Stellenbosch
UNIVERSITY
IYUNIVESITHI
UNIVERSITEIT

Hand vein-based biometric authentication using neural networks

by

Emile Beukes

Dissertation presented for the degree of Doctor of Philosophy (Applied Mathematics) in the Faculty of Science at Stellenbosch University

Supervisor: Dr Johannes Coetzer

February 2024

The financial assistance of the Division of Applied Mathematics of Stellenbosch University and Praelexis (PTY) LTD towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the aforementioned institutions.

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 2024/02/06

Abstract

Hand vein-based biometric authentication using neural networks

E. Beukes

Division of Applied Mathematics,
Stellenbosch University,
Private Bag X1, Matieland 7602, South Africa.

Dissertation: PhD (Applied Mathematics)

February 2024

The feasibility of employing convolutional neural networks for the purpose of authenticating an individual based on a near infra-red image of his/her dorsal hand vein pattern is investigated in this study. The proficiency of different architectural designs associated with similarity measure networks (SMNs), in particular two-channel SMNs and Siamese SMNs, are compared. Four different combinations of neural network layers are investigated for each of the aforementioned SMNs. Three different levels of preprocessing are applied to the hand vein images in order to investigate the relevance of information surrounding the actual hand veins on the proficiency of the networks. The proficiency of the proposed systems is gauged within the context of two real-world scenarios, namely the individual *dependent* scenario (IDS) and the individual *independent* scenario (IIS). A tailor-made network is trained for *each* client *during* enrolment in mere minutes within the context of the IDS, while a *single* network is trained in a *once-off* fashion *prior* to the enrolment of *any* clients within the context of the IIS. Two publicly available hand vein databases namely the Bosphorus and Wilches databases are investigated within the context of this study. An *artificially generated* hand vein database, namely the GenVeins database, is developed in this study for the purpose of acquiring a set of training individuals that is large enough so as to be *representative* of the entire population. The motivation behind the creation of the GenVeins database constitutes the fact that experimental results indicate that system proficiency is *severely* impaired when training on an *insufficient* number of *different* individuals within the context of the IIS. The systems proposed in this study are therefore considered *implementation-ready* in the sense that they are either trained in a (1) *tailor-made* fashion for *each* client enrolled into the system *in real time* or in a (2) *once-off* fashion on a set of *fictitious* individuals that is *sufficiently* representative of the *entire* population. The proposed systems do therefore *not merely* serve as so-called *proofs-of-concept* (POCs) in which a system is trained *and* tested on the *same* set of individuals. These POCs are clearly not feasible within the context of *any* real world scenario.

Uittreksel

Handbloedvat-gebaseerde biometriese verifikasie met neuraalnetwerke

E. Beukes

Afdeling Toegepaste Wiskunde,
Universiteit van Stellenbosch,
Privaatsak X1, Matieland 7602, Suid-Afrika.

Proefskrif: PhD (Toegepaste Wiskunde)

