Efficient software attack to multimodal biometric systems and its application to face and iris fusion. Pattern Recogn Lett 36:243–253
29. Gomez-Barrero M, Kolberg J, Busch C (2018) Towards fingerprint presentation attack detection based on short wave infrared imaging and spectral signatures. In: Proceedings of Norwegian information security conference (NISK)
30. Gomez-Barrero M, Kolberg J, Busch C (2018) Towards multi-modal finger presentation attack detection. In: Proceedings of international workshop on ubiquitous implicit biometrics and health signals monitoring for person-centric applications (UBIO)
31. Gomez-Barrero M, Kolberg J, Busch C (2019) Multi-modal fingerprint presentation attack detection: looking at the surface and the inside. In: Proceedings of international conference on biometrics (ICB)
32. González-Soler LJ, Chang L, Hernández-Palancar J, Pérez-Suárez A, Gomez-Barrero M (2017) Fingerprint presentation attack detection method based on a bag-of-words approach. In: Proceedings of Iberoamerican congress on pattern recognition (CIARP), pp 263–271. Springer
33. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press
34. Hengfoss C, Kulcke A, Mull G, Edler C, Püschel K, Jopp E (2011) Dynamic liveness and forgeries detection of the finger surface on the basis of spectroscopy in the 400–1650 nm region. Forensic Sci Int 212(1):61–68
35. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W et al (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861
36. Hsu CW, Chang CC, Lin CJ et al (2003) A practical guide to support vector classification
37. Hussein ME, Spinoulas L, Xiong F, Abd-Almageed W (2018) Fingerprint presentation attack detection using a novel multi-spectral capture device and patch-based convolutional neural networks. In: 2018 IEEE international workshop on information forensics and security (WIFS), pp 1–8. IEEE
38. International Organisation for Standardisation (2016) ISO/IEC JTC1 SC37 Biometrics: ISO/IEC 30107-1. Information technology—biometric presentation attack detection—part 1: framework
39. International Organisation for Standardisation (2017) ISO/IEC JTC1 SC37 Biometrics: ISO/IEC 30107-3. Information technology—biometric presentation attack detection—part 3: testing and reporting
40. Jia X, Yang X, Cao K, Zang Y, Zhang N, Dai R, Zhu X, Tian J (2014) Multi-scale local binary pattern with filters for spoof fingerprint detection. Inf Sci 268:91–102
41. Jiang Y, Liu X (2018) Uniform local binary pattern for fingerprint liveness detection in the gaussian pyramid. Hindawi J Electr Comput Eng
42. Kannala J, Rahtu E (2012) BSIF: binarized statistical image features. In: 2012 21st international conference on pattern recognition (ICPR), pp 1363–1366
43. Keilbach P, Kolberg J, Gomez-Barrero M, Busch C, Langweg H (2018) Fingerprint presentation attack detection using laser speckle contrast imaging. In: Proceedings international conference of the biometrics special interest group (BIOSIG), pp 1–6

44. Kocher D, Schwarz S, Uhl A (2016) Empirical evaluation of LBP-extension features for finger vein spoofing detection. In: Proceedings international conference of the biometrics special interest group (BIOSIG), pp 1–5. IEEE
45. Kono M, Umemura S, Miyatake T, Harada K et al (2004) Personal identification system. US Patent 6,813,010
46. Krizhevsky A, Sutskever I, Geoffrey E (2012) ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems, vol 25, pp 1097–1105. Curran Associates, Inc
47. Marasco E, Ross A (2015) A survey on antispoofing schemes for fingerprint recognition systems. ACM Comput Surv (CSUR) 47(2):28
48. Marcel S (2013) BEAT—biometrics evaluation and testing. Biom Technol Today 5–7
49. Marcel S, Nixon MS, Li SZ (eds) (2014) Handbook of biometric anti-spoofing. Springer
50. Memon S, Manivannan N, Balachandran W (2011) Active pore detection for liveness in fingerprint identification system. In: 2011 19th telecommunications forum (TELFOR), pp 619–622. IEEE
51. Nguyen DT, Park YH, Shin KY, Kwon SY et al (2013) Fake finger-vein image detection based on fourier and wavelet transforms. Digit Signal Process 23(5):1401–1413
52. Nguyen DT, Yoon HS, Pham TD, Park KR (2017) Spoof detection for finger-vein recognition system using NIR camera. Sensors 17(10):2261
53. Nikam SB, Agarwal S (2008) Texture and wavelet-based spoof fingerprint detection for fingerprint biometric systems. In: Proceedings of international conference on emerging trends in engineering and technology (ICETET), pp 675–680. IEEE
54. Nogueira RF, de Alencar Lotufo R, Machado RC (2016) Fingerprint liveness detection using convolutional neural networks. IEEE Trans Inf Forensics Secur 11(6):1206–1213
55. ODNI, IARPA: IARPA-BAA-16-04 (thor) (2016). https://www.iarpa.gov/index.php/research-programs/odin/odin-baa
56. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recogn 29(1):51–59
57. Qian X, Hua X, Chen P, Ke L (2011) PLBP: an effective local binary patterns texture descriptor with pyramid representation. Pattern Recogn 44(10):2502–2515
58. Qiu X, Kang W, Tian S, Jia W, Huang Z (2018) Finger vein presentation attack detection using total variation decomposition. IEEE Trans Inf Forensics Secur 13(2):465–477
59. Qiu X, Tian S, Kang W, Jia W, Wu Q (2017) Finger vein presentation attack detection using convolutional neural networks. In: Proceedings of Chinese conference on biometric recognition (CCBR), pp 296–305
60. Raghavendra R, Avinash M, Marcel S, Busch C (2015) Finger vein liveness detection using motion magnification. In: Proceedings of international conference on biometrics theory, applications and systems (BTAS), pp 1–7. IEEE
61. Raghavendra R, Busch C (2015) Presentation attack detection algorithms for finger vein biometrics: a comprehensive study. In: Proceedings of international conference on signal-image technology & internet-based systems (SITIS), pp 628–632
62. Raghavendra R, Raja K, Surbiryala J, Busch C (2014) A low-cost multimodal biometric sensor to capture finger vein and fingerprint. In: Proceedings of international joint conference on biometrics (IJCB)
63. Raghavendra R, Raja K, Venkatesh S, Busch C (2018) Fingervein presentation attack detection using transferable features from deep convolution neural networks. In: Vatsa M, Singh R, Majumdar A (eds) Deep learning in biometrics. CRC Press, Boca Raton
64. Ratha N, Connell J, Bolle R (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 40
65. Senarathna J, Rege A, Li N, Thakor NV (2013) Laser speckle contrast imaging: theory, instrumentation and applications. IEEE Rev Biomed Eng 6:99–110
66. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: Proceedings of international conference on learning representations (ICLR)

67. Sousedik C, Busch C (2014) Presentation attack detection methods for fingerprint recognition systems: a survey. IET Biom 3(1):1–15
68. Steiner H, Kolb A, Jung N (2016) Reliable face anti-spoofing using multispectral SWIR imaging. In: Proceedings of international conference on biometrics (ICB), pp 1–8
69. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: Proceedings of advances in neural information processing systems (NIPS)
70. TABULA RASA: Trusted biometrics under spoofing attacks (2010). http://www.tabularasa-euproject.org/
71. Tirunagari S, Poh N, Bober M, Windridge D (2015) Windowed DMD as a microtexture descriptor for finger vein counter-spoofing in biometrics. In: Proceedings of IEEE international workshop on information forensics and security (WIFS), pp 1–6
72. Tolosana R, Gomez-Barrero M, Busch C, Ortega-Garcia J (2019) Biometric presentation attack detection: beyond the visible spectrum. arXiv:1902.11065
73. Tolosana R, Gomez-Barrero M, Kolberg J, Morales A, Busch C, Ortega J (2018) Towards fingerprint presentation attack detection based on convolutional neural networks and short wave infrared imaging. In: Proceedings of international conference of the biometrics special interest group (BIOSIG)
74. Tome P, Marcel S (2015) On the vulnerability of palm vein recognition to spoofing attacks. In: Proceedings of international conference on biometrics (ICB), pp 319–325. IEEE
75. Tome P, Raghavendra R, Busch C, Tirunagari S et al (2015) The 1st competition on counter measures to finger vein spoofing attacks. In: Proceedings of international conference on biometrics (ICB), pp 513–518
76. Tome P, Vanoni M, Marcel S (2014) On the vulnerability of finger vein recognition to spoofing. In: Proceedings of international conference of the biometrics special interest group (BIOSIG), pp 1–10. IEEE
77. Vanoni M, Tome P, El Shafey L, Marcel S (2014) Cross-database evaluation with an open finger vein sensor. In: IEEE workshop on biometric measurements and systems for security and medical applications (BioMS)
78. Wu HY, Rubinstein M, Shih E, Guttag J, Durand F, Freeman W (2012)Eulerian video magnification for revealing subtle changes in the world. In: Proceedings of transaction on graphics (SIGGRAPH)
79. Yambay D, Czajka A, Bowyer K, Vatsa M, Singh R, Schuckers S (2019) Review of iris presentation attack detection competitions. In: Handbook of biometric anti-spoofing, pp 169–183. Springer
80. Yambay D, Ghiani L, Marcialis GL, Roli F, Schuckers S (2019) Review of fingerprint presentation attack detection competitions. In: Handbook of biometric anti-spoofing, pp 109–131. Springer
81. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: Proceedings of international conference on computer vision and pattern recognition (CVPR)

Chapter 15
On the Recognition Performance of BioHash-Protected Finger Vein Templates

Vedrana Krivokuća and Sébastien Marcel

Abstract This chapter contributes towards advancing finger vein template protection research by presenting the first analysis on the suitability of the BioHashing template protection scheme for finger vein verification systems, in terms of the effect on the system's recognition performance. Our results show the best performance when BioHashing is applied to finger vein patterns extracted using the Wide Line Detector (WLD) and Repeated Line Tracking (RLT) feature extractors, and the worst performance when the Maximum Curvature (MC) extractor is used. The low recognition performance in the Stolen Token scenario is shown to be improvable by increasing the BioHash length; however, we demonstrate that the BioHash length is constrained in practice by the amount of memory required for the projection matrix. So, WLD finger vein patterns are found to be the most promising for BioHashing purposes due to their relatively small feature vector size, which allows us to generate larger BioHashes than is possible for RLT or MC feature vectors. In addition, we also provide an open-source implementation of a BioHash-protected finger vein verification system based on the WLD, RLT and MC extractors, so that other researchers can verify our findings and build upon our work.

Keywords BioHashing · Finger veins · Biometric template protection · Wide Line Detector · Repeated Line Tracking · Maximum Curvature · EU General Data Protection Regulation (GDPR) · UTFVP

15.1 Introduction

As our world is transforming into an interconnected network of individuals and devices, we are beginning to realise that current data protection mechanisms are

V. Krivokuća (✉) · S. Marcel
Idiap Research Institute, Martigny, Switzerland
e-mail: vedrana.krivokuca@idiap.ch

S. Marcel
e-mail: sebastien.marcel@idiap.ch

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_15

becoming inadequate to meet our growing security needs. Traditional security mechanisms, such as passwords and access cards, are no longer sufficient for establishing an individual's true identity, which is why we are turning to biometrics for stronger identity assurance. While the unique link between an individual and their biometric characteristics is the very fact that makes biometric authentication so reliable, it is this same aspect of biometrics that makes this authentication factor vulnerable. For this reason, the past decade has seen the emergence of a new field of research into developing effective biometric template protection strategies to secure biometric features during storage and transmission in an authentication system.[1] Research in this area is particularly important in light of the recent EU General Data Protection Regulation (GDPR),[2] which legally obliges users of biometric data to exercise caution in processing and storing this data to protect individuals' digital identities.

A recent review paper on biometric template protection by Sandhya and Prasad [1] shows that, between the years 2005 to 2016, the smallest amount of effort has been invested into developing protection mechanisms for finger veins. Nevertheless, finger vein recognition has increased in popularity over the past few years, with several companies having already deployed finger vein recognition systems for public use, e.g. M2SYS, Idemia, Hitachi and NEC. This suggests that there is an urgent need to direct our attention towards researching effective mechanisms for protecting finger vein templates.

Although the finger vein template protection field is still in its infancy, a number of methods have been proposed in the literature. For example, in one of the earliest approaches towards finger vein template protection [2], the finger vein pattern image is first transformed using the Number Theoretic Transform,[3] after which the transformed template is masked by a random filter. Image-based transformations are also applied towards protecting the finger vein template in [3], where block re-mapping and mesh warping are (separately) applied to the finger vein image to derive two versions of a cancellable finger vein template. Random projection is the template protection method of choice in [4], where the finger vein template consists of end points and intersections. Hybrid template protection strategies have been proposed for finger veins in [5, 6]. In [5], the finger vein image is first transformed into a template where the number of black (background) and white (vein) pixels is approximately equal, and then the Fuzzy Commitment scheme is applied to this template. In [6], the authors propose generating two BioHashes from the same finger vein template, then encrypting one BioHash using Fuzzy Commitment and the other using Fuzzy Vault, after which the two encrypted BioHashes are combined. Finally, [7–9] have focused on multi-biometric systems. More specifically, in [7], finger vein, fingerprint, finger knuckle print and finger shape features are fused, and then the

[1] https://www.iso.org/standard/52946.html.

[2] https://ec.europa.eu/commission/priorities/justice-and-fundamental-rights/data-protection/2018-reform-eu-data-protection-rules_en.

[3] This is essentially the Fourier transform, constrained to a finite field.

resulting feature vector is secured via Fuzzy Commitment. A similar approach is presented in [8], except here the authors also consider score-level and decision-level fusion, whereby Fuzzy Commitment is used to secure each individual feature vector, then the scores or decisions, respectively, of the resulting biometric cryptosystems are fused. In [9], the finger vein feature vector is protected using the Bloom filter approach, and the authors also investigate a multi-biometric system whereby the Bloom filter-protected finger vein template is fused with a Bloom filter-protected face template.

This chapter contributes towards research on finger vein template protection by investigating whether the BioHashing template protection strategy [10] is suitable for protecting finger vein templates, in terms of its effect on the recognition performance of the underlying recognition system. BioHashing is one of the most widely studied biometric template protection schemes in the literature. It involves the projection of a biometric feature vector into a random subspace defined by a user-specific seed, followed by binarisation of the resulting projected vector to produce a so-called BioHash. Although BioHashing has been applied to a number of biometric characteristics (e.g. fingerprints [10], face [11], palm prints [12], and iris [13]), the only mention of BioHashing on finger vein templates that we have come across is the BioHashing/Fuzzy Vault and BioHashing/Fuzzy Commitment hybrid scheme in [6], mentioned earlier. To the best of our knowledge, there does not yet exist any published research on applying BioHashing on its own to finger vein templates. This is where our contribution lies. We also provide an open-source BioHash-protected finger vein verification system, which can be used by other researchers to verify and build upon our work.

We have chosen to focus on BioHashing for three main reasons. First, one of the biggest and most well-known advantages of BioHashing is that, theoretically, there is the possibility of achieving a 0% error rate. While low error rates may be characteristic of two-factor template protection schemes in general, BioHashing is currently the most popular in this category. Second, finger vein images tend to be fairly large, so we were interested in seeing whether BioHashing could be used to produce significantly smaller finger vein templates. Finally, since BioHashing is one of the most well-known template protection schemes in the literature, we wished to provide an open-source implementation of this method for comparison purposes against other template protection techniques developed for finger vein templates.

Note that the new standard[4] for the evaluation of biometric template protection schemes, ISO/IEC 30136:2018, specifies a number of requirements that should be considered when assessing the robustness of a biometric template protection scheme. These include the recognition performance of a biometric system employing template protection compared to that of the same system without template protection; the irreversibility of a template protection scheme, which refers to the difficulty of recovering information about the underlying biometric characteristic from its protected template; diversity, renewability (or cancellability), and unlinkability, all of which relate to the possibility of generating multiple protected templates from

[4] https://www.iso.org/standard/53256.html.

the same biometric characteristic, such that the protected templates are effectively seen as different identities and can thus be used to (i) replace a compromised protected template, and (ii) enroll into multiple applications using the same biometric characteristic without the risk of cross-matching the protected reference templates. The standard also specifies the need to evaluate the possibility of impersonating an enrolled individual using information about their underlying biometric characteristic leaked from one or more of their protected templates, which may largely be attributed to the template protection scheme's compliance with the irreversibility and unlinkability properties. A thorough evaluation of a biometric template protection scheme must, therefore, take into account all of the aforementioned requirements. While the evaluation of recognition performance is relatively established, there are currently no solid, agreed-upon methods for assessing requirements such as irreversibility and diversity/cancellability/unlinkability (despite some guidelines provided by the new standard). Consequently, a thorough evaluation of a biometric template protection scheme necessitates a dedicated treatise of each requirement, which, in many cases, may involve the development and justification of new evaluation methodologies. In light of these reasons, this chapter focuses on evaluating only the recognition performance of BioHash-protected finger vein templates, and we reserve the analysis of the remaining requirements for future work.

The remainder of this chapter is structured as follows. Section 15.2 briefly describes the implementation of our BioHash-protected finger vein verification system. Section 15.3 presents experimental results on the recognition performance of this system and discusses memory constraints that should be considered when applying BioHashing to finger veins. Section 15.4 concludes the chapter and suggests areas for future work.

15.2 BioHash-Protected Finger Vein Verification System

Our BioHash-protected finger vein verification system[5] is an adaptation of the baseline finger vein verification system implemented in the PyPI package.[6] Our adapted system consists of four modules, as illustrated in Fig. 15.1.

The *preprocessor* locates, crops and horizontally aligns the finger in each finger vein image, as per [14, 15].

The *extractor* extracts the vein pattern from the cropped finger image. We used three well-known extractors: Wide Line Detector (WLD) [15], Repeated Line Tracking (RLT) [16] and Maximum Curvature (MC) [17]. The output of each extractor is a binary image, in which white pixels represent the finger vein pattern and black pixels represent the background. For each binary image, we then concatenate its rows to generate a finger vein feature vector.

[5]Code available at the following link: https://gitlab.idiap.ch/bob/bob.chapter.fingerveins_biohashing.

[6]https://pypi.python.org/pypi/bob.bio.vein.

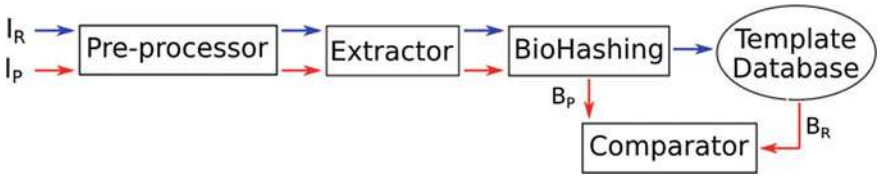


Fig. 15.1 Enrolment (blue arrows) and verification (red arrows) stages in our BioHash-protected finger vein verification system. I_R and I_P denote the reference and probe finger images, respectively. Similarly, B_R and B_P denote the reference and probe BioHashes, respectively

The finger vein feature vector obtained from the feature extraction stage is next *BioHashed*. Our implementation is based on the original BioHash method proposed in [10]. The steps are summarised below:

1. Generate a user-specific[7] random projection matrix of size $n \times l$ for each unique finger[8] in the database, where n represents the dimensionality of the finger vein feature vector and l denotes the desired BioHash length. To ensure that the same matrix can be generated for a specific finger during every verification attempt, the random matrix generation is seeded with a user-specific *seed*. (This seed should be stored on an external token, separately from the BioHash.)
2. Orthonormalise the random matrix.
3. Compute the dot product between the finger vein feature vector and each column of the orthonormalised random matrix. The result is an l-dimensional projected vector.
4. Binarise the projected vector using the mean of the vector as the binarisation threshold, such that all values greater than the mean are set to 1 and all values less than or equal to the mean are set to 0. The result is an l-dimensional binary vector, referred to as the "BioHash".

For the unprotected (without BioHashing) templates in our baseline finger vein verification system, *comparison* is performed on the extracted finger vein features separately for each of the three extractors (WLD, RLT and MC), using the comparison algorithm proposed in [16]. This method is based on a cross-correlation between the enrolled (reference) finger vein template and the probe template obtained during verification. For the protected (with BioHashing) templates in our BioHash-protected finger vein verification system, comparison is done by computing the Hamming distance between the reference and probe BioHashes.

[7]Note that "user" refers to an individual using the finger vein verification system. While the standardised term would be "biometric data subject" or "individual", we have chosen to retain the term "user" for consistency with [10].

[8]Each finger represents a different identity or "user".

15.3 Recognition Performance of BioHash-Protected Finger Vein Verification System

This section presents the results of the experiments we conducted to determine the recognition performance of our BioHash-protected finger vein verification system.

For the experiments reported in this paper, we employed the publicly available finger vein database UTFVP.[9] This database consists of four images for each of 60 subjects' left and right index, ring and middle fingers, which makes up 1,440 images in total. Each image has a height of 380 pixels and a width of 672 pixels. Associated with the database are a number of different evaluation protocols. We used the "nom" protocol,[10] for which the database is split into three sets ("world", "dev", and "eval"). We employed the "eval" set, which consists of fingers 29–60. The comparison protocol involved using the first two finger vein images from each finger for enrolment and the last two as probes.

We chose this database for two reasons. First, it is publicly available, which means that our results can be easily verified by other researchers. Second, it has been shown [18] that an EER of as low as 0.4% is achievable on this database, so we wanted to investigate the effects of BioHashing on such remarkable recognition performance.

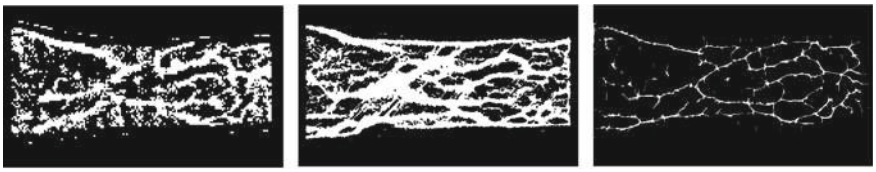
15.3.1 Baseline Recognition Performance

To determine how effective our BioHash-protected finger vein verification system is for finger verification purposes, it was necessary to first establish the recognition performance of our baseline verification system, i.e. using unprotected finger vein features. We had three baselines, one for each of the three extractors.

Figure 15.2 illustrates the outputs of each of the three feature extractors on a finger image from UTFVP, and Table 15.1 shows the dimensionalities of the finger vein feature vectors from each extractor. Although the images in Fig. 15.2 have all been scaled to the same size for easier visual comparison of the extracted patterns, the three extractors actually produce images of different sizes, as is evident from Table 15.1. The MC extractor is the only one that outputs a binary image of the same size as the original image from the database, plus a little extra background padding for comparison purposes. On the other hand, both the WLD and RLT extractors output binary images that are much smaller than the original image. Our adopted WLD extractor reduces the image to a quarter of its original size in each dimension prior to feature extraction to speed up the processing, and the RLT extractor reduces each dimension of the image to a third of its original size. These dimensionalities will be shown to play an important role in the practical feasibility of applying BioHashing to finger vein patterns, a point which will be discussed further in Sect. 15.3.3.

[9] http://scs.ewi.utwente.nl/downloads/show,Finger%20Vein/.

[10] Defined by Idiap Research Institute. See https://www.beat-eu.org/platform/databases/utfvp/1/ for more details.



(a)WLD (b)RLT (c)MC

Fig. 15.2 Finger vein patterns extracted using three different feature extractors on the same finger image from UTFVP

Table 15.1 Sizes of the extracted binary finger vein pattern images and corresponding finger vein feature vectors

Extractor	Image size (pixels)	Feature vector dimensionality
WLD	94×164	15,416
RLT	234×409	95,706
MC	390×682	265,980

Figure 15.3 presents a visual comparison of the recognition performance of the three extractors in terms of Receiver Operating Characteristic (ROC) plots. We refer to this as the *baseline* recognition performance (i.e. the performance of the finger vein recognition systems prior to incorporating BioHashing).

Considering the recognition performance of the three extractor baselines in Fig. 15.3, it is evident that the MC extractor has the best performance. Looking at Fig. 15.2, this makes sense, because the MC extractor seems to produce the cleanest, thinnest finger vein patterns, which would be expected to contribute to more accurate recognition. The fact that the recognition performance of the WLD and RLT extractors is very similar may be attributed to the fact that the two extractors produce finger vein patterns of similar quality (thick, with a fairly noisy background), even

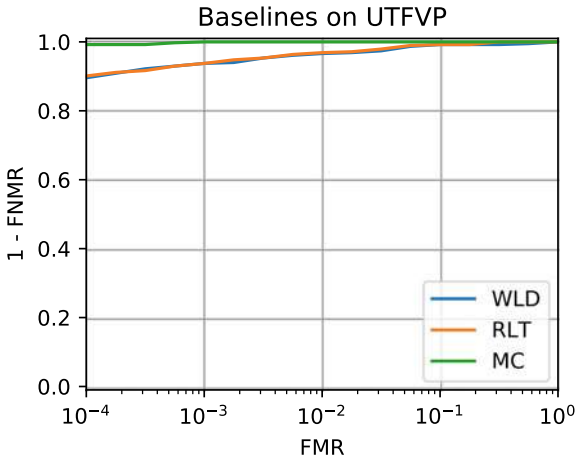


Fig. 15.3 Comparing baseline ROCs across the three feature extractors on the UTFVP database

though the RLT-extracted pattern in Fig. 15.2 appears cleaner than the WLD-extracted pattern.

15.3.2 BioHashing Recognition Performance

This section presents experimental results on the recognition performance of our BioHash-protected finger vein verification system. We consider two scenarios: the Normal scenario and the Stolen Token scenario. The Normal scenario refers to the scenario where each user of the verification system employs their own secret seed and associated random projection matrix in the generation of their BioHash. This is the expected scenario for most cases in practice. The Stolen Token scenario refers to the scenario where a genuine user's secret seed is stolen and used with the impostor's own finger vein template to generate the impostor's BioHash. While it is hoped that such a scenario would not occur in practice, the fact that the user-specific seed is a valuable secret means that we must consider the scenario where that secret is leaked.

To determine the recognition performance of our BioHash-protected finger vein verification system in both the Normal and Stolen Token scenarios, we generated BioHashes of lengths $l = \{100, 200, 300, 400, 500\}$ (number of bits) for finger vein feature vectors resulting from each of our three feature extractors (WLD, RLT and MC). For the Normal scenario, the unique ID of the finger image was used as the seed,[11] and for the Stolen Token scenario, the same seed (seed = 100) was used to generate the BioHashes for all fingers. Table 15.2 indicates the dimensionality reduction resulting from applying BioHashing to the finger vein feature vectors (refer to Table 15.1 for the original finger vein feature vector dimensionality). Figure 15.4 shows the recognition performance of the three finger vein extractors in both the Normal and Stolen Token scenarios, in terms of ROC plots.

From Table 15.2, it is evident that generating BioHashes of 100–500 bits from finger vein feature vectors results in a *significant* dimensionality reduction for all three feature extractors. The greatest dimensionality reduction is observed for the MC extractor, and the WLD extractor shows the smallest dimensionality reduction. This makes sense, since MC finger vein feature vectors have the largest dimensionality and WLD finger vein feature vectors the smallest (see Table 15.1). While "dimensionality" does not necessarily equal "information", and thus "dimensionality reduction" does not necessarily imply "information loss", the size of the dimensionality reductions noted in Table 15.2 makes it highly probable that mapping finger vein feature vectors to BioHashes *does* result in some information loss. In particular, from the results in Table 15.2, we would conclude that BioHashing on MC finger vein feature vectors would incur the largest information loss and WLD feature vectors the smallest. This should be evident when comparing the recognition performance of the BioHash-protected finger vein recognition system to the baseline system (i.e. the system without BioHashing). We refer to Fig. 15.4 for this purpose.

[11] In practice, the seed should be randomly generated. We only used the finger ID as the seed so that our results are more easily reproducible.

Table 15.2 Dimensionality reduction (percentage of dimensionality lost) as a result of converting finger vein feature vectors to BioHashes of different lengths (l)

Extractor	$l = 100$	$l = 200$	$l = 300$	$l = 400$	$l = 500$
WLD (%)	99.35	98.70	98.05	97.41	96.76
RLT (%)	99.90	99.79	99.69	99.58	99.48
MC (%)	99.96	99.92	99.89	99.85	99.81

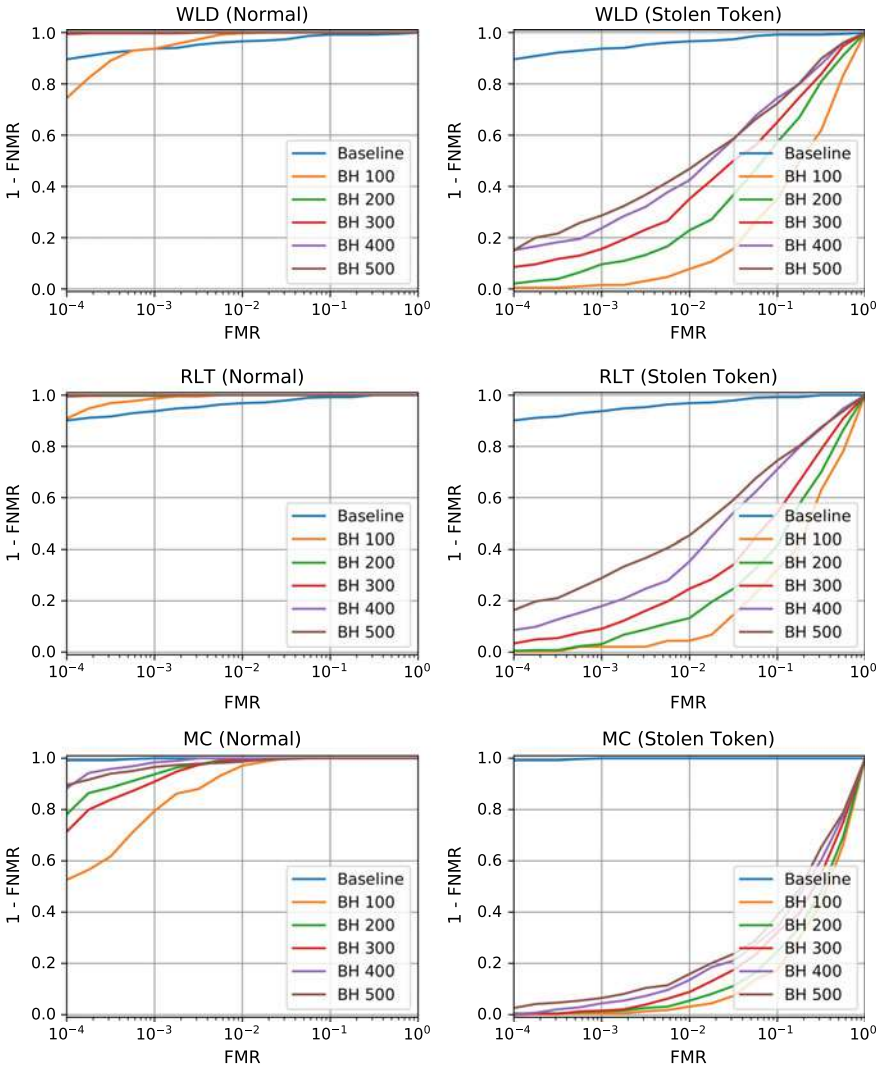


Fig. 15.4 Recognition performance of our BioHash-protected finger vein verification system in the Normal and Stolen Token scenarios

There a number of important observations from Fig. 15.4. First, in the Normal scenario, the BioHash-protected finger vein recognition performance for the WLD and RLT extractors is generally better than the baseline and has an error rate of approximately 0% at all FMR values, for $l > 100$. This is interesting, since the BioHashes are significantly smaller than the original finger vein feature vectors, as noted in Table 15.2. However, the additional entropy introduced by the user-specific projection matrices makes the resulting BioHashes more discriminative than the original finger vein feature vectors, so the superior performance of BioHashes is not surprising. The fact that the BioHashed MC finger vein patterns struggle to reach the baseline recognition performance as quickly as WLD or RLT BioHashes is probably because BioHashing on MC finger vein feature vectors results in the largest dimensionality reduction (see Table 15.2). It is interesting to note, however, that although the dimensionality reduction for both RLT and MC is greater than 99% for all BioHash lengths tested (refer to Table 15.2), RLT BioHashes perform much better than MC BioHashes. So, perhaps such a large dimensionality reduction is too severe for MC finger vein patterns. Nevertheless, we can see that the recognition performance improves as the BioHash length increases, and for all three extractors, the Normal scenario recognition performance in the BioHashed domain equalises or surpasses the baseline recognition performance as the FMR approaches 10^{-1}.

As for the Stolen Token scenario, from Fig. 15.4 we can see that the recognition performance for all three extractors is significantly worse than the baseline. Such a trend has been shown for other biometric characteristics in the literature (e.g. [19]), and it makes sense because in the Stolen Token scenario we are essentially performing a huge dimensionality reduction using the same projection matrix for each finger.[12] So, here we see the "real" effect (i.e. without the additional entropy introduced by the user-specific projection matrix in the Normal scenario) of the significant dimensionality reduction reported in Table 15.2. Since we cannot, in general, expect better recognition performance than the baseline when the dimensionality of our feature vectors is reduced via random projection, the best we can hope for is that the performance of our BioHash-protected finger vein verification system in the Stolen Token scenario is as close as possible to our baseline. From Fig. 15.4, we can see that, as in the Normal scenario, the recognition performance in the Stolen Token scenario approaches that of the baseline as the BioHash length increases.

If we were to rank our three extractors in the Normal scenario based on Fig. 15.4, we would place WLD and RLT first equal, followed by MC. This is an interesting turn of events, since the baseline ranking in Fig. 15.3 is the opposite. Our suspicion is that this is due to the thinness of the finger veins extracted by MC, which means that the MC feature vector may need a much higher resolution than the WLD or RLT feature vectors. So, a BioHash in the range of 100–500 bits might just be too small to represent the MC features.

Ranking the three extractors in the Stolen Token scenario, once again MC takes last place, with WLD and RLT fighting for first. It seems as if WLD has slightly better recognition performance than RLT for all but a BioHash length of 500, where

[12] Recall that each finger corresponds to a different identity.

RLT marginally takes over. We would expect that the smallest feature vector, that produced by WLD, would incur the smallest information loss as a result of the smallest dimensionality reduction in the projection to a 100–500 bit BioHash, while the greatest information loss would be incurred by the largest feature vector, that produced by MC. So, we would predict that the WLD extractor recognition performance would be closest to its baseline and MC furthest from its baseline in the Stolen Token scenario. This is, more or less, what we observe in Fig. 15.4.

If we had to draw a conclusion about the suitability of applying BioHashing to a finger vein verification system based on the recognition performance observed in Fig. 15.4 alone, we would probably have to say that BioHashing is *not* a suitable template protection scheme in this case. While we would assume that the system would operate in the Normal scenario most of the time, in which case BioHashing would be great for achieving a 0% error rate with the WLD or RLT feature extractors (or even the MC extractor, depending on what FMR the system needs to operate at), unfortunately we cannot ignore the possibility of the Stolen Token scenario. Since the recognition performance of all three extractors in the Stolen Token scenario is significantly worse than the baseline for the BioHash lengths tested, it seems too risky to recommend incorporating BioHashing into a finger vein verification system.

However, we have observed that the recognition performance of the BioHash-protected finger vein verification system improves as the BioHash length increases. So, this brings to mind a possible solution: Why not just try larger lengths? We discuss this point in Sect. 15.3.3.

15.3.3 Memory Constraints

This section investigates the possibility of increasing the BioHash length to gain better recognition performance for our BioHash-protected finger vein verification system in the Stolen Token scenario. Since we know that, theoretically, we cannot achieve better recognition performance than the baseline in the Stolen Token scenario, our first approach might be to choose the MC extractor, since Fig. 15.3 shows that it has the best baseline out of the three extractors tested. Even though the recognition performance of the BioHashed MC finger vein features in Fig. 15.4 was shown to be worse than the performance of the WLD and RLT features, our hope might be that if we choose a large enough BioHash length then perhaps it would be possible to push the performance of our BioHashed MC features up to the MC baseline performance. The question is, how large would this BioHash need to be in order for us to achieve such an improvement in the recognition performance?

Figure 15.5 shows a plot of the amount of memory required, in bytes, to generate the projection matrix for a single feature vector for each of our three extractors, as the BioHash length increases from 100 to 2,000. Remember that the projection matrix consists of n rows by l columns, where n denotes the number of bits in the binary feature vector (see Table 15.1) and l represents the BioHash length.

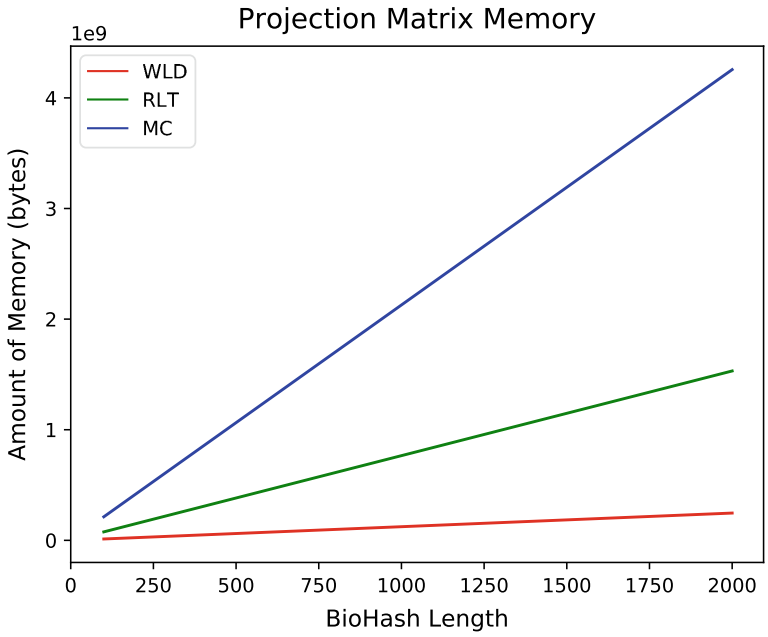


Fig. 15.5 Amount of memory required for the projection matrix as the BioHash length increases. Note that memory ranges from 0 to just over 4 GB in this plot

From Fig. 15.5, we can see that the amount of memory required for a projection matrix corresponding to a WLD feature vector grows quite gradually as the BioHash length increases, that for an RLT feature vector grows faster, and that for an MC feature vector the fastest. For example, it seems that for a 1,000-bit BioHash we would require less than 0.1 GB for a WLD projection matrix, about 0.75 GB for RLT, and over 2 GB for MC! This immediately suggests that anything close to or larger than a 1,000-bit BioHash would probably be impractical for MC features, possibly doable for RLT features but not for a much larger l, and manageable for larger BioHashes on WLD features.

We attempted 1,000-bit BioHashes for our three extractors. As expected, the result was a memory error for our MC feature vectors (i.e. insufficient memory available). This confirms our suspicion that, although MC has the best baseline, it may be impractical for BioHashing. We might consider re-scaling the MC-extracted finger vein pattern image so that we have a smaller feature vector to work with, but this is currently not a characteristic of our adopted MC extractor implementation. As for the WLD and RLT extractors, Fig. 15.6 compares their recognition performance on 1,000-bit BioHashes in the Stolen Token scenario (note that both extractors had an error rate of 0% in the Normal scenario, so this is not shown).

As expected from the Stolen Token plots in Fig. 15.4, the recognition performance of the two extractors in Fig. 15.6 is fairly close, with RLT doing slightly better at the larger BioHash length. Overall, however, this recognition performance may still be

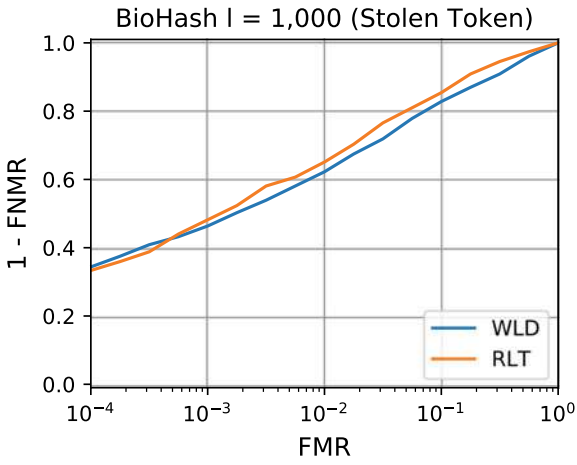


Fig. 15.6 WLD versus RLT when BioHash length is 1,000

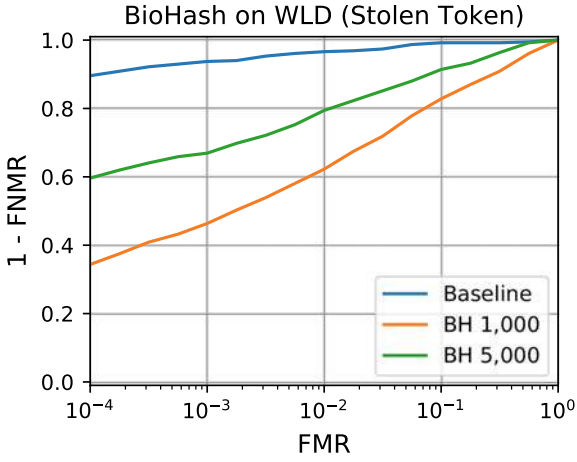


Fig. 15.7 1,000-bit versus 5,000-bit BioHashes on WLD compared to the baseline recognition performance

impractically low, so we might need to consider an even larger BioHash length to try to improve the performance.

We attempted a BioHash length of 5,000 for our WLD and RLT features. As expected, the RLT-based BioHash generation resulted in a memory error. This means that, with our current implementation of the RLT extractor, we cannot expect to gain a significant improvement in the recognition performance of RLT-based BioHashes in the Stolen Token scenario. The WLD-based BioHashes, on the other hand, had no memory issues. Figure 15.7 compares the recognition performance of our BioHash-protected finger vein verification system for 1,000-bit and 5,000-bit BioHashes on the WLD finger vein features in the Stolen Token scenario to the WLD baseline (note that both BioHash lengths had an error rate of 0% in the Normal scenario, so this is not shown).