Februarie 2024

Die uitvoerbaarheid van die gebruik van konvolusie neuraalnetwerke om ’n individu op grond van ’n naby-infrarooi beeld van sy of haar dorsale handbloedvatpatroon te identifiseer word in hierdie studie ondersoek. Die doeltreffendheid van twee sogenaamde ooreenstemmingsmaatstafnetwerke (SMNs), naamlik twee-kanaal SMNs en Siamese SMNs, word vergelyk. Vier verskillende kombinasies van neuraalnetwerk-lae word vir elk van die bogenoemde SMNs ondersoek. Drie verskillende vlakke van voorverwerking word op die handbloedvatbeelde toegepas ten einde die relevantheid van inligting rondom die werklike handare op die doeltreffendheid van die netwerke te ondersoek. Die doeltreffendheid van die voorgestelde stelsels word binne die konteks van twee regte-wêreld scenarios, naamlik die individu-*afhanklike* scenario (IDS) en die individu-*onafhanklike* scenario (IIS), ondersoek. ’n *Nommerpas* netwerk word *tydens* inskrywing binne die konteks van die IDS vir *elke* kliënt binne ’n paar minute afgerig, terwyl ’n *enkele* netwerk *vooraf* binne die konteks van die IIS afgerig word *voordat* enige kliënte ingeskryf word. Twee openbare handbloedvatdatabasisse, naamlik die Bosphorus en Wilches databasisse, word in hierdie studie ondersoek. ’n *Kunsmatig-gegenereerde* handbloedvatdatabasis, naamlik die GenVeins-databasis, is vir hierdie studie ontwikkel ten einde ’n stel afrig-individue te bekom wat groot genoeg is om *verteenwoordigend* van die hele bevolking te wees. Die motivering vir die skep van die GenVeins-databasis is die feit dat eksperimentele resultate aantoon dat stelseldoeltreffendheid *ernstig* belemmer word wanneer dit met ’n *ontoereikende* getal *verskillende* individue binne die konteks van die IIS afgerig word. Die stelsels wat in hierdie studie voorgestel word, word dus as *implementasie-gereed* beskou in die sin dat hulle óf op ’n (1) *nommerpas* wyse vir *elke* kliënt *tydens* inskrywing afgerig word, óf op ’n (2) *eenmalige* wyse op ’n stel *fiktiewe* individue wat *verteenwoordigend* van die *hele* bevolking is afgerig word *voordat* enige klient ingeskryf word. Die voorgestelde stelsels dien dus *nie bloot* as sogenaamde *konsepbewyse* (POCs) waarin ’n stelsel afgerig *en* getoets word op *dieselfde* stel individue nie. Hierdie POCs is duidelik nie uitvoerbaar binne die konteks van *enige* regte-wêreld scenario nie.

Acknowledgements

I would like to express my sincere gratitude to the following people and organisations:

1. My promoter and supervisor, Dr Johannes Coetzer. I first got to know Hanno during my Honours degree in a module called Digital Image Processing of which he was the lecturer. I really enjoyed learning about different image processing techniques and their implementation, and decided to pursue a Masters degree and subsequently a Doctoral degree in the same field under his supervision. I have always admired his sophistication towards the scientific process and scientific writing. He is always able to phrase a statement in such a way that it is clear and unambiguous to the reader, no matter the complexity of the content. I am extremely grateful for his unwavering patience, guidance and support over the past number of years. Working with him has significantly improved my skills as a scientist.

2. The Computer Science Division at Stellenbosch University, for enabling the free utilisation of one of their computationally powerful servers in order to conduct the experiments in this study.

3. The Bogazici University, for allowing the use of their hand vein data set in this study.

4. Praelexis (PTY) LTD, for their financial contribution towards this study, and my coworkers for their motivational support and assistance.

5. The Division of Applied Mathematics at Stellenbosch University, for their financial contribution towards this study.

6. My parents, fiancé and friends, for their continuous motivation and support throughout the duration of this study.

Contents

List of Figures

List of Tables

List of Abbreviations

AER	Average error rate
AER_{min}	The minimum average error rate
ANN	Artificial neural network
ASGD	A-stochastic gradient descent
BME	Benchmark experiment
BTH	Black-top hat
CLAHE	Contrast-limited adaptive histogram equalisation
CNN	Convolutional neural network
EER	Equal error rate
FAR	False acceptance rate
FAR_{zero}	The zero false acceptance rate
FE	Feature extractor
FRR	False rejection rate
FRR_{zero}	The zero false rejection rate
FFNN	Feed forward neural network
FGE	First GenVeins experiment
FRR	False rejection rate
IDE	Individual dependent experiment
IDS	Individual dependent scenario
IIS	Individual independent scenario
LoG	Laplacian of Gaussian
NCC	Normalised cross-correlation
NIR	Near infra-red
ROI	Region of interest
SE	Structuring element
SEN	Sensitivity
SPE	Specificity
SGE	Second GenVeins experiment
SHAP	Shapley Additive Explanations
SMN	Similarity measure network
SOTA	State of the art

Nomenclature

a_i	Activation of neuron i
w_{ij}	Weight of the connection between neurons i and j
b_i	Bias term associated with a_i
P	Number of positive samples used in a specific trial
N	Number of negative samples used in a specific trial
tp	An instance of a true positive
tn	An instance of a true negative
fp	An instance of a false positive
fn	An instance of a false negative
TP	Total number of true positive instances in a specific trial
TN	Total number of true negative instances in a specific trial
FP	Total number of false positive instances in a specific trial
FN	Total number of false negative instances in a specific trial

Chapter 1

Introduction

1.1 Background

The purpose of this study is to develop a system which automatically authenticates a questioned individual by comparing a near infra-red (NIR) image of his/her *dorsal hand veins* to a reference NIR image belonging to the claimed individual. Misclassifications and response time should be minimised, while *also* ensuring a non-intrusive protocol in order to motivate the adoption of the proposed system into the real world.