Figure 15.7 confirms our previously observed trend (in Fig. 15.4) that the recognition performance of our WLD-based BioHash-protected finger vein verification

system approaches the performance of the corresponding baseline in the Stolen Token scenario as the BioHash length increases. The final length will depend on how much of a drop in recognition performance is acceptable in the Stolen Token scenario. Technically, we can expect the BioHash recognition performance to be approximately the same as the baseline performance when the BioHash length is the same as the length of the original feature vector. The issue here is that, in this case, the BioHash is more or less fully invertible, meaning that it would be possible to recover the original feature vector if the user's secret seed and thus their projection matrix is leaked to an attacker. So, it is important to try to find a large enough BioHash length to ensure we have reasonable recognition performance in both the Normal and Stolen Token scenarios, while keeping the length small enough to ensure that the resulting BioHash is sufficiently privacy-preserving. The privacy-preserving properties of our BioHash-protected finger vein verification system must be investigated before we can fully justify any conclusions on whether or not BioHashing is a suitable template protection scheme for finger veins.

15.4 Conclusions and Future Work

This chapter presented the first investigation into the suitability of BioHashing as a finger vein template protection scheme for finger vein verification systems based on three feature extractors (WLD, RLT and MC), in terms of recognition performance only. Our experiments showed that, in the Normal scenario, it is possible to achieve a 0% error rate for BioHashes that are significantly smaller than the original finger vein feature vectors. BioHashes generated from WLD and RLT finger vein feature vectors were found to perform the best, while BioHashed MC features were shown to approach the baseline recognition performance as the FMR approached 10^{-1}. As expected, the recognition performance for all three extractors was worse than the baseline in the Stolen Token scenario due to the huge dimensionality reduction that is incurred in projecting a finger vein feature vector to a relatively small BioHash. While the recognition performance was shown to improve by increasing the length of the BioHash vectors, it was also demonstrated that the choice of length is constrained in practice by the amount of memory required for the projection matrix. Consequently, the WLD extractor was found to be the most promising for BioHashing purposes, since the relatively small size of WLD feature vectors allows for much larger BioHashes than would be possible for RLT or MC feature vectors. One issue with generating large BioHashes, however, is that, the larger the BioHash length, the easier it becomes to invert the BioHash to recover the original feature vector, thereby jeopardising the privacy of the verification system's users. To determine an optimal BioHash length that would ensure a reasonable balance between recognition performance and privacy preservation, we would need to conduct a full security and privacy analysis for the BioHashed WLD finger vein patterns. This will form part of our future work. Another area for future work could be to investigate the effect on BioHashing recognition performance when the three extractors are modified to produce feature vectors of the same size.

Acknowledgements The authors would like to acknowledge the following sources of funding for supporting this work: the Secure Access Control over Wide Area Network (SWAN) project and the Swiss Center for Biometrics Research and Testing.

References

1. Sandhya M, Prasad MVNK (2017) Biometric template protection: a systematic literature review of approaches and modalities. Springer International Publishing, Cham, pp 323–370
2. Hirata S, Takahashi K (2009) Cancelable biometrics with perfect secrecy for correlation-based matching. Springer, Berlin, Heidelberg, pp 868–878
3. Piciucco E, Maiorana E, Kauba C, Uhl A, Campisi P (2016) Cancelable biometrics for finger vein recognition. In: 2016 First international workshop on sensing, processing and learning for intelligent machines (SPLINE), July 2016, pp 1–5
4. Liu Y, Ling J, Liu Z, Shen J, Gao C (2017) Finger vein secure biometric template generation based on deep learning. Soft Comput 1–9
5. Favre M, Picard S, Bringer J, Chabanne H (2015) Balancing is the key: performing finger vein template protection using fuzzy commitment. In: 2015 International conference on information systems security and privacy (ICISSP), Feb 2015, pp 1–8
6. Yang W, Hu J, Wang S (2013) A finger-vein based cancellable bio-cryptosystem. Springer, Berlin, Heidelberg, pp 784–790
7. Lu L, Peng J (2014) Finger multi-biometric cryptosystem using feature-level fusion. Int J Signal Process Image Process Pattern Recognit 7(3):223–236
8. Peng J, Li Q, Abd El-Latif AA, Niu X (2014) Finger multibiometric cryptosystems: fusion strategy and template security. J Electron Imaging 23(2):023001
9. Gomez-Barrero M, Rathgeb C, Li G, Ramachandra R, Galbally J, Busch C (2018) Multi-biometric template protection based on bloom filters. Inf Fusion 42:37–50
10. Jin ATB, Ling DNC, Goh A (2004) Biohashing: two factor authentication featuring fingerprint data and tokenised random number. Pattern Recognit 37(11):2245–2255
11. Goh A, Ngo DCL (2003) Computation of cryptographic keys from face biometrics, pp 1–13. Springer, Berlin, Heidelberg
12. Connie T, Teoh A, Goh M, Ngo D (2005) Palmhashing: a novel approach for cancelable biometrics. Inf Process Lett 93(1):1–5
13. Chin CS, Jin ATB, Ling DNC (2006) High security iris verification system based on random secret integration. Comput Vis Image Underst 102(2):169–177
14. Lee EC, Lee HC, Park KR (2009) Finger vein recognition using minutia-based alignment and local binary pattern-based feature extraction. Int J Imaging Syst Technol 19(3):179–186
15. Huang B, Dai Y, Li R, Tang D, Li W (2010) Finger-vein authentication based on wide line detector and pattern normalization. In: 2010 20th International conference on pattern recognition, Aug 2010, pp 1269–1272
16. Miura N, Nagasaka A, Miyatake T (2004) Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach Vis Appl 15(4):194–203
17. Miura N, Nagasaka A, Miyatake T (2007) Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans Inf Syst 90(8):1185–1194
18. Ton BT, Veldhuis RNJ (2013) A high quality finger vascular pattern dataset collected using a custom designed capturing device. In: 2013 International conference on biometrics (ICB), June 2013, pp 1–5
19. Kong A, Cheung KH, Zhang D, Kamel M, You J (2006) An analysis of biohashing and its variants. Pattern Recognit 39(7):1359–1368

Chapter 16
Cancellable Biometrics for Finger Vein Recognition—Application in the Feature Domain

Simon Kirchgasser, Christof Kauba and Andreas Uhl

Abstract Privacy preservation is a key issue that has to be addressed in biometric recognition systems. Template protection schemes are a suitable way to tackle this task. Various template protection approaches originally proposed for other biometric modalities have been adopted to the domain of vascular pattern recognition. Cancellable biometrics are one class of these schemes. In this chapter, several cancellable biometrics methods like block re-mapping and block warping are applied in the feature domain. The results are compared to previous results obtained by the use of the same methods in the image domain regarding recognition performance, unlinkability and the level of privacy protection. The experiments are conducted using several well-established finger vein recognition systems on two publicly available datasets. Furthermore, an analysis regarding subject- versus system-dependent keys in terms of security and recognition performance is done.

Keywords Finger vein recognition · Template protection · Cancellable biometrics · Biometric performance evaluation · Block re-mapping · Warping

16.1 Introduction

Various methods exist to protect the subject-specific information contained in biometric samples and/or templates. According to several studies, e.g. Maltoni et al. [16], and ISO/IEC Standard 24745 [7] each method should exhibit four properties: *Security*, *Diversity*, *Revocability* and *Performance*. These shall ensure that the capture subject's privacy is protected and at the same time a stable and sufficient recognition

S. Kirchgasser · C. Kauba · A. Uhl (✉)
Department of Computer Sciences, University of Salzburg, Jakob-Haringer-Str. 2, 5020 Salzburg, Austria
e-mail: uhl@cs.sbg.ac.at

S. Kirchgasser
e-mail: skirch@cs.sbg.ac.at

C. Kauba
e-mail: ckauba@cs.sbg.ac.at

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_16

performance during the authentication process is achieved. The first aspect deals with the computational hardness to derive the original biometric template from the protected one (*security-irreversability*). *Diversity* is related to the privacy enhancement aspect and should ensure that the secured templates cannot be matched across different databases (*unlinkability*). The third aspect, *revocability*, should ensure that a compromised template can be revoked without exposing the biometric information, i.e. the original biometric trait/template remains unaltered and is not compromised. After removing the compromised data, a new template representing the same biometric instance can be generated. Finally, applying a certain protection scheme should not lead to a significant recognition performance degradation of the whole recognition system (*performance*).

One possibility to secure biometric information, cancellable biometrics, are introduced and evaluated on face and fingerprint data by Ratha et al. in [22]. The applied template protection schemes, *block re-mapping* and *warping*, have also been applied in the image domain and evaluated on iris [5, 14] and finger vein [20] datasets, respectively. Opposed to the latter study we want to investigate these schemes not in the image domain, but in the feature domain as several advantages and disadvantages exist in both spaces. These positive and negative aspects will be described in Sect. 16.2.

A detailed discussion on finger vein related template protection schemes, that can be found in literature, is given in Chap. 1 [26]. Thus, the interested reader is referred to this part of the handbook.

The rest of this chapter is organised as follows: The considered experimental questions are discussed in Sects. 16.2, 16.3 and 16.4 respectively. The employed non-invertible transform techniques are described in Sect. 16.5. Section 16.6 introduces the datasets utilised during the experimental evaluation, the finger vein recognition tool-chain as well as the evaluation protocol. The performance and unlinkability evaluation results are given and discussed in Sect. 16.7. Section 16.8 concludes this chapter and gives an outlook on future work.

16.2 Application in the Feature or Image Domain

If a template protection scheme is applied in the image/signal domain immediately after the image acquisition, the main advantage is that the biometric features extracted from the transformed sample do not correspond to those features computed from the original image/signal. So, the "real" template is never computed and does occur at no stage in the system and further, the sample is never processed in the system except at the sensor device. This provides the highest level of privacy protection for the capture subject. The main disadvantage of the application in the image/signal domain is that the feature extraction based on the protected image/signal might lead to incorrect features and thus, to inferior recognition performance. Especially in finger vein recognition, most of the well-established feature extraction schemes rely on tracking the vein lines, e.g. based on curvature information. By applying

template protection methods like block re-mapping in the image domain right after the sample is captured, connected vein structures will become disconnected. These veins are then no longer detected as continuous vein segments which potentially causes problems during the feature extraction and might lead to an incomplete or faulty feature representation of the captured image. Consequently, the recognition performance of the whole biometric system can be negatively influenced by the application of the template protection scheme.

On the contrary, if template protection is conducted in the feature domain, the feature extraction is finished prior to the application of the template protection approach. Thus, the extracted feature vector or template is not influenced by the template protection scheme at this stage and represents the biometric information of the capture subject in an optimal way.

16.3 Key Selection: Subject- Versus System-Specific

There are two different types of key selection, subject- and system-specific keys. In the subject-specific key approach, the template of each subject is generated by a key which is specific for each subject while for a system-specific key, the templates of all subjects are generated by the same key.

Subject dependent keys have advantages in terms of preserving the capture subjects' privacy compared to system-dependent keys. Assigning an individual key to each capture subject ensures that if an adversary gets to know the key of one of the capture subjects, he can not compromise the entire database as each key is individual. A capture subject-specific key also ensures that insider attacks performed by legitimate registered subjects can not be performed straight forward. Such an attack involves a registered capture subject, who is been granted access to the system and has access to the template database as well. This adversary capture subject wants to be legitimated as one of the other capture subjects of the same biometric system. So he/she could just try to copy one of his/her templates over the template belonging to another capture subject and claim that this is his/her identity, thus trying to get authenticated as this other, genuine capture subject. If capture subject-specific keys are used, this is not easily possible as each of the templates stored in the database has been generated using an individual key. However, it remains questionable if such an insider attack is a likely one. In fact, it would probably be easier for an advisory who has access to the entire template database to simply create and store a new genuine capture subject that exhibits his/her biometric information together with a key he sets in order to get the legitimation he wants to acquire. Another advantage of capture subject-specific keys is that the system's recognition performance in enhanced by introducing more inter-subject variabilities and thus impacting the performed impostor comparisons. The additional variability introduced by the subject-specific key in combination with the differences between different biometric capture subjects leads to a better separation of genuine and impostor pairs which enhances the overall system's performance.

One drawback of using capture subject-specific keys is that the system design gets more complex, depending on how the capture subject-specific keys are generated and stored. In contrast to a system-specific key, which is valid for all capture subjects and throughout all components of the biometric recognition system, the individual capture subject-specific keys have to be generated and/or stored somehow. One possibility is to generate the key based on the capture subject's biometric trait every time the capture subject wants to get authenticated. This methodology refers to the basic idea of Biometric Cryptosystems (BCS), which have originally been developed for either securing a cryptographic key applying biometric features or generating a cryptographic key based on the biometric features [9]. Thus, the objective to employ a BCS is different but the underlying concept is similar to the one described earlier. The second option can be used to generate the capture subject specific key once and then store this key which is later retrieved during the authentication process. This key can either be stored in a separate key database or with the capture subject itself. Storing the keys in a key database of course poses the risk of the key database getting attacked and eventually the keys getting disclosed to an adversary. Storing the keys with the capture subject is the better option in terms of key security, however it lowers the convenience of the whole system from the capture subjects' perspective as they have to be aware of their key, either by remembering the key or by using smart cards or similar key storage devices.

16.4 Unlinkability

The ISO/IEC Standard 24745 [7] defines that irreversibility is not sufficient for protected templates, as they also need to be unlinkable. Unlinkability guarantees that stored and protected biometric information can not be linked across various different applications or databases. The standard defines templates to be *fully linkable* if a method exists which is able to decide if two templates protected using a different key were extracted from the same biometric sample with a certainty of 100%. The degree of linkability depends on the certainty of the method which decides if two protected templates originate from the same capture subject. However, the standard only defines what unlinkability means but gives no generic way of quantifying it. Gomez-Barrero et al. [4] present a universal framework to evaluate the unlinkability of a biometric template protection system based on the comparison scores. They proposed the so-called D_{sys} measurement as a global measure to evaluate a given biometric recognition and template protection system. Further details are given in Sect. 16.6.3 where the experimental protocol is introduced.

The application of the proposed framework [4] allows a comparison to previous work done on the aspect of key-sensitivity using the same protection schemes by Piciucco et al. [20]. Protected templates generated from the same biometric data by using different keys should not be comparable. Thus, the authors of [20] used the so-called Renewable Template Matching Rate (RTMR) to prove a low matching rate between templates generated using different keys on both protection schemes.

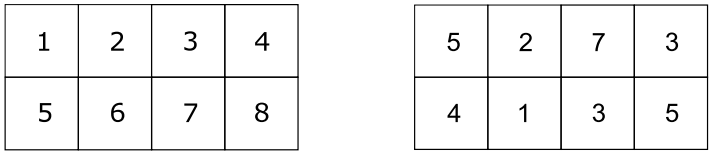
This can also be interpreted as a high amount of unlinkability as the RTMR can be interpreted as a restricted version of the D_{sys} measure.

16.5 Applied Cancellable Biometrics Schemes

The two investigated non-invertible transforms, block re-mapping and warping, are both based on a regular grid. Some variants of them have been investigated and discussed in [21, 22]. The input (regardless if a binary matrix or image) is subdivided into non-overlapping blocks using a predefined block size. The constructed blocks are processed individually, generating an entire protected template or image. As we aim to utilise the same comparison module for the unprotected and protected templates, there is one preliminary condition that must be fulfilled for the selected schemes: The protected template must exhibit a structure similar to the original input template. In particular, we interpret the feature vector (template) as binary image, representing vein patterns as 1s and background information as 0s. Based on this representation, each x-/y-coordinate position (each pixel) in the input image can be either described as background pixel or as vein pattern pixel. Thus, our approach can be used in the signal domain as well as in the feature domain and the template protection performance results obtained in image domain can be directly compared to results obtained in the feature domain. Note that in the signal domain the input as well as the protected output images are no binary but greyscale ones, which does not change the way the underlying cancellable biometrics schemes are applied (as they only change positions of pixels and do not relate single pixel values to each other). In the following, the basic block re-mapping scheme as well as the warping approach are described.

16.5.1 Block Re-mapping

In block re-mapping [22], the number of predefined blocks is separated into two classes, where the total number of blocks remains unaltered. The blocks belonging to the first class are randomly placed at different positions to the ones they have been located in the original input. This random allocation is done by assigning random numbers generated by a number generator according to a predefined key. This key must be stored, such that a new image acquired during authentication can be protected using the same number generator specification. The blocks belonging to the second class are dismissed and do not appear in the output. This aspect ensures the irreversibility property of the block re-mapping scheme. The percentage of blocks belonging to each of the two classes is set by a predefined value. The more blocks in the second class, the less biometric information is present in the output. Usually, the percentage of blocks in the first class is between 1/4 and 3/4 of the total blocks.



(a) Grid with blocks at their original positions.

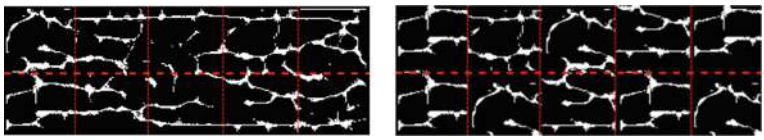
(b) Block positions after re-mapping.

Fig. 16.1 Schematic block re-mapping scheme

Figure 16.1 shows the block re-mapping scheme which has been implemented in a slightly adopted version compared to the original one done by Piciucco et al. [20]. The main difference is the randomised block selection: We introduce an additional parameter, which controls the number of blocks that remain in the transformed template. To enable comparable results, we fixed the number of blocks that remain in the transformed templates to be at 75% of the original blocks. The required key information consists of the two set-up keys for the random generator and the block-size information for the grid construction. By comparing Fig. 16.1 (a) and (b) the following can be observed: While the blocks 4, 6 and 8 are present in (a) they do not occur in the protected, re-mapped image. All the other blocks are used to construct the re-mapped version (b) that has the same size as the original unprotected image or feature representation (a). It also becomes obvious that the blocks 3 and 5 are inserted multiple times into (b) in order to compensate for the absence of the non-considered blocks 6 and 8.

Due to the random selection, it is possible that some blocks are used more than once and others are never used. Otherwise, the re-mapping would resemble a permutation of all blocks, which could be reverted by applying a brute-force-attack testing all possible permutations or some more advanced attacks based on square jigsaw puzzle solver algorithms, e.g. [2, 19, 23].

The bigger the block size, the more biometric information is contained per block and thus, the higher the recognition performance is assumed to be after the application of block re-mapping. Of course, this argument also might depend on the feature extraction and comparison method as well as if it is done in signal or feature domain. Block re-mapping creates discontinuities at the block boundaries which influences the recognition performance if applied in the image domain as several of the feature extraction schemes try to follow continuous vein lines, which are not there any longer. This gets worse with decreasing block sizes. If block re-mapping is applied in the feature domain, this is not an issue as the feature extraction was done prior to applying the block re-mapping. However, due to the shifting process involved during comparison, the re-mapping of blocks can cause problems as a normalised region-of-interest is considered, especially for blocks that are placed at the boundaries of the protected templates. This might eventually lead to a degradation in the biometric systems performance because the information contained in those blocks is then “shifted out” of the image and the vein lines present in the blocks do not



(a) Original template. (b) Re-mapped template.

Fig. 16.2 Finger vein templates displaying the variations that can occur during the re-mapping process using a block size of 64 × 64 pixel and MC as feature extraction method. The red dashed lines represent the grid

contribute to the comparison score anymore. In addition, blocks that share a common vein structure in the original template might be separated after performing the block re-mapping, posing a more severe problem due to the shifting applied during the comparison step. The vein structures close to the block borders are then shifted to completely different positions and cannot be compared any longer, leading to a decrease in the genuine comparison scores. Furthermore, it can also happen that the block re-mapping introduces new vein structures by combining two blocks that originally do not belong to each other. Both of the aforementioned possibilities have a potentially negative influence on the recognition performance. These problems due to the shifting applied during the comparison step are visualised in Fig. 16.2. It clearly can be seen that most of the vein structures visible in the original—left template, are not present in the protected—right template, but other structures have been newly introduced.

On the other hand, the larger the block size, the more of the original biometric information is contained per single block, lowering the level of privacy protection. Hence, we assume that a suitable trade-off between loss of recognition accuracy and level of privacy protection has to be found. Furthermore, the block size also corresponds to the irreversibility property of the transformation. The bigger the block size, the more information is contained per single block and the lower is the total number of blocks. The lower the number of blocks and the higher the information per block, the more effective are potential attacks on this protection scheme as discussed in the literature, e.g. [2, 19, 23].

16.5.2 Block Warping

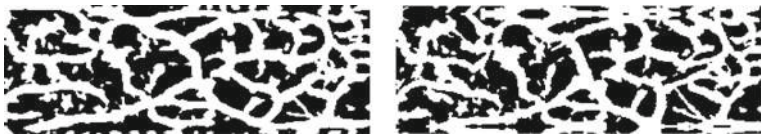
Another non-invertible transformation in the context of cancellable biometrics is the so called "warping" (originally named "mesh warping" [27]). Warping can be applied in the image as well as in the template domain. Using this transformation, a function is applied to each pixel in the image which maps the pixel of the input at a given position to a certain position in the output (can also be the same position as in the input again). Thus, this mapping defines a new image or template containing the same information as the original input but in a distorted representation. The warping

(a) Basis grid, used to separate input information.

(b) Warped grid, built according to the initial grid.

(c) Positions of the blocks after block warping.

Fig. 16.3 Block Warping scheme including resize enhancement displayed schematically



(a) Original template.

(b) Warped template.

Fig. 16.4 Finger vein templates displaying the variations that can occur during the warping process using a block size of 32 × 32 pixel and PC as feature extraction method

approach utilised in this chapter is a combination of using a regular grid, as in the block re-mapping scheme, and a distortion function based on spline interpolation. The regular grid is deformed per each block and adjusted to the warped output grid. The number of blocks in the output is the same as in the input, but the content of each individual block is distorted in the warped output.