This research is motivated by the fact that identity verification systems which are based on popular authentication factors such as handwritten signatures or personal identification numbers (PINs) are *highly* susceptible to fraud. For example, handwritten signatures may be imitated either skilfully or randomly [4] [5] [6] [7], while PINs and bank cards may be stolen either at an automated teller machine (ATM) or at a point of sale machine (POSM). Hand veins on the other hand *cannot* be imitated *nor* stolen, since the individual in question must *willingly* present his/her hand to the system in *real time* in order to be authenticated. At the same time, the acquisition of a NIR hand vein image is sufficiently non-intrusive and timely efficient in order for such a system to be employed either at an ATM or at a POSM. The structure of an individual's hand veins also does not change over time [8], only in size [9], which can be mitigated by employing appropriate normalisation techniques.

A number of hand vein-based authentication systems are proposed in this study (see Section 1.4.1). A suitable region of interest (ROI) is first extracted from a NIR hand vein image, after which one of three different preprocessing protocols, each of which is associated with a different level of contrast enhancement, is employed for the purpose of enhancing the hand veins inside the ROI. A number of different convolutional neural networks (CNNs) are investigated for the purpose of *automatically* extracting suitable features from the NIR images, after which a suitable verifier is employed in order to achieve authentication. The proficiency of these systems are gauged within the context of a number of real world scenarios in which identity verification systems are typically employed. Experimental results clearly indicate that these systems are viable replacements to many popular authentication factors such as payment cards, handwritten signatures and access cards, *given* the feasibility of necessary adjustments to the real world application in question.

A summary of fraud statistics is provided in the following section, which further motivates the immediate need for authentication systems which are significantly more resistant to fraudulent activity.

1.1.1 Bank card fraud statistics

Bank (payment) cards, that is debit and credit cards, have become one of the most popular methods of authenticating an individual during a financial transaction, whether at an ATM or at a point of sale machine (POSM). Possible reasons include (1) their non-intrusive and seamless nature and (2) the fact that they at least provide an additional layer of security (PINs) when compared to cash transactions. This however does *not* prevent financial fraud in the slightest.

According to the 2022 consumer sentinel network data book of the Federal Trade Commission [1], a total of 2.3 million instances of fraud were reported in the US in 2022 alone, with an estimated \$8.8 billion financial loss. This is an increase of nearly \$2.6 billion over the previous year. These fraud instances are categorised by payment type and depicted in Figure 1.1.