This distortion is introduced by randomly altering the edge positions of the regular grid, leading to a non-predictable deformation of the regular grid. Spline based interpolation of the input information/pixels is applied to adopt the area of each block with respect to the smaller or larger block area obtained after the deformation application (warping might either stretch or shrink the area of the block as the edge positions are changed). This distortion is key dependent and the key defines the seed value for the random generator responsible for the replacement of the grid edges. This key needs to be protected by some cryptographic encryption methods and stored in a safe place. However, if the key gets disclosed, it is not possible to reconstruct all of the original biometric data in polynomial time due to the applied spline based interpolation. Figure 16.3 shows the basic warping scheme, while in Fig. 16.4 an example of a original—left template and its protected—right template is given.

The application of interpolation does increase the template protection degree as the relation between original vein structures is distorted. However, these transformations might destroy dependencies between the vein lines which are necessary in the feature extraction step in order to enable the same recognition performance as on the original, unprotected data. On the one hand, the application of warping transformations increases the capture subject's privacy but on the other hand the recognition performance is likely to decrease. For more information about other warping methods, the interested reader is referred to [3], where a review of several different possible solutions including the use of parametric and non-parametric functions can be found.

16.6 Experimental Set-Up

In the following, the experimental set-up, including the datasets, the finger vein recognition tool-chain as well as the experimental protocol are explained.

16.6.1 Finger Vein Datasets

The experiments are conducted on two datasets: The first one is the University of Twente Finger Vascular Pattern Database (UTFVP) [25]. It consists of 1440 images, which were acquired from 60 subjects in a single session. Six fingers were captured, including the index, ring and middle finger of both hands with 4 images per finger. The finger vein images have a resolution of 672 $\times$ 380 pixels and a density of 126 pixels/cm, resulting in a width of 4–20 pixels for the visible blood vessels.

The second dataset we utilise here is the PLUSVein-FV3 Dorsal–Palmar finger vein dataset and which has been introduced in [10] and is partly discussed in Chap. 3 [12]. To enable a meaningful comparison with the UTFVP results, we only use the palmar subset. Region-Of-Interest (ROI) images containing only the centre part of the finger where most of the vein pattern information is located have been extracted from the captured images as well. Some example images of the PLUSVein-FV3 subsets are given in Fig. 16.5.

16.6.2 Finger Vein Recognition Tool-Chain

In this subsection an overview of the most important parts of a typical finger vein recognition tool-chain is given. There are several studies about finger vein recognition

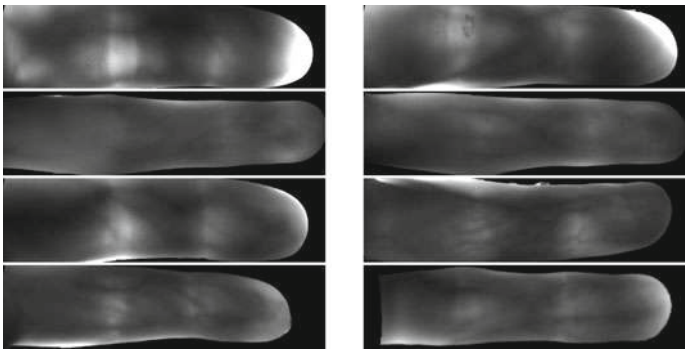


Fig. 16.5 Finger vein images of the PLUSVein-FV3 finger vein laser (first two rows) and LED subset (last two rows) showing 8 different fingers

systems, e.g. [8], that present and discuss different designs, but they all include a few common parts or modules. These main modules consist of: the finger vein scanner (image acquisition), the preprocessing module (preprocessing), the feature extraction module (feature extractor), the template comparison module (matcher) and the decision module (final decision). The system may contain an optional template protection module, either after the preprocessing module (image domain) or after the feature extraction module (feature domain). As the main focus of this chapter is on template protection applied in the feature domain, the system used during the experiments contains the template protection as part of the feature extractor. For feature extraction we selected six different methods: Gabor Filter (GF) [13], Isotropic Undecimated Wavelet Transform (IUWT) [24], Maximum Curvature (MC) [18], Principal Curvature (PC) [1], Repeated Line Tracking (RLT) [17] and Wide Line Detector (WLD) [6].

To calculate the final comparison scores an image correlation based comparison scheme as introduced by Miura et al. in [17] is applied to the baseline (unprotected) templates (features) as well as to the templates protected by block re-mapping and block warping. As the comparison scheme is correlation based, including a necessary pixel wise shifting, we selected a shift range of 80 pixels in x- and 30 pixels in y-direction, respectively. Further details on the deployed recognition tool-chain can be found in Chap. 4 [11] of this handbook.

16.6.3 Experimental Protocol and Types of Experiments

The necessary comparison scores are calculated using the correlation based comparison scheme described before and the comparison to be performed are based on the Fingerprint Verification Contests' (FVC) protocol [15]. To obtain the genuine scores, all possible comparisons are performed, i.e. the number of genuine scores is $60 * 6 * \frac{4*3}{2} = 2160$ (UTFVP) and $60 * 6 * \frac{5*4}{2} = 3600$ (PLUSVein-FV3), respectively. For the impostor scores, only a subset of all possible comparisons is performed. The first image of each finger is compared against the first image of all other fingers. This results in $\frac{60*6*(60*6-1)}{2} = 64{,}620$ impostor comparisons for each dataset (as both of them contain 60 subjects and 6 fingers per capture subject). As the employed comparison scheme is a symmetric measure, no symmetric comparisons (e.g. 1–2 and 2–1) are performed. The FVC protocol reduces the number of impostor comparisons in order to keep the computation time low for the whole performance evaluation while ensuring that every finger is compared against each other finger at least once. To quantify the recognition performance, several well-known measures are utilised: The equal error rate (EER, point where the FMR and the FNMR are equal), FMR_{100} (the lowest false Non-Match Rate (FNMR) for false match rate (FMR) $\leq 1\%$), FMR_{1000} (the lowest FNMR for FMR $\leq 0.1\%$) as well as the *ZeroFMR* (the FNMR for FMR $= 0\%$).

We conduct four sets of experiments:

1. In the first set of experiments the unprotected templates are considered. The first experiments provide a baseline to compare the recognition performance of the protected templates to
2. The second set of experiments deals with the protected templates, generated by applying one of the aforementioned cancellable biometrics schemes to the same templates that have been used during the first set of experiments. For score calculation, these protected templates are compared against each other. For both employed cancellable schemes, 10 runs using different system keys are performed to assess the recognition performance variability and key dependency of the recognition performance.
3. The third set of experiments compares capture subject specific and system-specific keys. Therefore, a different key (note: the key is controlling the random selection of blocks or the repositioning of the grid) is assigned to each finger, thus resulting in 360 virtual subjects (not only the 60 physical ones). Again, 10 runs with different keys per run are performed and averaged afterwards. These capture subject-specific key results are then compared to the system-specific key ones as obtained in the second set of experiments.
4. The last set of experiments is committed to the unlinkability analysis. The approach by Gomez et al. [4], introduced in Sect. 16.4 describes the extent of linkability in the given data, with a range of D_{sys} from [0, 1]. The higher the value, the more linkable are the involved templates. Thus, the resulting measure represents a percentage of linkability that is present. Of course, full unlinkability is given if the score is 0. D_{sys} is based on the local $D(s)$ value, which is calculated based on the comparison scores of several mated (genuine) as well as non-mated (impostor) comparison between templates protected by the same template protection system but using different keys, thus originating from different applications or systems. We utilise this measure to assess the unlinkability of the presented cancellable biometric schemes for finger vein recognition (block re-mapping and warping).

To comply with the principles of reproducible research we provide all experimental details, results as well as the used vein recognition SDK, settings files and scripts to run the experiments for download on our website: http://www.wavelab.at/sources/Kirchgasser19b/. The used datasets are publicly available as well, hence it is possible to reproduce our results for anyone who is interested to do so.

16.6.4 Selecting the Processing Level to Insert Template Protection

If template protection is done in the signal domain cancellable biometrics schemes are applied directly after the image acquisition and before the feature extraction. Otherwise, template protection is applied to the extracted binary vein features in

order to protect the contained private biometric information right after the feature extraction is finished (feature domain).

The main purpose of this chapter and the experiments performed here is to provide a recognition performance comparison to the previous results obtained by Piciucco et al. [20]. The authors used the same cancellable methods on the UTFVP finger vein images, but as opposed to this chapter, not in the feature domain, but in the image domain. To ensure that our results are comparable with the previous ones by Piciucco et al. [20], we use the same block sizes during our experiments and select the same maximum offset for the block warping approach. Thus, we select block sizes of $16 \times 16, 32 \times 32, 48 \times 48$, and 64×64 for block re-mapping and block warping. For block warping, maximum offset values of 6, 12, 18, and 24 pixel are considered. In the following result tables block re-mapping is abbreviated using *remp_16* (block size: 16×16) till *remp_64* (block size: 64×64), while all warping experiments correspond to *warp_16_6* (block size: 16×16, offset: 6) till *warp_64_24* (block size: 64×64, offset: 24).

In contrast to the work of Piciucco et al. [20], we do not perform an analysis of the renewability and the key-sensitivity of the employed cancellable biometrics schemes. The key-sensitivity and renewability are expected to be similar for the schemes applied in the feature domain and in the image domain. Instead, we consider different issues like the comparison of capture subject vs. system-depended keys, and a thorough unlinkability analysis.

16.7 Experimental Results

This section presents and discusses all relevant results concerning the various template protection methods' impact on the recognition performance and unlinkability in the four sets of experiments that have been considered. As we aim to compare the experimental results to the corresponding results reported in [20], we first summarise their main results:

(a) The best performance results regarding EER were found for the block re-mapping scheme using a block size of 64×64.
(b) The best achieved *EER* was 1.67% for the protected data and 1.16% for the unprotected templates of the UTFVP dataset (using GF features).
(c) Block re-mapping outperformed block warping.

16.7.1 Baseline Experiments

Table 16.1 lists the performance results of the baseline experiments in percentage terms for the UTFVP and the PLUSVein-FV3 dataset. Overall, the performance on the UTFVP dataset is slightly superior compared to the PLUSVein-FV3 dataset for most of the evaluated recognition schemes.

Table 16.1 Baseline performance on the UTFVP and PLUSVein-FV3 database in terms of *EER*, FMR_{100}, FMR_{1000} and *ZeroFMR*. The best performing results are highlighted in bold numbers

Features	GF	IUWT	MC	PC	RLT	WLD
	UTFVP					
EER (%)	0.64	0.36	**0.09**	0.14	0.60	0.46
FMR_{100} (%)	0.60	0.27	**0.04**	0.13	0.32	0.27
FMR_{1000} (%)	1.00	0.55	**0.09**	0.13	1.15	0.60
ZeroFMR (%)	1.00	1.34	**0.23**	0.87	1.89	1.43
	PLUSVein-FV3 Laser					
EER(%)	0.74	1.49	**0.33**	1.47	1.71	1.38
FMR_{100}	0.63	1.66	**0.22**	1.50	1.91	1.44
FMR_{1000}	1.47	2.08	**0.44**	2.19	2.52	1.75
ZeroFMR	1.75	2.77	**0.72**	2.75	3.77	1.94
	PLUSVein-FV3 LED					
EER(%)	0.61	0.63	**0.28**	0.35	0.79	0.53
FMR_{100}	0.52	0.52	**0.27**	0.33	0.72	0.52
FMR_{1000}	0.63	0.97	**0.27**	0.38	1.16	0.55
ZeroFMR	1.00	3.05	**0.30**	0.66	1.77	0.69

On the UTFVP, the best recognition performance result with an *EER* of 0.09% is achieved by MC, followed by PC with an *EER* of 0.14%, then IUWT, WLD and RLT while GF has the worst performance with an *EER* of 0.64%. On both subsets of the PLUSVein-FV3 the best results are achieved by using MC as well, with an *EER* of 0.28% and 0.33% on the LED and laser subset, respectively. RLT performed worst compared to the other schemes on both subsets. Nevertheless, each of the evaluated recognition schemes achieves a competitive performance on all of the tested datasets. The other performance figures, i.e. FMR_{100}, FMR_{1000} and *ZeroFMR* are in line with the *EER* values and support the general trend that most of the applied feature extraction methods perform reasonably well on the given data sets using the baseline, unprotected templates.

16.7.2 Set 2—Protected Template Experiments (System Key)

As mentioned before, there are several parameters that have an essential influence on the recognition performance results obtained by applying the different cancellable biometrics schemes.

Table 16.2—feature domain and 16.3—signal domain, respectively, present the *EER* by using the mean ($\bar{x}$) and the standard deviation (σ) for both datasets. These results are calculated by randomly choosing 10 different keys and running the experiments first before the presented results are obtained by calculating $\bar{x}$ and σ of the performed computations.

Table 16.2 Recognition performance results (%) for template protection in the feature domain using system keys. The best performing results for each template protection method are highlighted in bold numbers

tempProt	*EER*											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	13.76	1.58	6.29	0.51	8.41	1.09	8.31	0.56	5.77	0.71	6.71	0.37
remp_32	9.12	2.08	4.85	1.25	4.52	1.46	4.79	1.54	5.06	0.94	4.34	1.34
remp_48	7.18	4.20	4.07	2.33	3.62	2.61	4.55	3.38	4.26	2.07	3.59	1.98
remp_64	8.43	2.23	3.94	0.77	**3.27**	**0.83**	3.81	0.97	4.68	0.89	3.72	0.67
warp_16_6	3.36	0.74	0.74	0.18	0.78	0.23	**0.71**	**0.21**	1.20	0.24	1.16	0.25
warp_32_12	3.00	1.11	1.24	0.60	0.96	0.47	1.01	0.46	1.56	0.42	1.52	0.66
warp_48_18	2.45	1.15	1.15	0.60	0.87	0.42	0.92	0.47	1.44	0.45	1.34	0.67
warp_64_24	3.38	1.51	1.30	0.65	1.02	0.58	1.00	0.55	1.55	0.64	1.28	0.63
	PLUSVein-FV3 Laser											
remp_16	14.29	0.80	9.00	0.34	9.63	0.27	15.50	0.96	11.87	0.61	9.43	0.52
remp_32	12.02	1.12	7.72	0.97	10.24	0.57	12.38	2.07	11.54	1.42	6.60	0.94
remp_48	11.55	3.47	6.86	1.71	10.45	2.03	14.10	3.42	12.51	3.41	**5.52**	**2.04**
remp_64	10.79	5.10	7.20	2.19	10.60	0.88	14.82	6.78	15.90	11.24	5.60	2.90
warp_16_6	6.33	0.99	2.21	0.20	8.78	0.10	3.30	0.41	4.27	0.39	**2.02**	**0.18**
warp_32_12	6.20	2.26	3.00	0.75	8.80	0.10	4.29	1.38	5.09	1.67	2.66	0.70
warp_48_18	4.38	1.38	2.50	0.61	8.75	0.15	3.53	1.13	4.22	1.10	2.20	0.58
warp_64_24	4.59	1.59	2.86	0.73	8.73	0.30	3.76	1.38	4.16	1.22	2.32	0.68
	PLUSVein-FV3 LED											
remp_16	14.03	1.03	10.01	0.47	14.57	0.62	16.50	1.10	12.64	0.73	10.67	0.57
remp_32	11.84	1.68	8.12	1.14	9.72	2.00	12.81	2.51	12.18	1.85	6.79	1.05
remp_48	10.32	3.08	6.68	1.57	7.71	2.47	13.43	4.00	12.21	2.92	**4.42**	**1.27**
remp_64	10.08	5.76	7.21	2.20	9.73	8.87	14.48	7.78	16.11	11.32	5.14	3.08
warp_16_6	5.27	0.99	1.33	0.17	2.01	0.52	2.30	0.53	3.88	0.58	**1.00**	**0.17**
warp_32_12	5.67	2.29	2.51	0.92	3.32	1.92	3.76	1.66	4.87	1.86	1.84	0.72
warp_48_18	3.95	1.45	1.81	0.72	2.23	1.11	3.05	1.39	3.84	1.36	1.36	0.57
warp_64_24	4.07	1.69	2.48	1.15	2.27	1.06	3.51	1.66	3.90	1.65	1.55	0.90

At first we will discuss the results given by Table 16.2. Not surprisingly, the worst performance is observed for block re-mapping (*remp_16*, *remp_32*, *remp_48* and *remp_64*) using 16 $\times$ 16 as smallest block size while GF was applied (UTFVP). This trend is in line with the findings of Piciucco et al. [20], which have been observed in the signal domain. It has to be mentioned that the observed results are strongly depending on the particular feature extraction method. As in [20] only the GF method was used for feature extraction, a direct comparison can only be done based on the GF results using the UTFVP dataset. This direct comparison shows that our best results on GF are worse compared to the results presented in [20] as we used a different implementation of the scheme. However, the best results using UTFVP are obtained by MC using a block size of 64 $\times$ 64 (*EER* 3.27). In general *remp_48* and *remp_64*

always resulted in the best performance for all datasets and not only on UTFVP (best *EER* of 5.52/4.42 for Laser/LED was achieved by applying WLD and *remp_48*). The only exception to this trend is given by RLT on the Laser/LED dataset. In this particular case, *remp_64* was performing worst, but this is a feature extraction type based observation.

In contrast to the block re-mapping based methods, the recognition performance of the warping based experiments (*warp_16_6* till *warp_64_24*) is better as observed for block re-mapping. This is in line with results reported for warping based experiments done in other biometric applications, e.g. [22] but opposed to the result of [20]. The best result on UTFVP is obtained for using PC and *warp_16_6* (*EER* 0.71). Nevertheless, there is not a big difference to the *EER* given by *warp_32_12*, *warp_48_18* and *warp_64_24*. It seems that the parameter choice has not a very high influence on the reported performance. For the other two datasets using WLD is resulting in the best *EER* values (Laser: 2.02, LED: 1.00).

As we want to compare the recognition performance of the feature domain template protected data to the same experiments which have been considering the transformations in the signal domain we will discuss the corresponding results now. The *EER* values applying template protection in the signal domain using system based keys are presented in Table 16.3.

The most important aspect using block re-mapping in the signal domain instead of applying the template protection schemes in the feature domain is a quite high-performance degradation in most of the conducted experiments. As mentioned in Sect. 16.2 it is likely that the feature extraction of the vein patterns after the template protection done in the signal domain might cause problems. This overall trend is confirmed by the observed *EER* results presented in Table 16.3. On UTFVP data, IUWT and PC resulted in the same trend that bigger block sizes are favourable in terms of performance (best average *EER*, 12.84, is given by IUWT using *remp_64*). For all other extraction schemes the *EER* values for *remp_16* or *remp_32* are better compared to *remp_64*. However, the performance difference is quite small.

Using warping, the influence on extracting the finger vein based features in the signal domain as compared to conducting the extraction in the feature domain is not so high as reported for block re-mapping. Hence, the overall performance trend using warping regardless of which dataset is considered, is similar to the results given in Table 16.2 (feature domain). IUWT again performs best in terms of *EER*. For *warp_16_6* the best performance can be reported. Surprisingly, the best average *EER*, 1.08, and the other performance values which are achieved applying IUWT on the template protected images are very similar for UTFVP and the LED dataset among each other.

Table 16.3 Recognition performance results (%) for template protection in the signal domain using system keys. The best performing results for each template protection method are highlighted in bold numbers

tempProt	EER											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	14.66	0.66	15.43	0.80	15.04	0.58	15.67	0.62	16.09	1.04	15.02	0.58
remp_32	15.14	1.35	13.75	1.54	14.67	1.65	14.17	1.32	14.25	1.54	14.58	1.31
remp_48	17.88	1.63	13.24	1.16	16.03	1.25	13.91	1.16	15.29	1.37	15.34	1.31
remp_64	17.68	1.52	**12.84**	**1.11**	16.33	1.46	13.01	1.06	15.27	1.79	14.95	1.51
warp_16_6	4.13	0.48	**1.08**	**0.16**	2.28	0.39	1.23	0.20	1.35	0.16	2.00	0.34
warp_32_12	3.49	0.64	2.31	0.55	3.35	0.62	2.70	0.64	1.81	0.29	3.04	0.53
warp_48_18	3.93	0.81	2.53	0.72	3.44	0.84	2.79	0.82	2.33	0.61	3.05	0.76
warp_64_24	3.40	0.83	2.15	0.67	2.66	0.95	2.11	0.82	2.04	0.40	2.47	0.72
	PLUSVein-FV3 Laser											
remp_16	9.87	0.54	9.43	0.49	9.74	0.41	10.61	0.38	10.47	0.62	9.24	0.45
remp_32	9.44	0.77	**8.30**	**0.60**	8.46	0.35	10.07	0.71	9.53	0.60	8.42	0.59
remp_48	10.36	1.05	9.14	0.89	9.04	0.85	10.15	0.69	10.49	0.75	9.17	0.99
remp_64	11.40	0.87	9.67	0.81	9.04	0.76	11.16	1.28	10.78	0.63	9.38	0.99
warp_16_6	6.94	0.82	**2.61**	**0.20**	7.01	0.19	5.84	0.95	4.80	0.62	2.72	0.20
warp_32_12	8.38	1.01	3.99	0.52	6.63	0.19	9.37	0.92	6.58	0.67	3.83	0.56
warp_48_18	6.12	1.49	3.49	0.66	6.84	0.26	7.71	1.49	5.70	1.17	3.38	0.79
warp_64_24	6.00	1.60	3.39	0.65	6.96	0.27	7.52	1.66	5.67	1.39	3.07	0.70
	PLUSVein-FV3 LED											
remp_16	14.99	0.89	15.29	0.71	15.27	0.57	16.24	1.05	16.79	1.37	15.02	0.58
remp_32	15.88	1.43	13.75	1.54	15.49	1.60	15.20	1.34	15.28	1.60	14.58	1.31
remp_48	18.97	2.18	13.21	1.23	17.01	1.86	14.51	1.77	16.49	1.73	15.34	1.31
remp_64	19.15	2.51	**12.84**	**1.11**	17.23	2.31	13.90	1.77	15.46	2.59	14.95	1.51
warp_16_6	4.85	0.51	**1.08**	**0.16**	3.13	0.46	2.04	0.26	2.59	0.22	2.00	0.34
warp_32_12	5.06	0.86	2.33	0.58	4.63	1.02	4.05	0.86	3.09	0.38	3.04	0.53
warp_48_18	5.05	0.94	2.53	0.72	4.71	0.88	4.49	0.69	3.56	0.65	3.05	0.76
warp_64_24	4.46	0.92	2.15	0.67	3.93	1.09	3.71	0.82	3.38	0.51	2.47	0.72

16.7.3 Set 3—Subject Dependent Versus System-Dependent Key

In this subsection, the capture subject-specific key experiments and their results are described and compared to the performance values obtained by using a system-dependent key. For the capture subject specific key experiments, a different and unique key for each finger is selected, compared to only one system-specific key, which is the same for all fingers. This should lead to a better differentiation of single capture subjects as the inter-subject variability is increased. Considering the subject dependent template protection experiments the results are summarised in Tables 16.4—feature domain, and 16.5—signal domain, respectively. As expected,

Table 16.4 Recognition performance results (%) for template protection in the feature domain using subject-specific keys. The best performing results for each template protection method are highlighted in bold numbers

tempProt	EER											
	GF		IUWT		MC		**PC**		**RLT**		**WLD**	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	3.28	0.20	4.82	0.38	4.70	0.29	7.68	0.47	3.19	0.24	5.79	0.31
remp_32	2.83	0.35	3.49	0.29	2.63	0.30	4.65	0.40	2.74	0.20	3.31	0.24
remp_48	2.93	0.28	3.18	0.41	2.27	0.25	3.41	0.36	2.90	0.33	2.31	0.15
remp_64	4.35	0.41	2.28	0.27	2.09	0.27	2.35	0.29	2.27	0.41	**1.90**	**0.27**
warp_16_6	2.74	0.23	0.72	0.18	**0.68**	**0.20**	0.82	0.13	1.00	0.10	1.24	0.16
warp_32_12	3.01	0.26	1.35	0.16	1.12	0.13	1.23	0.15	1.54	0.15	1.63	0.24
warp_48_18	2.20	0.24	1.16	0.19	0.90	0.16	1.08	0.16	1.26	0.20	1.33	0.15
warp_64_24	2.72	0.26	1.39	0.18	0.99	0.29	1.37	0.27	1.33	0.18	1.45	0.30
	PLUSVein-FV3 Laser											
remp_16	12.83	0.53	5.58	0.30	3.44	0.17	14.95	0.52	6.58	0.27	6.11	0.37
remp_32	9.50	0.40	3.46	0.28	3.60	0.23	10.96	0.45	6.58	0.38	3.81	0.21
remp_48	8.30	0.43	3.27	0.22	3.94	0.14	12.53	0.53	7.72	0.52	3.31	0.27
remp_64	5.15	0.31	**1.84**	**0.17**	3.93	0.26	11.09	0.93	6.93	0.62	2.52	0.15
warp_16_6	5.14	0.19	**1.81**	**0.10**	6.04	0.15	2.78	0.15	3.82	0.16	1.95	0.12
warp_32_12	6.07	0.28	2.41	0.16	5.83	0.12	4.54	0.33	5.02	0.27	2.41	0.15
warp_48_18	4.31	0.33	2.12	0.18	6.05	0.10	3.63	0.32	3.93	0.30	2.02	0.17
warp_64_24	4.42	0.35	2.26	0.24	4.71	0.23	4.35	0.22	3.96	0.40	2.18	0.19
	PLUSVein-FV3 LED											
remp_16	12.23	0.54	5.90	0.32	13.13	0.43	15.30	0.46	6.85	0.27	6.70	0.37
remp_32	9.08	0.42	3.67	0.36	8.02	0.38	11.00	0.65	6.79	0.28	3.81	0.25
remp_48	7.61	0.31	3.58	0.16	5.65	0.34	12.69	0.64	8.01	0.39	2.95	0.29
remp_64	4.64	0.33	**1.93**	**0.15**	5.51	0.40	10.35	0.80	7.32	0.73	2.15	0.26
warp_16_6	4.10	0.22	0.90	0.09	1.42	0.14	1.78	0.23	3.18	0.29	**0.85**	**0.09**
warp_32_12	5.85	0.41	2.00	0.21	3.36	0.28	4.22	0.27	4.83	0.19	1.65	0.23
warp_48_18	3.83	0.43	1.54	0.17	2.39	0.32	3.44	0.26	3.69	0.44	1.32	0.24
warp_64_24	3.88	0.30	1.85	0.29	2.54	0.43	4.00	0.31	3.66	0.33	1.50	0.44

it becomes apparent that the overall performance of all experiments using subject dependent keys is much better compared to the system-specific key results. This can be explained as the usage of subject dependent keys provides a better separation of genuine and impostor score distributions after applying the transformation.

The best feature domain based performance (see Table 16.4) is obtained on UTFVP using WLD during *remp_64* (*EER* 1.90) and MC during *warp_16_6* (*EER* 0.68), on the Laser dataset using IUWT (*EER* 1.84 for *remp_64*, *EER* 1.81 for *warp_16_6*) and finally on the LED dataset using IUWT/WLD (*EER* 1.93/0.85) applying *remp_64*/*warp_16_6*. According to the *EER* values highlighted in Table 16.5 (signal domain) the overall best recognition performance is achieved by applying the template protection schemes in the signal domain using subject-specific

Table 16.5 Recognition performance results (%) for template protection in the signal domain using subject-specific keys. The best performing results for each template protection method are highlighted in bold numbers

tempProt	EER											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	0.98	0.15	0.59	0.09	0.78	0.10	0.98	0.13	1.13	0.13	0.53	0.04
remp_32	0.50	0.09	0.48	0.07	0.69	0.12	0.86	0.18	0.96	0.18	0.41	0.09
remp_48	**0.39**	**0.04**	0.46	0.07	0.54	0.05	0.77	0.14	0.52	0.10	0.45	0.08
remp_64	0.48	0.12	0.44	0.15	0.57	0.13	0.60	0.13	0.57	0.10	0.41	0.08
warp_16_6	1.39	0.16	0.57	0.07	0.64	0.12	**0.42**	**0.09**	0.80	0.06	0.43	0.10
warp_32_12	2.16	0.22	1.31	0.28	1.65	0.19	1.66	0.17	1.22	0.20	1.47	0.16
warp_48_18	2.15	0.19	1.55	0.14	1.64	0.20	1.70	0.15	1.51	0.14	1.76	0.16
warp_64_24	1.80	0.23	1.64	0.25	1.50	0.20	1.68	0.31	1.44	0.18	1.77	0.17
	PLUSVein-FV3 Laser											
remp_16	3.44	0.17	**0.96**	**0.11**	0.69	0.06	3.04	0.16	3.6	0.25	1.08	0.10
remp_32	3.21	0.33	1.29	0.20	0.96	0.08	4.29	0.26	4.34	0.29	1.38	0.20
remp_48	3.95	0.33	1.47	0.20	1.09	0.12	4.71	0.27	5.48	0.20	1.68	0.16
remp_64	4.09	0.31	1.88	0.19	1.11	0.10	5.45	0.40	4.89	0.38	2.17	0.24
warp_16_6	5.62	0.24	2.35	0.06	2.82	0.14	5.65	0.15	4.57	0.21	**2.12**	**0.10**
warp_32_12	6.74	0.27	2.87	0.10	2.88	0.12	8.46	0.26	5.74	0.26	2.65	0.11
warp_48_18	6.23	0.26	3.02	0.19	3.23	0.14	8.17	0.31	5.76	0.28	2.77	0.23
warp_64_24	5.71	0.30	2.81	0.14	3.87	0.20	7.61	0.31	5.54	0.34	2.45	0.12
	PLUSVein-FV3 LED											
remp_16	1.93	0.20	0.59	0.09	1.54	0.10	1.93	0.21	2.27	0.19	0.53	0.04
remp_32	1.06	0.16	0.48	0.07	1.51	0.20	2.53	0.27	2.14	0.21	0.41	0.09
remp_48	1.08	0.09	0.46	0.07	1.76	0.13	2.50	0.25	1.25	0.18	0.45	0.08
remp_64	0.88	0.16	0.44	0.15	1.46	0.23	1.87	0.25	1.21	0.21	**0.41**	**0.08**
warp_16_6	3.64	0.23	0.57	0.07	2.7	0.27	1.78	0.22	2.25	0.25	**0.43**	**0.10**
warp_32_12	4.81	0.28	1.31	0.28	4.05	0.22	3.95	0.24	2.83	0.23	1.47	0.16
warp_48_18	4.07	0.22	1.55	0.14	3.61	0.33	3.89	0.25	3.03	0.32	1.76	0.16
warp_64_24	3.72	0.30	1.64	0.25	3.37	0.25	3.77	0.29	3.06	0.13	1.77	0.17

keys. This observation is interesting because it seems that in most cases subject-specific keys have a more positive effect on the protected features' performance if the corresponding transformation was applied in the signal domain. However, there are also some cases where the subject-specific keys' signal domain performance is lower compared to the best results obtained in the feature domain, e.g. Laser dataset using WLD and *warp_16_6*. Compared to [20] the recognition performance presented in Table 16.5 using GF is outperforming the findings stated by Piciucco et al. no matter if block re-mapping or warping is considered. All other results obtained for UTFVP are better as well.

16.7.4 Set 4—Unlinkability Analysis

The unlinkability analysis is performed to ensure that the applied template protection schemes meet the principles established by the ISO/IEC 24745 standard [7], in particular the unlinkability requirements. If there is a high amount of linkability for a certain template protection scheme, it is easy to match two protected templates from the same finger among different applications using different keys. In that case, it is easy to track the capture subjects across different applications, which poses a threat to the capture subjects' privacy. The unlinkability is likely to be low (linkability high) if there is too little variation between protected templates based on two different keys (i.e. the key-sensitivity is low) or the unprotected and the protected template in general. Tables 16.6, 16.7, 16.8 and 16.9 lists the global unlinkability scores, D_{sys},

Table 16.6 D_{sys} unlinkability scores for the selected template protection schemes applied in feature domain using system dependent keys. The best results (low values, representing unlinkability) for each template protection method are highlighted in bold numbers

tempProt	Dsys											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	3.43	0.52	3.02	0.58	3.91	0.97	**2.90**	**0.32**	3.09	0.46	4.35	0.53
remp_32	13.96	13.89	10.32	16.79	15.83	16.09	9.34	17.09	8.21	5.88	16.71	19.25
remp_48	18.72	22.27	14.92	23.90	21.49	25.76	13.86	23.66	10.65	17.60	18.54	26.33
remp_64	25.67	18.69	20.03	21.95	29.72	22.93	20.81	22.36	16.32	19.89	27.24	22.93
warp_16_6	56.35	8.94	85.01	6.92	82.61	6.31	79.54	8.02	81.66	6.00	74.87	8.19
warp_32_12	37.54	17.97	40.18	21.74	55.86	17.66	40.18	21.45	44.37	18.8	43.19	19.43
warp_48_18	**36.15**	**22.52**	39.08	27.65	52.76	22.89	37.17	26.75	42.92	25.3	41.56	25.20
warp_64_24	42.43	29.21	41.21	32.14	53.13	28.26	48.81	32.36	44.68	31.45	43.43	21.12
	PLUSVein-FV3 Laser											
remp_16	4.07	0.50	2.73	0.44	3.42	0.70	2.79	0.53	**2.64**	**0.49**	4.28	0.81
remp_32	20.16	17.10	13.77	18.96	21.00	20.97	10.40	18.23	8.80	9.95	17.13	21.00
remp_48	14.00	17.39	9.18	17.69	14.53	20.09	7.18	16.55	5.26	8.39	12.75	20.81
remp_64	19.58	22.01	14.37	22.48	24.51	22.42	10.06	18.07	7.38	10.77	17.77	21.53
warp_16_6	63.42	10.55	81.26	10.17	86.37	4.36	83.99	6.77	68.19	9.82	82.1	8.65
warp_32_12	34.62	17.86	35.90	20.96	53.34	17.51	44.14	18.82	**29.83**	**13.38**	46.36	18.48
warp_48_18	44.30	21.56	42.61	23.67	58.42	18.86	47.80	21.23	34.44	18.93	52.10	20.69
warp_64_24	33.33	26.48	35.28	28.94	43.99	28.59	34.27	27.20	28.83	24.97	47.23	17.95
	PLUSVein-FV3 LED											
remp_16	3.81	0.42	2.86	0.46	3.34	0.62	2.55	0.35	**2.34**	**0.45**	4.04	0.65
remp_32	19.67	17.36	13.07	18.9	21.71	20.47	10.62	18.32	8.69	10.20	17.03	21.01
remp_48	14.06	17.78	9.18	17.51	14.99	20.62	7.42	16.15	5.56	9.35	13.23	21.38
remp_64	19.44	22.26	13.58	22.05	23.53	22.61	10.13	17.94	7.71	10.23	16.91	21.19
warp_16_6	67.02	10.53	81.95	10.31	86.66	6.62	84.38	7.23	67.51	10.59	82.58	8.37
warp_32_12	37.51	17.38	35.27	20.74	56.52	16.52	44.58	18.94	**28.62**	**12.99**	47.66	18.31
warp_48_18	45.41	22.65	42.56	24.21	60.46	19.33	48.14	21.48	34.10	19.43	52.99	21.83
warp_64_24	32.81	26.35	32.99	28.49	45.09	28.34	33.88	27.47	27.70	24.25	48.52	19.11

Table 16.7 D_{sys} unlinkability scores for the selected template protection schemes applied in signal domain using system dependent keys. The best results (low values, representing unlinkability) for each template protection method are highlighted in bold numbers

tempProt	Dsys											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	**2.97**	**0.63**	2.97	0.66	3.05	0.59	2.77	0.33	2.86	0.51	3.34	0.81
remp_32	6.03	12.42	6.17	12.76	6.07	12.19	5.80	12.48	5.84	12.55	6.46	12.39
remp_48	6.74	12.04	6.70	13.94	6.91	12.34	6.93	13.81	6.00	13.35	7.58	13.07
remp_64	6.43	11.24	6.56	13.61	6.37	11.53	6.63	13.79	6.16	12.94	7.30	13.04
warp_16_6	73.00	4.02	87.21	2.49	82.97	3.05	84.81	2.82	83.14	2.60	84.17	2.89
warp_32_12	42.85	14.28	50.90	14.94	52.36	12.40	47.03	15.45	55.01	12.49	49.88	14.39
warp_48_18	32.64	17.20	33.24	19.71	42.26	16.50	33.26	20.97	36.79	18.98	37.10	18.63
warp_64_24	26.00	17.83	**17.92**	**11.65**	28.71	14.28	23.11	20.41	26.72	19.19	26.40	18.45
	PLUSVein-FV3 Laser											
remp_16	2.68	0.44	2.30	0.34	2.41	0.40	**2.25**	**0.32**	2.57	0.38	3.59	1.03
remp_32	8.83	16.19	6.96	16.48	7.20	16.63	6.32	16.16	6.74	16.63	9.66	16.12
remp_48	9.12	15.05	7.27	15.86	8.52	14.69	6.71	15.86	6.30	16.13	8.90	15.21
remp_64	9.78	15.43	7.76	15.97	8.88	15.36	7.83	16.08	7.43	15.97	10.50	15.92
warp_16_6	76.36	5.87	89.06	2.95	87.14	1.66	77.23	2.55	80.01	2.64	86.3	1.51
warp_32_12	42.37	20.44	44.63	12.57	51.05	12.38	35.13	14.56	42.47	14.04	53.03	7.96
warp_48_18	29.59	16.50	31.37	18.84	41.96	18.63	24.50	16.14	27.83	19.52	47.16	23.77
warp_64_24	27.99	20.84	25.37	21.61	37.85	20.00	**20.92**	**17.16**	23.03	18.72	21.00	4.95
	PLUSVein-FV3 LED											
remp_16	2.97	0.63	3.00	0.5	3.00	0.56	**2.80**	**0.60**	2.86	0.51	3.34	0.81
remp_32	6.20	12.78	6.37	13.12	6.07	12.19	6.02	12.44	2.88	0.44	6.46	12.39
remp_48	6.74	12.04	7.05	14.79	6.91	12.34	6.99	13.62	6.38	14.16	7.58	13.07
remp_64	6.43	11.24	6.56	13.61	6.37	11.53	6.74	13.87	6.16	12.94	7.30	13.04
warp_16_6	78.42	5.77	90.04	3.05	88.48	1.96	78.32	3.05	81.22	3.46	88.02	1.94
warp_32_12	41.32	21.14	45.32	12.85	52.04	11.89	36.03	15.06	44.86	15.21	53.97	8.69
warp_48_18	30.55	17.05	31.34	19.79	41.58	19.25	23.99	15.12	28.66	20.12	48.86	24.97
warp_64_24	27.89	21.00	25.27	20.99	37.92	19.53	21.45	18.10	24.26	19.27	**20.95**	**6.01**

for all datasets using block re-mapping and warping, similar to the tables that have been used to describe the recognition performance. The D_{sys} ranges normally from 0 to 1, where 0 represents the best achievable unlinkability score. We shifted the range from [0, 1] to values in [0, 100] to improve the readability of the results.

The D_{sys} ranges reveal that there are several block re-mapping configurations leading to a low linkability score, indicating that the protected templates cannot be linked across different applications (high unlinkability). This can be observed not only for applying block re-mapping in the feature domain using system-specific keys but also for the application in all other feature spaces and key selection strategies. The lowest D_{sys} scores can be detected for the usage of *remp_16*. For most block sizes

Table 16.8 D_{sys} unlinkability scores for the selected template protection schemes applied in feature domain using subject dependent keys. The best results (low values, representing unlinkability) for each template protection method are highlighted in bold numbers

tempProt	Dsys											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	3.43	0.52	3.02	0.58	3.95	0.96	**2.90**	**0.32**	3.09	0.46	4.35	0.53
remp_32	13.93	13.48	10.32	16.79	15.83	16.09	9.34	17.09	8.21	5.88	15.75	17.80
remp_48	18.72	22.27	14.92	23.90	21.49	25.76	13.86	23.66	11.82	17.72	18.78	25.23
remp_64	24.41	18.90	20.03	21.95	30.95	22.85	20.81	22.36	15.54	19.52	25.92	22.94
warp_16_6	58.04	1.07	83.14	0.75	82.94	0.85	79.15	0.62	79.77	0.73	74.92	1.05
warp_32_12	**31.73**	**11.71**	35.67	11.85	49.61	8.56	33.63	12.28	38.96	1.45	37.93	11.00
warp_48_18	39.17	10.46	41.63	10.63	53.54	7.86	40.12	10.94	44.94	1.53	44.28	9.66
warp_64_24	35.15	11.68	32.15	12.61	44.85	9.88	37.82	11.54	36.36	1.57	36.56	10.85
	PLUSVein-FV3 Laser											
remp_16	4.07	0.50	2.73	0.44	3.41	0.75	2.79	0.53	**2.65**	**0.51**	4.29	0.84
remp_32	20.38	17.60	13.77	18.96	21.00	20.97	10.40	18.23	9.08	10.21	17.95	21.35
remp_48	14.61	17.73	9.18	17.69	14.53	20.09	7.18	16.55	5.26	8.39	12.75	20.81
remp_64	19.58	22.01	14.37	22.48	24.51	22.42	10.48	18.53	7.38	10.77	17.77	21.53
warp_16_6	67.33	0.71	84.02	0.88	87.14	0.92	85.30	0.67	70.99	0.97	83.39	0.93
warp_32_12	32.71	12.16	34.82	13.06	49.23	9.01	41.37	10.71	29.76	0.94	44.48	10.19
warp_48_18	38.73	11.35	39.00	11.80	50.29	9.14	40.83	11.08	34.65	1.93	46.5	10.38
warp_64_24	31.46	13.65	32.65	13.16	41.79	11.84	31.79	13.04	**28.76**	**1.34**	43.25	10.12
	PLUSVein-FV3 LED											
remp_16	3.81	0.42	2.86	0.46	3.34	0.62	2.55	0.35	**2.34**	**0.45**	4.04	0.65
remp_32	19.67	17.36	13.07	18.90	21.71	20.47	10.62	18.32	8.69	10.20	17.03	21.01
remp_48	14.06	17.78	9.18	17.51	14.99	20.62	7.42	16.15	5.56	9.35	13.23	21.38
remp_64	19.44	22.26	13.58	22.05	23.53	22.61	10.13	17.94	7.71	10.23	16.91	21.19
warp_16_6	71.08	0.81	83.99	0.87	88.52	0.64	86.55	0.66	70.93	1.10	84.96	0.80
warp_32_12	34.39	11.73	33.42	12.73	51.79	8.49	41.2	10.74	**28.42**	**0.75**	45.48	9.75
warp_48_18	39.69	11.23	38.46	11.82	51.87	8.79	40.85	11.3	34.03	1.65	46.56	9.92
warp_64_24	31.96	12.91	31.72	13.33	43.29	10.78	31.66	13.12	28.79	1.73	44.34	9.68

48 $\times$ 48 or 64 $\times$ 64 the unlinkability values are higher compared to the schemes using lower block sizes. Thus, the linkability is increased.