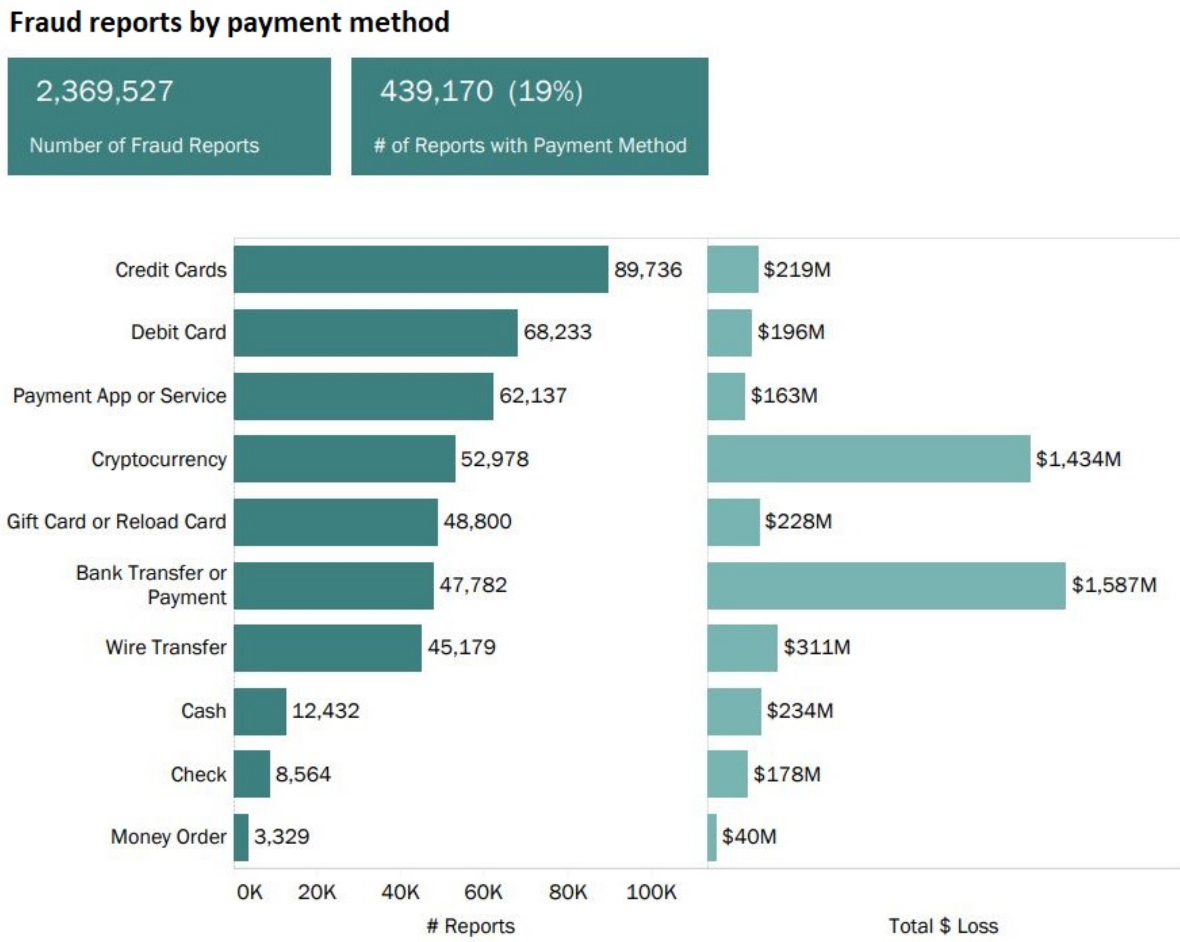


Figure 1.1: A categorisation of fraud cases by payment type in the US for 2022 (courtesy of [1]).

It is clear from Figure 1.1 that payment card fraud is a leading cause of identity theft, which is a clear indication of their susceptibility to fraud. An investigation is therefore warranted into feasible alternatives that are *significantly* more resistant to fraud, while maintaining the non-intrusive and efficient nature of payment cards.

The authentication factor proposed in this study, namely the dorsal hand vein patterns of an individual, is *not* susceptible to *any* of the aforementioned examples of fraud, since it cannot be stolen, imitated *or* counterfeited. The acquisition of the hand veins of an individual may also be facilitated in a non-intrusive and timely efficient manner, rendering said authentication factor a viable replacement for payment cards within the context of many real world scenarios.

A number of key concepts that are relevant to the systems proposed in this dissertation are discussed in the following section.

1.2 Key concepts

1.2.1 Pattern recognition

A *pattern* is also referred to as a *named entity*, and constitutes any distinguishable entity or event, for example a tree or a cloud. All living beings have the ability to recognise patterns in the environment. Bats for example have very sophisticated sonar pattern recognition, while dogs have an acute sense of smell, which is a chemical pattern. Pattern recognition is therefore a fundamental part of existence.

The ability of humans to recognise and distinguish between patterns is far more *sophisticated* than that of computers in the majority of pattern recognition tasks. Machine pattern recognition on the other hand is far more *efficient* and *convenient* when employed for the purpose of performing repetitive recognition tasks such as handwritten signature and facial recognition. Named entities are typically referred to as *classes* within the context of machine pattern recognition. The objective of such a system is therefore to automatically *classify* unknown entities or patterns as accurately as possible.