For warping the situation is different. First, the obtained D_{sys} is mostly quite high which indicates a high linkability regardless the choice of key selection strategy or the domain. Second, *warp_32_12* or *warp_48_18* exhibit the lowest unlinkability scores, clearly the highest amount of linkability detected for *warp_16_6*. The reason for this is given by the applied warping scheme. If small block sizes are used the offset, which is responsible for the amount of introduced degradation during the transformation, is small as well. Thus, for an offset of 6 only a little amount of variation in the original image (signal domain) or extracted template (feature domain) is caused. Of course,

Table 16.9 D_{sys} unlinkability scores for the selected template protection schemes applied in signal domain using subject dependent keys. The best results (low values, representing unlinkability) for each template protection method are highlighted in bold numbers

tempProt	Dsys											
	GF		IUWT		MC		PC		RLT		WLD	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
	UTFVP											
remp_16	2.97	0.63	2.97	0.66	3.05	0.59	**2.77**	**0.33**	2.86	0.51	3.34	0.81
remp_32	6.16	12.79	6.17	12.76	6.07	12.19	5.80	12.48	5.84	12.55	6.67	12.74
remp_48	6.74	12.04	6.70	13.94	6.91	12.34	6.93	13.81	6.00	13.35	7.58	13.07
remp_64	6.43	11.24	6.56	13.61	6.37	11.53	6.63	13.79	6.16	12.94	7.30	13.04
warp_16_6	75.41	0.56	88.16	0.35	84.89	0.55	87.22	0.52	85.51	0.53	85.8	0.43
warp_32_12	46.76	8.57	54.42	7.20	55.75	6.54	51.39	7.64	58.88	6.22	54.15	6.97
warp_48_18	33.87	12.06	33.68	12.37	41.34	10.28	32.58	12.28	38.74	11.10	36.49	11.48
warp_64_24	29.82	13.06	**25.12**	**14.42**	34.74	12.11	25.41	14.21	30.16	13.12	29.96	13.16
	PLUSVein-FV3 Laser											
remp_16	2.68	0.44	2.30	0.34	2.41	0.40	**2.25**	**0.32**	2.57	0.38	3.59	1.03
remp_32	8.83	16.19	6.96	16.48	7.20	16.63	6.32	16.16	6.74	16.63	9.66	16.12
remp_48	9.12	15.05	7.27	15.86	8.52	14.69	6.71	15.86	6.30	16.13	8.90	15.21
remp_64	9.78	15.43	7.76	15.97	8.88	15.36	7.83	16.08	7.43	15.97	10.50	15.92
warp_16_6	78.38	0.56	90.26	0.46	87.71	0.51	78.41	0.58	80.34	0.48	88.56	0.43
warp_32_12	42.94	0.86	49.98	1.11	54.95	1.50	36.00	0.99	42.75	1.14	56.58	1.22
warp_48_18	29.25	0.95	29.10	0.97	37.86	1.20	21.38	0.70	25.02	1.17	37.98	1.07
warp_64_24	23.59	1.18	19.47	1.44	31.20	1.64	**15.97**	**0.98**	18.19	1.08	29.28	1.29
	PLUSVein-FV3 LED											
remp_16	2.97	0.63	3.00	0.5	3.00	0.56	**2.80**	**0.60**	2.86	0.51	3.34	0.81
remp_32	6.03	12.42	6.17	12.76	6.07	12.19	6.02	12.44	5.84	12.55	6.46	12.39
remp_48	6.74	12.04	7.05	14.79	6.91	12.34	6.99	13.62	6.38	14.16	7.58	13.07
remp_64	6.43	11.24	6.56	13.61	6.37	11.53	6.74	13.87	6.16	12.94	7.30	13.04
warp_16_6	79.05	0.78	91.02	0.59	88.45	0.89	79.88	0.75	81.45	0.75	89.16	0.66
warp_32_12	43.65	1.22	50.25	1.89	55.66	1.87	36.99	1.36	43.83	2.04	57.39	1.77
warp_48_18	29.85	1.36	30.02	1.48	38.58	1.67	22.22	1.12	26.28	1.99	38.75	1.39
warp_64_24	24.58	1.77	20.23	2.06	32.52	2.24	**16.28**	**1.25**	19.12	1.89	30.42	1.58

this results in a high linkability score as the transformed biometric information is minimally protected.

In Fig. 16.6 4 examples exhibiting score distributions and corresponding D_{sys} values are shown for block re-mapping: First row—*remp_16* (a) and *remp_54* (b), and warping: Second row—*warp_16_6* (c) and *warp_64_24* (d). The blue line represents the process of D_{sys} for all threshold selections done during the computation (see [4]). The green distribution corresponds to the so called *mated* samples scores. These comparison scores are computed from templates extracted from samples of a single instance of the same subject using different keys [4]. The red coloured distribution describes the *non-mated* samples scores, which are yielded by templates generated

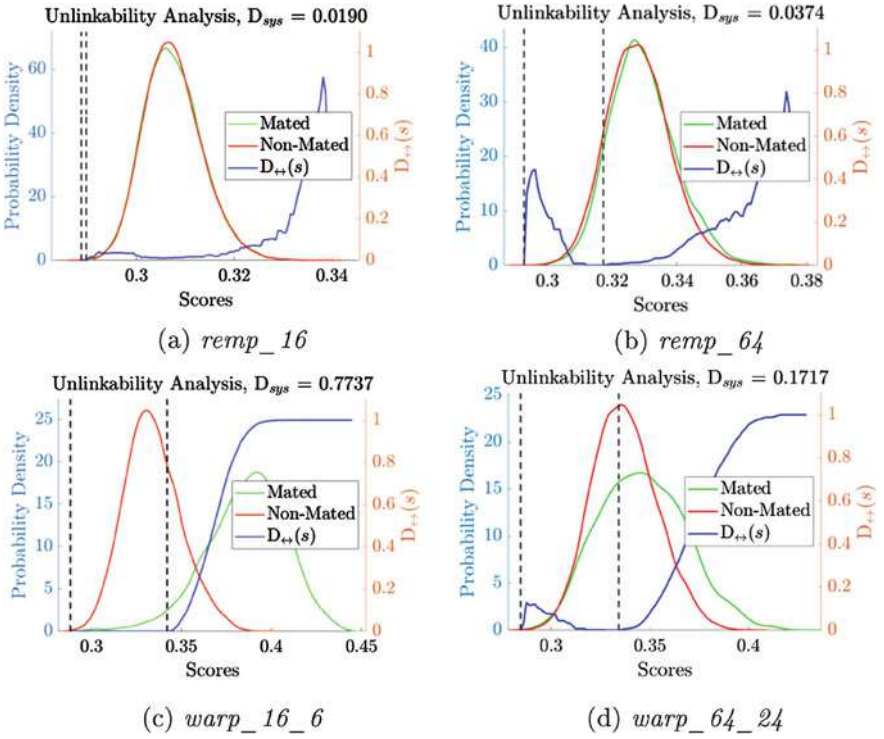


Fig. 16.6 Example images which display unlinkability scores. In all four examples signal domain, PC features on the PLUSVein-FV3 Laser dataset and subject-specific key selection was applied

from samples of different instances using different keys. According to [4] a *fully unlinkable* scenario can be observed if both coloured distributions are identical, while *full linkability* is given if mated and non-mated distributions can be fully separated from each other. For block re-mapping, (a) and (b) almost full unlinkability is achieved in both cases, while for the warping examples, (c) and (d) the distributions can be partly separated from each other. The worst result regarding the ISO/IEC Standard 24745 [7] property of unlinkability is exhibited by example (c) as both distributions are separated quite well, which leads to a high amount of linkability. Thus, in *warp_16_6* it is possible to decide with high probability to which dataset a protected template belongs. Furthermore, from a security point of view warping is not really a suitable template protection scheme using the given parameters. As the amount of linkability decreases using bigger block sizes and more importantly larger offsets it seems to be possible to select a parameter set-up that is providing both a good recognition performance and a quit low linkability at the same time.

According to these results, it is possible to summarise the findings taking the recognition performance and unlinkability evaluation into account:

(a) Only a very low amount of capture subject's privacy protection for configurations exhibiting a low *EER* is obtained for the application of warping schemes.
(b) A high *EER* is observed for the configurations that exhibit a high unlinkability (e.g. detected during the application of block re-mapping schemes in most cases).

Additionally, it must be mentioned that the template protection application in feature or signal domain shows differences regarding the unlinkability aspect. For both, block re-mapping and warping, it is better to apply template protection in the signal domain as the D_{sys} values are lower for almost all cases. If the recognition performance is taken into account as well the best obtained experimental setting is the template protection application in the signal domain using subject-specific keys.

However, the provided level of privacy protection, especially if it comes to unlinkability is clearly not enough for a practical application of warping based cancellable schemes in the feature domain and several signal domain settings using the selected parameters. Furthermore, the worse recognition performance restricts the use of block re-mapping schemes for real-world biometric systems in the most cases as well.

16.8 Conclusion

In this chapter, template protection schemes in finger vein recognition with a focus on cancellable schemes and their application in the feature domain were presented and evaluated. The focus was hereby on cancellable schemes that can be applied in both the signal and the feature domain in the context of finger vein recognition. Two well-known representatives of those schemes, namely, block re-mapping and block warping were evaluated in signal and feature domain on two different publicly available finger vein data sets: the UTFVP and the palmar subsets of the PLUSVein-FV3. These schemes are the same ones that have been applied in the image domain in the previous work of Piciucco et al. [20].

Compared to the previous results obtained in [20], none of the block re-mapping methods performed well in the feature and signal domain using system-specific keys. The experiments considering a capture subject-specific key instead of a system specific one lead to an improvement regarding the recognition performance, especially in the signal domain. Warping performed much better in both domains but further results on the unlinkability revealed that the privacy protection amount is very limited. Thus, an application in real-world biometric systems is restricted for the most experimental settings according to the fact that it is possible to track a subject across several instances generated with various keys.

Nevertheless, it was possible to observe the following trend that leads to an optimistic conclusion. Of course, both template protection schemes have their weaknesses, block re-mapping exhibits recognition performance problems, while warping lacks in terms of unlinkability, but according to the results it seems that the selection of a larger offset could reduce the unlinkability issue for warping in the signal domain. In particular, the larger the offset was selected the better the unlinkability

performed, while the recognition performance was hardly influenced. According to this observation, we claim that warping is a suitable cancellable template protection scheme for finger vein biometrics if it is applied in the signal domain using subject-specific keys and a large offset to achieve sufficient unlinkability.

References

1. Choi JH, Song W, Kim T, Lee S-R, Kim HC (2009) Finger vein extraction using gradient normalization and principal curvature. Proc SPIE 7251:9
2. Gallagher AC (2012) Jigsaw puzzles with pieces of unknown orientation. In: 2012 IEEE Conference on computer vision and pattern recognition (CVPR). IEEE, pp 382–389
3. Glasbey CA, Mardia KV (1998) A review of image-warping methods. J Appl Stat 25(2):155–171
4. Gomez-Barrero M, Galbally J, Rathgeb C, Busch C (2018) General framework to evaluate unlinkability in biometric template protection systems. IEEE Trans Inf Forensics Secur 13(6):1406–1420
5. Hämmerle-Uhl J, Pschernig E, Uhl A (2009) Cancelable iris biometrics using block re-mapping and image warping. In: Samarati P, Yung M, Martinelli F, Ardagna CA (eds)Proceedings of the 12th international information security conference (ISC'09), volume 5735 of LNCS. Springer, pp 135–142
6. Huang B, Dai Y, Li R, Tang D, Li W (2010) Finger-vein authentication based on wide line detector and pattern normalization. In: 2010 20th International conference on pattern recognition (ICPR). IEEE, pp 1269–1272
7. ISO/IEC 24745—Information technology—Security techniques—biometric information protection. Standard, International Organization for Standardization, June 2011
8. Jadhav M, Nerkar PM (2015) Survey on finger vein biometric authentication system. Int J Comput Appl (3):14–17
9. Jain AK, Nandakumar K, Nagar A (2008) Biometric template security. EURASIP J Adv Signal Process 2008:113
10. Kauba C, Prommegger B, Uhl A (2018) The two sides of the finger—an evaluation on the recognition performance of dorsal vs. palmar finger-veins. In: Proceedings of the international conference of the biometrics special interest group (BIOSIG'18), Darmstadt, Germany (accepted)
11. Kauba C, Prommegger B, Uhl A (2019) An available open source vein recognition framework. In: Uhl A, Busch C, Marcel S, Veldhuis R (eds) Handbook of vascular biometrics. Springer Science+Business Media, Boston, MA, USA, pp 77–112
12. Kauba C, Prommegger B, Uhl A (2019) Openvein—an open-source modular multi-purpose finger-vein scanner design. In: Uhl A, Busch C, Marcel S, Veldhuis R (eds) Handbook of vascular biometrics. Springer Science+Business Media, Boston, MA, USA, pp 77–112
13. Kumar A, Zhou Y (2012) Human identification using finger images. IEEE Trans Image Process 21(4):2228–2244
14. Lai Y-L, Jin Z, Teoh ABJ, Goi B-M, Yap W-S, Chai T-Y, Rathgeb C (2017) Cancellable iris template generation based on indexing-first-one hashing. Pattern Recognit 64:105–117
15. Maio D, Maltoni D, Cappelli R, Wayman JL, Jain AK (2002) FVC2002: second fingerprint verification competition. In: 16th international conference on pattern recognition, 2002. Proceedings, vol 3. IEEE, pp 811–814
16. Maltoni D, Maio D, Jain AK, Prabhakar S (2009) Handbook of fingerprint recognition, 2nd edn. Springer

17. Miura N, Nagasaka A, Miyatake T (2004) Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach Vis Appl 15(4):194–203
18. Miura N, Nagasaka A, Miyatake T (2007) Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans Inf Syst 90(8):1185–1194
19. Paikin G, Tal A (2015) Solving multiple square jigsaw puzzles with missing pieces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4832–4839
20. Piciucco E, Maiorana E, Kauba C, Uhl A, Campisi P (2016) Cancelable biometrics for finger vein recognition. In: 2016 First international workshop on sensing, processing and learning for intelligent machines (SPLINE). IEEE, pp 1–5
21. Ratha NK, Chikkerur S, Connell JH, Bolle RM (2007) Generating cancelable fingerprint templates. IEEE Trans Pattern Anal Mach Intell 29(4):561–572
22. Ratha NK, Connell J, Bolle R (2001) Enhancing security and privacy in biometrics-based authentication systems. IBM Syst J 40(3):614–634
23. Sholomon D, David O, Netanyahu NS (2013) A genetic algorithm-based solver for very large jigsaw puzzles. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1767–1774
24. Starck J-L, Fadili J, Murtagh F (2007) The undecimated wavelet decomposition and its reconstruction. IEEE Trans Image Process 16(2):297–309
25. Ton BT, Veldhuis RNJ (2013) A high quality finger vascular pattern dataset collected using a custom designed capturing device. In: 2013 International conference on biometrics (ICB). IEEE, pp 1–5
26. Uhl A (2019) State-of-the-art in vascular biometrics. In: Uhl A, Busch C, Marcel S, Veldhuis R (eds) Handbook of vascular biometrics. Springer Science+Business Media, Boston, MA, USA, pp 3–62
27. Wolberg G (1998) Image morphing: a survey. Vis Comput 14(8):360–372

Chapter 17
Towards Measuring the Amount of Discriminatory Information in Finger Vein Biometric Characteristics Using a Relative Entropy Estimator

Vedrana Krivokuća, Marta Gomez-Barrero, Sébastien Marcel, Christian Rathgeb and Christoph Busch

Abstract This chapter makes the first attempt to quantify the amount of discriminatory information in finger vein biometric characteristics in terms of Relative Entropy (RE) calculated on genuine and impostor comparison scores using a Nearest Neighbour (NN) estimator. Our findings indicate that the RE is system-specific, meaning that it would be misleading to claim a universal finger vein RE estimate. We show, however, that the RE can be used to rank finger vein recognition systems (tested on the same database using the same experimental protocol) in terms of their expected recognition accuracy, and that this ranking is equivalent to that achieved using the EER. This implies that the RE estimator is a reliable indicator of the amount of discriminatory information in a finger vein recognition system. We also propose a Normalised Relative Entropy (NRE) metric to help us better understand the significance of the RE values, as well as to enable a fair benchmark of different biometric systems (tested on different databases and potentially using different experimental protocols) in terms of their RE. We discuss how the proposed NRE metric can be used as a complement to the EER in benchmarking the discriminative capabilities of different biometric systems, and we consider two potential issues that must be taken into account when calculating the RE and NRE in practice.

V. Krivokuća (✉) · S. Marcel
Idiap Research Institute, Martigny, Switzerland
e-mail: vedrana.krivokuca@idiap.ch

S. Marcel
e-mail: sebastien.marcel@idiap.ch

M. Gomez-Barrero · C. Rathgeb · C. Busch
Hochschule Darmstadt, Darmstadt, Germany
e-mail: marta.gomez-barrero@h-da.de

C. Rathgeb
e-mail: christian.rathgeb@h-da.de

C. Busch
e-mail: christoph.busch@h-da.de

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4_17

Keywords Finger veins · Relative entropy · Nearest neighbour estimator · Biometric template protection · Security · Privacy · Discriminatory information · Kullback–Leibler divergence · VERA · UTFVP · Wide Line Detector · Repeated Line Tracking · Maximum Curvature

17.1 Introduction

There is no doubt that biometrics are fast becoming ubiquitous in response to a growing need for more robust identity assurance. A negative consequence of this increasing reliance on biometrics is the looming threat of serious privacy and security concerns in the event that the growing biometric databases are breached.[1] Fortunately, the past decade has seen notable efforts in advancing the field of biometric template protection, which is dedicated to protecting the biometric data that is collected and used for recognition purposes, thereby safeguarding the privacy of the data subjects and preventing "spoofing" attacks using stolen biometric templates. Unfortunately, we are still lacking solid methods for evaluating the effectiveness of the proposed solutions. An important missing ingredient is a measure of the amount of discriminatory information in a biometric system.

A few approaches, for example, [1–3], have focused on estimating the "individuality" (or discrimination capability) of biometric templates in terms of the inter-class variation alone (i.e. the False Match Rate or False Accept Rate). Along the same lines, the best-known attempt to measure the amount of information in a biometric system is probably the approach proposed by Daugman [4]. This method computes the Hamming distance between every pair of non-mated IrisCodes, and the resulting distance distribution is then fitted to a binomial distribution. The number of degrees of freedom of the representative binomial distribution approximates the number of independent bits in each binary IrisCode, which in turn provides an estimate for the discrimination entropy of the underlying biometric characteristic. This approach was adopted to measure the entropy of finger vein patterns in [5]. However, as explained in [5], while this method of measuring entropy is correct from the source coding point of view, the issue with calculating the entropy in this way is that it only provides a reasonable estimate of the amount of biometric information if there is no variation between multiple samples captured from the same biometric instance. Since this intra-class variation is unlikely to be zero in practice, the discrimination entropy would probably overestimate the amount of available biometric information [6, 7].

In an attempt to extend the idea of using entropy as a measure of biometric information while more practically incorporating both inter- and intra-class variation, several authors have adopted the *relative entropy* approach. Adler et al. [8] defined the term "biometric information" as *the decrease in uncertainty about the identity*

[1] For a real-life example, see: http://money.cnn.com/2015/09/23/technology/opm-fingerprint-hack.

of a person due to a set of biometric measurements. They proposed estimating the biometric information via the *relative entropy* or *Kullback–Leibler (KL) Divergence* between the intra-class and inter-class biometric feature distributions. Takahashi and Murakami [6] adopted a similar approach to [8], except that they used *comparison score* distributions instead of *feature* distributions, since this ensures that the whole recognition pipeline is considered when estimating the amount of discriminative biometric information in the system. Around the same time, Sutcu et al. [9] adopted the same method as that employed in [6], with an important difference: they used a Nearest Neighbour (NN) estimator for the KL divergence, thereby removing the need to establish models for the comparison score distributions prior to computing the relative entropy.

This paper adopts the approach proposed in [9] to estimate the amount of discriminatory information in finger vein biometrics. We show that the Relative Entropy (RE) metric is equivalent to the Equal Error Rate (EER) in terms of enabling us to rank finger vein biometric systems according to their expected recognition accuracy. This suggests that the RE metric can provide a reliable estimation of the amount of discriminatory information in finger vein recognition systems. We additionally propose a Normalised Relative Entropy (NRE) metric to help us gain a more intuitive understanding of the significance of RE values and to allow us to fairly benchmark the REs of different biometric systems. The new metric can be used in conjunction with the EER to determine the best-performing biometric system.

The remainder of this chapter is structured as follows. Section 17.2 explains the adopted RE metric in more detail. Section 17.3 presents our results for the RE of finger vein patterns and shows how this metric can be used to rank finger vein recognition systems in comparison with the EER. Section 17.4 proposes the new NRE metric and presents NRE results on various finger vein recognition systems. Section 17.5 discusses how the NRE could be a useful complement to the EER in benchmarking the discrimination capabilities of different biometric systems, and we also present two issues that must be considered when calculating the RE and NRE in practice. Section 17.6 concludes this chapter and proposes a primary direction for future work.

17.2 Measuring Biometric Information via Relative Entropy

Let us say that $G(x)$ represents the probability distribution of *genuine* (mated) comparison scores in a biometric recognition system, and $I(x)$ represents the probability distribution of *impostor* (non-mated) comparison scores. The RE between these two distributions is then defined in terms of the KL divergence as follows:

$$D(G||I) = \sum_{i=1}^{n} G(x_i) \log_2 \frac{G(x_i)}{I(x_i)} \tag{17.1}$$

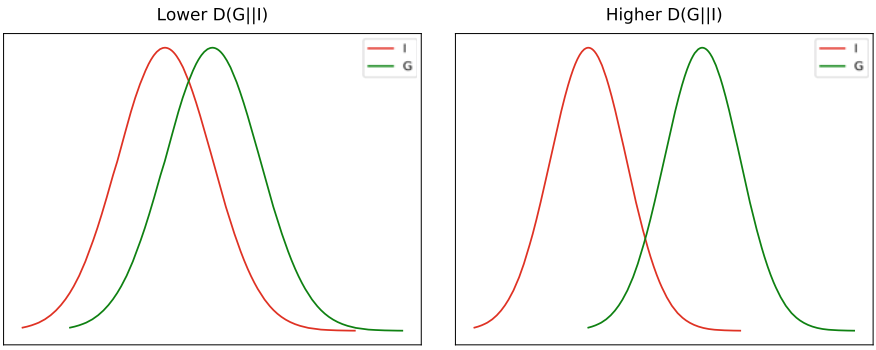


Fig. 17.1 Examples of G and I relationships producing lower and higher $D(G||I)$ values

In information-theoretic terms, $D(G||I)$ tells us the number of extra bits that we would need to encode samples from G when using a code based on I, compared to simply using a code based on G itself. Relating this to our biometric system, we can think of $D(G||I)$ as providing some indication of how closely our genuine score distribution corresponds to our impostor score distribution. The worse the match, the *higher* the $D(G||I)$ value and the easier it is to tell the two distributions apart. Consequently, the higher the RE, the easier it should be for our biometric recognition system to differentiate between genuine users and impostors based on their corresponding comparison scores, and thus the better the expected recognition accuracy. Figure 17.1 shows a simple illustration of what the relationship between G and I might look like for lower and higher $D(G||I)$ values.

One issue with using Eq. (17.1) to estimate the RE is evident when we consider what is represented by n. Technically, n is meant to denote the total number of comparison scores, and it is expected that the G and I distributions extend over the same range of scores. This, however, is not usually the case, since the overlap between the two distributions should only be partial. One consequence of this is that we will have at least one division by 0, for the range where $I(x) = 0$ but $G(x) \neq 0$. The result will be $D(G||I) = \infty$. This makes sense theoretically, since if a score does not exist in I then it is impossible to represent it using a code based on I. For our purposes, however, an RE of ∞ does not tell us much, since we already expect only partial overlap between G and I. So, we would like our RE metric to generate a finite number to represent the amount of information in our biometric recognition system.

Another issue with Eq. (17.1) is that this approach requires us to produce models for the genuine and impostor score distributions, G and I. Since the number of scores we have access to is generally not very large (this is particularly likely to be the case for genuine scores), it may be difficult to generate accurate models for the underlying score distributions.

In light of the issues mentioned above, Sutcu et al. [9] proposed approximating the RE using the NN estimator from [10]. Let $s_g^1, \ldots, s_g^{N_g}$ and $s_i^1, \ldots, s_i^{N_i}$ represent the comparison scores from the sets of genuine and impostor scores, respectively. Further,

let $d_{gg}(i) = \min_{j \neq i} ||s_g^i - s_g^j||$ represent the distance between the genuine score s_g^i and its nearest neighbour in the set of genuine scores, and let $d_{gi}(i) = \min_j ||s_g^i - s_i^j||$ denote the distance between the genuine score s_g^i and its nearest neighbour in the set of impostor scores. Then the NN estimator of the KL divergence is defined as

$$\hat{D}(G||I) = \frac{1}{N_g} \sum_{i=1}^{N_g} \log_2 \frac{d_{gi}(i)}{d_{gg}(i)} + \log_2 \frac{N_i}{N_g - 1} \tag{17.2}$$

Using Eq. (17.2), we can estimate the RE of a biometric system using the genuine and impostor comparison scores directly, without establishing models for the underlying probability densities. Moreover, using the proposed KL divergence estimator, we can circumvent the issue of not having complete overlap between the genuine and impostor score distributions. For these reasons, this is the approach we adopted to estimate the amount of information in finger vein patterns.

17.3 Relative Entropy of Finger Vein Patterns

We used the NN estimator approach from [9] to estimate the RE of finger vein patterns.[2] Section 17.3.1 describes our adopted finger vein recognition systems, and Sect. 17.3.2 presents our RE results for finger vein patterns.

17.3.1 Finger Vein Recognition Systems

We used two public finger vein databases for our investigation: VERA[3] [11] and UTFVP[4] [12]. VERA consists of two images for each of 110 data subjects' left and right index fingers, which makes up 440 samples in total. UTFVP consists of four images for each of 60 data subjects' left and right index, ring and middle fingers, which makes up 1,440 samples in total. Both databases were captured using the same imaging device, but with slightly different acquisition conditions. Figure 17.2 shows an example of a finger image from each database.

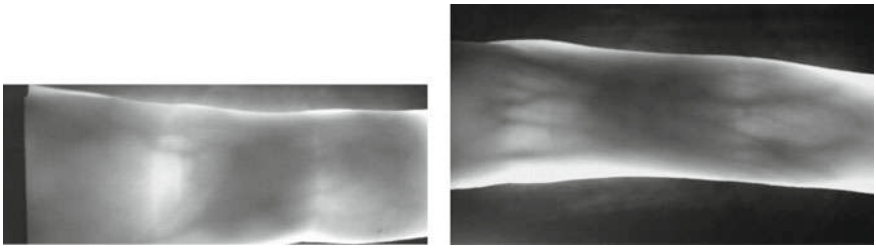
Finger vein patterns were extracted and compared using the `bob.bio.vein` PyPI package.[5] To extract the vein patterns from the finger images in each database, the fingers were first cropped and horizontally aligned as per [13, 14]. Next, the finger vein pattern was extracted from the cropped finger images using three well-known

[2] Code available at https://gitlab.idiap.ch/bob/bob.chapter.fingerveins_relative_entropy.

[3] https://www.idiap.ch/dataset/vera-fingervein.

[4] http://scs.ewi.utwente.nl/downloads/show,Finger%20Vein/.

[5] https://pypi.python.org/pypi/bob.bio.vein.



(a) VERA (b) UTFVP

Fig. 17.2 Examples of finger images from the VERA and UTFVP databases. Note that the UTFVP images are larger in size, as shown in this figure

feature extractors: Wide Line Detector (WLD) [14], Repeated Line Tracking (RLT) [15] and Maximum Curvature (MC) [16].

The comparison between the extracted finger vein patterns was performed separately for each extractor, using the algorithm proposed in [15]. This method is based on a cross-correlation between the enrolled finger vein template and the probe template obtained during verification. The resulting comparison scores lie in the range [0, 0.5], where 0.5 represents maximum cross-correlation and thus a perfect match.

17.3.2 Relative Entropy of Finger Veins

We used Eq. (17.2) to calculate the RE of finger vein patterns[6] for each of the three feature extractors (WLD, RLT, and MC) on both the VERA and UTFVP databases. One issue we faced when implementing this equation was dealing with the case where the $d_{gg}(i)$ and/or $d_{gi}(i)$ terms were zero. If $d_{gi}(i) = 0$ (regardless of what value $d_{gg}(i)$ takes), this would result in $\hat{D}(G||I) = -\infty$, whereas $d_{gg}(i) = 0$ (regardless of what value $d_{gi}(i)$ takes) would result in $\hat{D}(G||I) = \infty$. This is one of the issues we wanted to circumvent by using the NN estimator in the first place! Neither the paper that proposed the NN estimator for KL divergence [10], nor the paper that proposed using this estimator to calculate the RE of biometrics [9], suggests how to proceed in this scenario. So, we decided to add a small value (ϵ) of 10^{-10} to every $d_{gg}(i)$ and $d_{gi}(i)$ term that turned out to be 0. The choice of ϵ was based on the fact that our comparison scores are rounded to 8 decimal places, so we wanted to ensure that ϵ would be smaller than 10^{-8} to minimise the impact on the original score distribution.[7]

[6]Note: $RE = \hat{D}(G||I)$.

[7]This choice of ϵ may not necessarily be optimal, but it seems sensible.

Table 17.1 Relative Entropy (RE) and Equal Error Rate (EER) for different extractors on the VERA and UTFVP databases. The RE and EER ranks refer to the rankings of the three extractors (separately for each database) in terms of the highest RE and lowest EER, respectively

DB	Extractor	RE	EER (%)	RE rank	EER rank
VERA	WLD	11.8	9.5	2	2
VERA	RLT	4.2	24.3	3	3
VERA	MC	13.2	4.3	1	1
UTFVP	WLD	18.9	2.7	2	2
UTFVP	RLT	18.0	3.2	3	3
UTFVP	MC	19.5	0.8	1	1

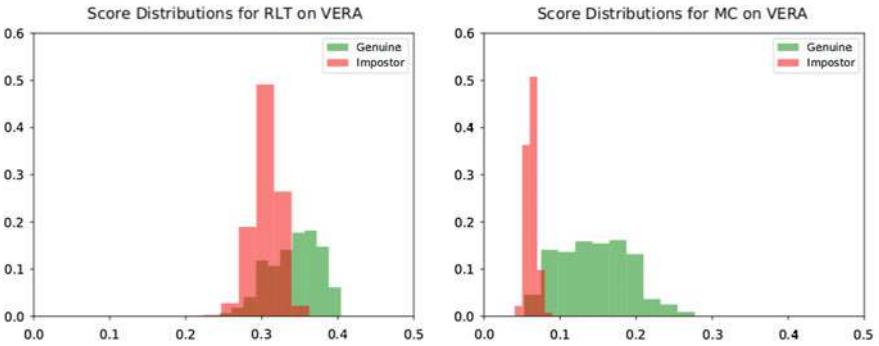


Fig. 17.3 Genuine and impostor score distributions corresponding to the lowest (left) and highest (right) RE values for the VERA database from Table 17.1

For this experiment, a comparison score was calculated between a finger vein template and *every other* finger vein template in the database. The resulting RE values are summarised in Table 17.1, along with the corresponding EERs.[8]

We can interpret the RE results in Table 17.1 as providing an indication of how many bits of discriminatory information are contained in a particular finger vein recognition system. For example, we can see that using the RLT extractor on the VERA database results in a system with only 4.2 bits of discriminatory information, while the MC extractor on the same database contains 13.2 bits of discriminatory information. Figure 17.3 illustrates the genuine and impostor score distributions for these two RE results.

Since our results show the RE to be dependent upon both the feature extractor and database adopted, it would be misleading to claim a universal finger vein RE estimate; rather, it makes more sense for the RE to be system-specific.

[8]Note that we have chosen to compare the RE to the EER, because the EER is a widely used metric for evaluating the overall recognition accuracy (in terms of the trade-off between the False Match Rate (FMR) and False Non-Match Rate (FNMR)) of a biometric recognition system. The comparison seems appropriate, since RE aims to provide us with an idea of a biometric system's overall discrimination capability.

Intuitively, we can see that, the higher the RE, the greater the amount of discriminatory information, and thus the greater the expected recognition capabilities of the underlying system. This intuition is confirmed when we compare the REs and EERs of the different systems in Table 17.1, in terms of the RE-based versus EER-based rankings. From this analysis, it is evident that the ranking of the three extractors for each database is the same regardless of whether that ranking is based on the RE or the EER. In particular, MC has the highest RE and lowest EER, while RLT has the lowest RE and highest EER. This implies that the most discriminatory information is contained in finger vein patterns that have been extracted using the MC extractor, and the least discriminatory information is contained in RLT-extracted finger veins. These results suggest the possibility of using the REs of different finger vein recognition systems to rank the systems according to the amount of discriminatory information and thus their expected recognition accuracies. Consequently, it appears reasonable to conclude that the RE estimator is a reliable indicator of the amount of discriminatory information in a finger vein recognition system.

While RE quantifies the amount of discriminatory information in a biometric system, it is difficult to gauge what exactly this number, on its own, means. For example, what exactly does x bits of discriminatory information signify, and is a y-bit difference in the REs of two biometric systems significant? Furthermore, benchmarking different biometric systems in terms of their RE is not straightforward, since the RE estimate depends on both the comparison score range as well as on the number of genuine (N_g) and impostor scores (N_i) for each database and experimental protocol. Consequently, REs reported for different biometric systems usually do not lie in the same $[RE_{\min}, RE_{\max}]$ range.[9] To help us better understand the meaning of the RE metric in the context of a biometric system, as well as to enable fair cross-system RE benchmarking, Sect. 17.4 adapts Eq. (17.2) to propose a *normalised* RE metric.

17.4 Normalised Relative Entropy

This section proposes a normalised version of the RE (NRE), based on the NN estimator in Eq. (17.2). The reason for this normalisation is to help us interpret the RE in a more intuitive way, and to enable fair benchmarking of different biometric systems in terms of their RE.

We propose using the well-known "min–max" normalisation formulated by Eq. (17.3):

$$NRE = \frac{RE - RE_{\min}}{RE_{\max} - RE_{\min}} \tag{17.3}$$

[9]For the finger vein systems we used, the comparison scores for both the VERA and UTFVP databases lie in the same range of [0, 0.5]. However, the N_g values across the two databases are different as are the N_i values. Consequently, the $[RE_{\min}, RE_{\max}]$ range is not the same for both databases, meaning that we cannot fairly compare the RE results across the two databases.

In Eq. (17.3), $RE_{\min}$ and $RE_{\max}$ refer to the minimum and maximum possible RE values, respectively, for a particular biometric system. Thus, we need to begin by establishing $RE_{\min}$ and $RE_{\max}$. In this formulation, we assume that comparison scores are similarity values, such that small scores indicate low similarity and large scores indicate high similarity. Keeping this in mind, the minimum RE would occur when all d_{gi} values are zero and all d_{gg} values are as large as possible. Therefore, for each genuine score, there would need to be at least one impostor score with exactly the same value, and all the genuine scores would need to be spread apart as far as possible. Let us say that all scores lie in the range $[s_{\min}, s_{\max}]$, and that the number of genuine scores for a particular database and experimental protocol is denoted by N_g. Then, the maximum possible d_{gg} value would be $\frac{s_{\max}-s_{\min}}{N_g}$. By adapting Eq. (17.2), our equation for the minimum RE thus becomes

$$RE_{\min} = \frac{1}{N_g}\sum_{i=1}^{N_g} \log_2 \frac{0}{\frac{s_{\max}-s_{\min}}{N_g}} + \log_2 \frac{N_i}{N_g - 1} \tag{17.4}$$

If we now tried to solve Eq. (17.4), we would get $RE_{\min} = -\infty$, because of the 0 d_{gi} term. Since this is an impractical result for measuring the (finite) amount of information in a biometric system, we replace the 0 with ϵ. Furthermore, we can see that the division by N_g gets cancelled out by the summation across N_g, so we can simplify Eq. (17.4) as follows:

$$RE_{\min} = \log_2 \frac{\epsilon}{\frac{s_{\max}-s_{\min}}{N_g}} + \log_2 \frac{N_i}{N_g - 1} \tag{17.5}$$

Equation (17.5) thus becomes the final $RE_{\min}$ equation.

The maximum RE would occur when all d_{gi} values are as large as possible and all d_{gg} values are zero. The only way this could occur would be if all the genuine scores took on the largest possible value, $s_{\max}$, and all the impostor scores took on the smallest possible value, $s_{\min}$. In this case, the genuine and impostor score sets would be as different as possible. By adapting Eq. (17.2), we thus get the following equation for the maximum RE:

$$RE_{\max} = \frac{1}{N_g}\sum_{i=1}^{N_g} \log_2 \frac{s_{\max} - s_{\min}}{0} + \log_2 \frac{N_i}{N_g - 1} \tag{17.6}$$

If we tried to solve Eq. (17.6), we would get $RE_{\max} = \infty$ due to the 0 term in the denominator. So, once again we replace the 0 term with ϵ. Furthermore, just like we did for Eq. (17.4), we can simplify Eq. (17.6) by removing the N_g division and summation. Our final equation for $RE_{\max}$ thus becomes

$$RE_{\max} = \log_2 \frac{s_{\max} - s_{\min}}{\epsilon} + \log_2 \frac{N_i}{N_g - 1} \tag{17.7}$$

We can now use Eq. (17.3), with Eq. (17.5) for $RE_{\min}$ and Eq. (17.7) for $RE_{\max}$, to calculate the NRE of a particular biometric system.

Due to the "min–max" operation in Eq. (17.3), the NRE will lie in the range [0.00, 1.00]. We can thus interpret the NRE as follows. An NRE of 0.00 would suggest that the system in question contains zero discriminative information (i.e. recognition would actually be impossible), whereas an NRE of 1.00 would indicate that the system contains the maximum amount of discriminative information possible for that system (i.e. the recognition accuracy would be expected to be perfect).

Figure 17.4 illustrates what the impostor and genuine comparison score distributions might look like for a minimum NRE system and a maximum NRE system, when the comparison score range is [0, 0.5] (i.e. the score range corresponding to our finger vein recognition systems).

In general, therefore, we can look at the NRE as providing an indication of the proportion of the maximum amount of discriminatory information that the corresponding biometric system contains. An NRE of 0.50, for example, would indicate that the biometric system achieves only 50% of the maximum attainable recognition accuracy. Therefore, the higher the NRE, the better the expected recognition accuracy of the biometric system we are measuring.

Table 17.2 shows the NRE results for our aforementioned finger vein recognition systems. Note that, for these finger vein systems: $s_{\min} = 0$; $s_{\max} = 0.5$; $N_g = 440$ for VERA; $N_g = 4,320$ for UTFVP; $N_i = 192,720$ for VERA; $N_i = 2,067,840$ for UTFVP.

Note that the first column of Table 17.2 refers to the finger vein recognition system constructed using the specified database and feature extractor. We have pooled the databases and extractors into "systems" now to indicate that the NRE values can be benchmarked *across systems* (as opposed to, for example, in Table 17.1, where the databases were separate to indicate that RE-based benchmarking of the different extractors should be *database-specific*).

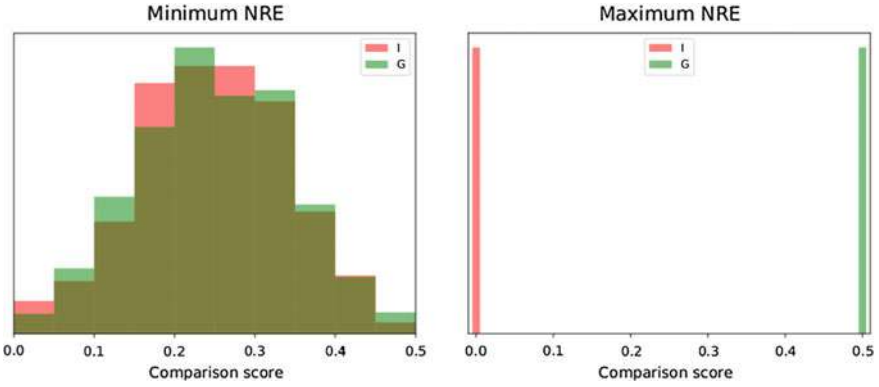


Fig. 17.4 Illustration of impostor and genuine score distributions for a minimum and a maximum NRE system, when the comparison score range is [0, 0.5]

Table 17.2 Relative Entropy (RE) and Normalised Relative Entropy (NRE) for different finger vein recognition systems

System	RE	NRE
VERA-WLD	11.8	0.48
VERA-RLT	4.2	0.34
VERA-MC	13.2	0.50
UTFVP-WLD	18.9	0.58
UTFVP-RLT	18.0	0.56
UTFVP-MC	19.5	0.59

As an example of how the NRE results from Table 17.2 can be interpreted, let us compare the NRE of VERA-RLT to that of UTFVP-MC. The NRE of 0.34 for VERA-RLT tells us that this system achieves only 34% of the maximum attainable discrimination capability. Comparatively, the UTFVP-MC system contains 59% of the maximum amount of discriminative information. So, we could conclude that the UTFVP-MC finger vein recognition system contains 25% more discriminatory information than the VERA-RLT system.

Using the NRE also helps us gauge the significance of the differences in the REs across different biometric systems. For example, if we look at the RE on its own for the UTFVP-WLD and UTFVP-MC systems in Table 17.2, we can see that the latter system's RE is 0.6 bits larger than the former system's RE. It is difficult to tell, however, whether or not this is a significant difference. If we then look at the NREs of the two systems, we can see that their difference is only 0.01. This indicates that the 0.6-bit difference between the two systems' REs is not too significant in terms of the proportion of the maximum discriminatory information the two systems contain. On the other hand, the 15.3-bit difference in the REs between the VERA-RLT and UTFVP-MC systems seems much more significant, and we may be tempted to conclude that the latter system contains about five times more discriminative information than the former system. Looking at the two systems' NREs, we do see a fairly significant difference, but we would have to conclude that the UTFVP-MC system contains not five times, but two times, more discriminative information than the VERA-RLT system.

In this section, we have shown how the NRE can be used for RE-based benchmarking of different finger vein recognition systems, for which comparison scores were evaluated on different databases. The main reason for using the NRE in our case was thus to conduct fair cross-database system benchmarking. Our proposed NRE metric, however, can also be used to fairly benchmark the REs of systems based on different biometric modalities, tested on different databases using different experimental protocols. For example, part of our future work will involve benchmarking the NRE of our best finger vein recognition system, UTFVP-MC, against NREs of systems based on different types of biometrics. This makes the proposed NRE metric a flexible tool for both quantifying and benchmarking the amount of discriminative information contained in different biometric systems.

17.5 Discussion

In this section, we begin by presenting a discussion on an important aspect of the NRE, which supports its adoption in the biometrics community. We then discuss two potential issues that may arise when calculating the NRE, and we suggest the means of dealing with them. Sections 17.5.1, 17.5.2 and 17.5.3, respectively, tackle these three discussion points.

17.5.1 NRE as a Complement to EER

So far, we have shown how the RE can be used to measure the amount of discriminatory information in finger vein recognition systems. We also proposed the NRE metric to fairly benchmark the REs across different biometric systems. In this section, we discuss how an NRE estimate could complement the EER to provide a more complete picture of the performance of a biometric recognition system.

In Sect. 17.2, we explained how, in the context of a biometric recognition system, the RE metric provides some indication of how closely our genuine score distribution matches our impostor score distribution. Let us explore the meaning of this by considering Eq. (17.2). Equation (17.2) tells us that we are attempting to estimate the relative entropy of a set of genuine comparison scores (G) in terms of a set of impostor comparison scores (I). In other words, we wish to quantify the "closeness" of these two sets[10] of scores. The d_{gi} and d_{gg} terms represent the distance between a genuine score and its closest score in the set of impostor and genuine scores, respectively. Larger d_{gi} values will result in *larger* RE results, whereas larger d_{gg} values will result in *smaller* RE results.[11] We can thus see that larger REs favour a larger inter-class variance (i.e. greater separation between genuine comparison trials and impostor trials) and a smaller intra-class variance (i.e. smaller separation between multiple biometric samples from the same biometric instance). This makes the RE suitable as a measure of the performance of a biometric recognition system: the larger the RE value, the better the recognition accuracy. The best (highest) RE would, therefore, be obtained in the case where all the d_{gi} values are as large as possible, while the d_{gg} values are as small as possible, and vice versa for the worst (lowest) RE.

The RE metric thus informs us about two things: how far genuine scores are from impostor scores, and how far genuine scores are from each other. Consider the case where we have a set of impostor scores, I, and a set of genuine scores, G. The larger the intersection between I and G, the smaller the d_{gi} values and thus the lower the RE. Conversely, the smaller the intersection between the two sets, the greater the d_{gi} values and thus the higher the RE. So far, the RE metric appears to tell us the same thing as the EER, since a smaller EER indicates less overlap between genuine and

[10] Note: We are purposely using the word "set" as opposed to "distribution", since the NN estimator in Eq. (17.2) works directly on the scores as opposed to distributions representing the scores.

[11] Assume constant N_g and N_i values.

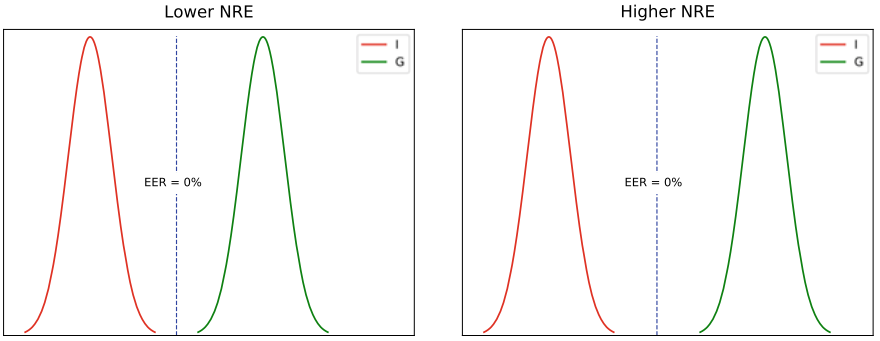


Fig. 17.5 Two biometric systems with the same EER of 0%, but where the system on the right has greater separation between the impostor and genuine comparison scores, and thus a higher NRE than the system on the left

impostor comparison scores, while a larger EER indicates more overlap. Where the two metrics differ, however, is in the scenario where I and G are completely separated. In this case, the further apart the two sets of scores are the higher the resulting RE. The EER, however, would be 0% regardless of whether the separation is small or large. Imagine if we had to benchmark two biometric systems, both of which had complete separation between the genuine and impostor comparison scores, but where for one system the separation was much larger than for the other, as illustrated[12] in Fig. 17.5. If we considered only the EER, it would indicate that the two systems are the same (i.e. both have an EER of 0%). The NRE,[13] however, would clearly indicate that the system with greater separation is better in terms of distinguishing genuine trials from impostors, since the NRE value would be higher for that system. In this case, complementing the EER with an NRE estimate would provide a more complete picture of the system comparison. This could come in useful particularly in situations where the data used for testing the biometric system was collected in a constrained environment, in which case an EER of 0% could be expected. The NRE, on the other hand, would provide us with more insight into the separation between the genuine and impostor score distributions.

Another example of a scenario in which the NRE metric would be a useful complement to the EER is when we have two biometric systems for which I is the same and the separation (or overlap) between I and G is the same, but G differs. In particular, in the first system the genuine scores are closer together, while in the second system the genuine scores are further apart from each other. Figure 17.6 illustrates

[12]Note: The only reason for using probability density plots in this figure is to present a cleaner illustration of our point. Probability density functions are *not* used to represent genuine and impostor score distributions for the NRE calculation.

[13]When benchmarking different biometric systems, the NRE should be used instead of the RE to ensure that the benchmarking is fair. The only exception to this rule would be in the case where the different systems had the same comparison score range, and the same N_g and N_i values, in which case the resulting REs would lie in the same $[RE_{\min}, RE_{\max}]$ range.

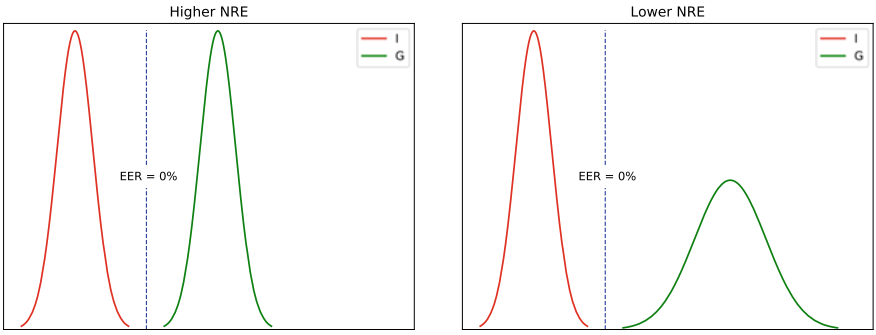


Fig. 17.6 Two biometric systems with the same I, the same separation between I and G and thus the same EER, but with different G. In particular, G for the system on the right has a larger variance, and thus the NRE is lower to reflect this

this scenario.[14] In this case, since the separation between I and G for both systems is the same, the EER would also be the same, thereby indicating that one system is just as good as the other. The NRE, however, would be smaller for the second system due to the larger d_{gg} values. The NRE would thus indicate that the larger intra-class variance in the second system makes this system less preferable in terms of biometric performance when compared to the first system, for which the genuine scores are closer together and thus the intra-class variance is smaller. Using both NRE and EER together, we could thus conclude that, although both systems can be expected to achieve the same error rate, the system with the smaller intra-class variance would be a superior choice.

When choosing between the EER and NRE metrics for evaluating the performance of a biometric system, we would still recommend using the EER as the primary one, since it is more practical in providing us with a solid indication of our system's expected error rate. The NRE, however, would be a useful complement to the EER when we are trying to decide on the best of n biometric systems that have the *same* EER.

17.5.2 Selecting the ϵ Parameter

As mentioned in the introductory paragraph of Sect. 17.3.2, ϵ is a parameter chosen to deal with zero score differences (i.e. $d_{gg} = 0$ or $d_{gi} = 0$) in order to avoid an RE of $\pm\infty$ (which would be meaningless in the context of measuring the amount of discriminatory information in a biometric system). It is clear from Eqs. (17.2), (17.3), (17.5) and (17.7), however, that the choice of ϵ could potentially have a significant effect on the resulting RE and, therefore, NRE, particularly if the number of zero score

[14]Note: In Fig. 17.6, the EER for both systems is 0%; however, it could also be possible for both systems to have the same non-zero EER. In this case, I and G would partially overlap.

differences is large. While the number of zero score differences will be dependent on the biometric system in question and this number is, therefore, difficult to generalise, we wished to see what effect the choice of ϵ would have on the RE and NRE of our best finger vein recognition system, that obtained when using MC-extracted finger veins from the UTFVP database. Figure 17.7 shows plots of the RE and NRE versus ϵ, when ϵ is selected to lie in the range $[10^{-12}, 10^{-8}]$. For convenience, Table 17.3 summarises the RE and NRE values from Fig. 17.7.

From Fig. 17.7 and Table 17.3, we can see that, while the choice of ϵ does affect the RE and NRE to some degree (more specifically, the RE and NRE decrease as ϵ decreases[15]), this effect does not appear to be significant. So, we may conclude that, as long as the ϵ parameter is sensibly chosen (i.e. smaller than the comparison scores, but not so small that it is effectively zero), then the RE and NRE estimates should be reasonable.

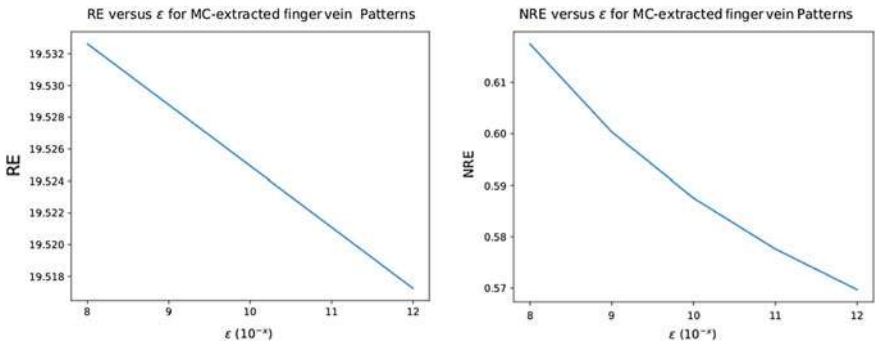


Fig. 17.7 RE versus ϵ and NRE versus ϵ, when ϵ takes on different values in the range $[10^{-12}, 10^{-8}]$, for MC-extracted finger vein patterns in the UTFVP database

Table 17.3 RE and NRE for MC-extracted finger veins from UTFVP, when ϵ is varied in the range $[10^{-12}, 10^{-8}]$. Note that, for consistency with Table 17.2, RE and NRE values are rounded to 1 d.p. and 2 d.p., respectively

ϵ	RE	NRE
10^{-8}	19.5	0.62
10^{-9}	19.5	0.60
10^{-10}	19.5	0.59
10^{-11}	19.5	0.58
10^{-12}	19.5	0.57

[15] In general, the RE, and thus the NRE, would be expected to *decrease* with a decrease in ϵ when there are more d_{gi} than d_{gg} zero score differences. Alternatively, the RE, and thus the NRE, would be expected to *increase* with a decrease in ϵ when there are more d_{gg} than d_{gi} zero score differences.

17.5.3 Number of Nearest Neighbours

The method proposed in [9] to estimate the RE of biometrics uses only the *first* nearest genuine and impostor neighbours of each genuine score. An issue with this approach is that it makes the RE estimate highly dependent on any single score, even if that score is an outlier. This might be particularly problematic if we do not have a large number of scores to work with, which is often the case.

It seems that a safer approach would be to use k nearest neighbours, where $k > 1$, then average the resulting $d_{gg}(i)$ and $d_{gi}(i)$ values over these k neighbours prior to estimating the RE. This would introduce some smoothing to the underlying score distributions, thereby stabilising the RE estimates. While the effect of k on the RE, and therefore NRE, is difficult to generalise since it would, in practice, be dependent on the biometric system in question, we wished to test the effect of the choice of k on the RE and NRE of our best finger vein recognition system, that obtained when using MC-extracted finger veins from the UTFVP database. Figure 17.8 shows plots of the RE and NRE versus k, when k increases from 1 to 5. For convenience, Table 17.4 summarises the RE and NRE values from Fig. 17.8. Note that, for this experiment, $\epsilon = 10^{-10}$, as for the RE and NRE experiments in Sects. 17.3 and 17.4.

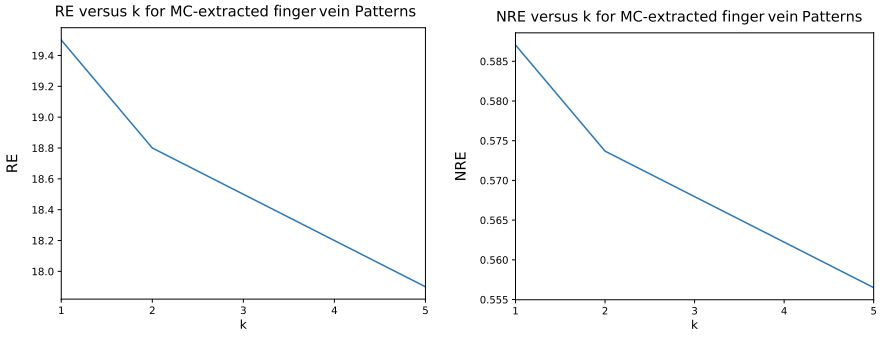


Fig. 17.8 RE versus k and NRE versus k, when k increases from 1 to 5, for MC-extracted finger vein patterns in the UTFVP database

Table 17.4 RE and NRE for MC-extracted finger veins from UTFVP, when k increases from 1 to 5. Note that, for consistency with Tables 17.2 and 17.3, RE and NRE values are rounded to 1 d.p. and 2 d.p., respectively

k	RE	NRE
1	19.5	0.59
2	18.8	0.57
3	18.5	0.57
4	18.2	0.56
5	17.9	0.56

From Fig. 17.8 and Table 17.4, it is evident that increasing k tends to decrease both the RE and NRE, but the decrease is not drastic for $k \leq 5$. This decrease makes sense, since a larger k means a greater degree of smoothing, which decreases the effects of individual comparison scores. Another consequence of using a larger k would be that the effect of the ϵ parameter on RE and NRE would be expected to be less pronounced. This is because a larger k means that a larger number of neighbouring scores are averaged when calculating the RE and NRE, so we are less likely to encounter zero average scores than in the scenario where only *one* nearest neighbouring score is considered. Keeping the aforementioned points in mind, it is important to sensibly tune the k and ϵ parameters depending on the biometric system in question (e.g. if there are outlier scores, use $k > 1$, and select ϵ based on the score precision, as discussed in Sect. 17.5.2). Furthermore, we urge researchers adopting the RE and NRE measures to be transparent about their selection of these parameters to ensure fair system comparisons across the biometrics community.

Note that the NN estimator on which Eq. (17.2) is based [10] *is* actually a k-NN estimator, where k denotes the number of nearest neighbours. It is not clear, however, whether the proposed k-NN estimator is based on *averaging* the k nearest neighbouring scores, as we have done for Fig. 17.8 and Table 17.4, or whether the authors meant that *only* the kth neighbour should be used. If their intention is the latter, then our averaging approach represents an effective new way of stabilising the k-NN estimator for RE measures.

17.6 Conclusions and Future Work

This chapter represents the first attempt at estimating the amount of information in finger vein biometrics in terms of score-based Relative Entropy (RE), using the previously proposed Nearest Neighbour estimator. We made five important contributions.

First, we showed that the RE estimate is system-specific. In our experiments, the RE differed across finger vein recognition systems employing different feature extractors and different testing databases. For this reason, we refrain from claiming a universal finger vein RE estimate, since this would be misleading.

Second, we showed that the RE can be used to rank different finger vein recognition systems, which are tested on the same database using the same experimental protocol (in our case, the difference was the feature extractor employed), in terms of the amount of discriminative biometric information available. The ranking was shown to be comparable to an EER-based ranking, which implies that the RE estimate is a reliable indicator of the amount of discriminatory information in finger vein recognition systems.

Third, we proposed a new metric, the Normalised Relative Entropy (NRE), to help us gauge the significance of individual RE scores as well as to enable fair benchmarking of different biometric systems (in particular, systems tested on different databases using different experimental protocols) in terms of their RE. The NRE lies in the range [0.00, 1.00] and represents the proportion of the maximum amount of

discriminatory information that is contained in the biometric system being measured. The higher the NRE, the better the system is expected to be at distinguishing genuine trials from impostors.

Fourth, we discussed how the NRE metric could be a beneficial complement to the EER in ranking different biometric systems in terms of their discrimination capabilities. The NRE would be particularly useful in choosing the best of n biometric systems that have the same EER.

Finally, we discussed two potential issues in calculating the RE and NRE, namely, the effects of the ϵ parameter and the number of nearest neighbours (k) used for computing the genuine–genuine and genuine–impostor score differences. We showed that, as long as ϵ is sensibly selected, its effect on the RE and NRE is unlikely to be significant. We also showed that increasing the number of nearest score neighbours may be expected to slightly decrease the RE and NRE, but the upside is that using a larger number of nearest neighbours would help to dilute the effects of outliers among the genuine and impostor comparison scores. We concluded by suggesting that ϵ and k be tuned according to the biometric system being evaluated and that researchers be transparent in terms of reporting their selection of these two parameters.

At the moment, our primary aim for future work in this direction is to use our proposed NRE metric to benchmark finger vein recognition systems against systems based on other biometric modalities, in terms of the amount of discriminatory information contained in each system.

Acknowledgements The authors would like to acknowledge the following sources of funding for supporting this work: the Secure Access Control over Wide Area Network (SWAN) project, the German Federal Ministry of Education and Research (BMBF) and the Hessen State Ministry for Higher Education, Research and the Arts (HMWK) within the Center for Research in Security and Privacy (CRISP).

References

1. Ye Y, Zheng H, Ni L, Liu S, Li W (2016) A study on the individuality of finger vein based on statistical analysis. In: 2016 international conference on biometrics (ICB), pp 1–5, June 2016
2. Yanagawa T, Aoki S, Ohyama T (2007) Human finger vein images are diverse and its patterns are useful for personal identification. MHF Prepr Ser 12:1–7
3. Jeffers J, Arakala A, Horadam KJ (2010) Entropy of feature point-based retina templates. In: 2010 20th international conference on pattern recognition, pp 213–216, Aug 2010
4. Daugman J (2004) How iris recognition works. IEEE Trans Circuits Syst Video Technol 14(1):21–30
5. Krivokuca V, Marcel S (2018) Towards quantifying the entropy of fingervein patterns across different feature extractors. In: 2018 IEEE 4th international conference on identity, security, and behavior analysis (ISBA), pp 1–8, Jan 2018
6. Takahashi K, Murakami T (2010) A metric of information gained through biometric systems. In: Proceedings of the 2010 20th international conference on pattern recognition. IEEE Computer Society, pp 1184–1187
7. Sutcu Y, Tabassi E, Sencar HT, Memon N (2013) What is biometric information and how to measure it? In: 2013 IEEE international conference on technologies for homeland security (HST), pp 67–72, Nov 2013

8. Adler A, Youmaran R, Loyka S (2006). Towards a measure of biometric information. In 2006 Canadian conference on electrical and computer engineering, pp 210–213, May 2006
9. Sutcu Y, Sencar HT, Memon N (2010) How to measure biometric information? In : 2010 20th international conference on pattern recognition, pp 1469–1472, Aug 2010
10. Wang Q, Kulkarni SR, Verdu S (2009) Divergence estimation for multidimensional densities via k-nearest-neighbor distances. IEEE Trans Inf Theory 55(5):2392–2405
11. Vanoni M, Tome P, El-Shafey L, Marcel S (2014) Cross-database evaluation using an open finger vein sensor. In: 2014 IEEE workshop on biometric measurements and systems for security and medical applications (BIOMS) proceedings, pp 30–35, Oct 2014
12. Ton BT, Veldhuis RNJ (2013) A high quality finger vascular pattern dataset collected using a custom designed capturing device. In: 2013 international conference on biometrics (ICB), pp 1–5, June 2013
13. Lee EC, Lee HC, Park KR (2009) Finger vein recognition using minutia-based alignment and local binary pattern-based feature extraction. Int J Imaging Syst Technol 19(3):179–186
14. Huang B, Dai Y, Li R, Tang D, Li W (2010) Finger-vein authentication based on wide line detector and pattern normalization. In: 2010 20th international conference on pattern recognition, pp 1269–1272, Aug 2010
15. Miura N, Nagasaka A, Miyatake T (2004) Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach Vis Appl 15(4):194–203
16. Miura N, Nagasaka A, Miyatake T (2007) Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Trans Inf Syst 90(8):1185–1194

Index

A. Uhl et al. (eds.), *Handbook of Vascular Biometrics*, Advances in Computer Vision and Pattern Recognition, https://doi.org/10.1007/978-3-030-27731-4

D

E