The difference between pattern *recognition* and pattern *verification* is explained in the following section, since the fundamental problem addressed in this study is one of pattern *verification*.

1.2.2 Pattern recognition and verification

The purpose of a pattern *recognition* system is to determine which class from a number of predefined classes a so-called *questioned* pattern belongs to. Such a system typically accomplishes the aforementioned task by comparing the questioned pattern to a so-called *reference* pattern associated with each class, after which the system classifies the questioned pattern to the class of which the reference pattern is most similar to the questioned pattern. A pattern recognition system is therefore also referred to as a *multi-class* classifier.

A pattern *verification* system on the other hand is tasked to determine whether or not a questioned pattern belongs to a certain class. Such a system is in other words provided with a questioned pattern *as well as* a *claim* as to which class said pattern belongs. The questioned pattern is considered *authentic* if the system accepts the claim and *inauthentic* otherwise. A pattern verification system in other words *only* has to compare the questioned pattern to the reference pattern associated with the class in question, and is therefore referred to as a *binary* classifier.

The concept of *identity* verification, which is a specific application of pattern verification, is discussed in the following section.

1.2.3 Identity verification

The classes of an identity verification system correspond to the identities of the individuals enrolled into the system. These individuals are referred to as *clients*. The patterns that represent the clients are called *authentication factors*. A categorisation of existing authentication factors is depicted in Figure 1.2. The authentication factor employed in this study, namely the hand vein structure of an individual, is depicted in boldface.

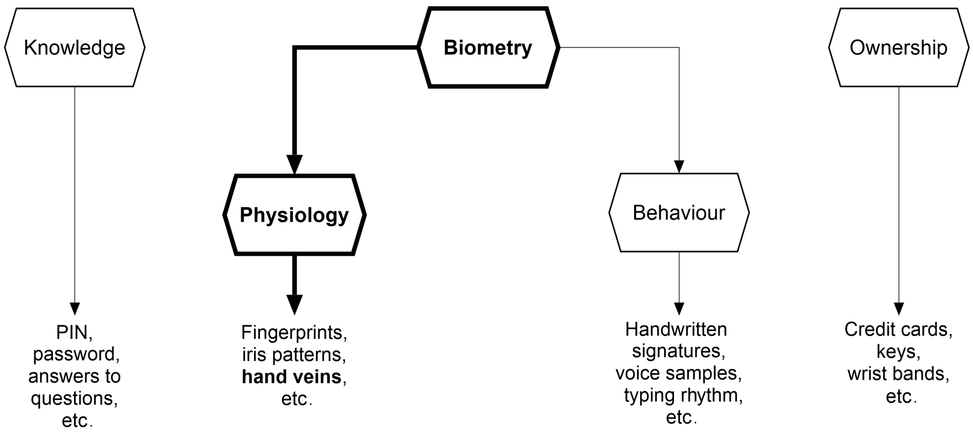


Figure 1.2: A categorisation of existing authentication factors. The authentication factor employed in this study is depicted in boldface.

The purpose of an identity verification system is to protect clients against identity fraud. The authentication factors employed in a well-designed identity verification system should therefore be sufficiently resistant to fraud, imitation and theft, while also being easily and efficiently obtainable (non-intrusive).

The majority of authentication factors however face a *trade-off* between resistance to fraud and intrusiveness. Bank cards and PINs are for example seamless ways of transacting in the fast-paced modern economy, but are extremely susceptible to fraud. An iris-based identity verification system on the other hand is extremely resistant to forgery, but cannot be employed in typical point of sale transactions at supermarkets for example, due to their intrusive and time-consuming nature.

Hand veins are one exception within the context of the aforementioned trade-off, since they are *sufficiently* resistant to forgery *and* may be acquired *seamlessly* and in a relatively *non-intrusive* manner. This renders the adoption of a hand vein-based identity verification system economically viable.

The typical architecture of an automated identity verification system is discussed in the following section.

1.2.4 Automated verification systems

An automatic verification system comprises of a number of components that are executed consecutively in order to achieve the desired output, which constitutes an authentication result. An overview of the aforementioned components is conceptualised in Figure 1.3.

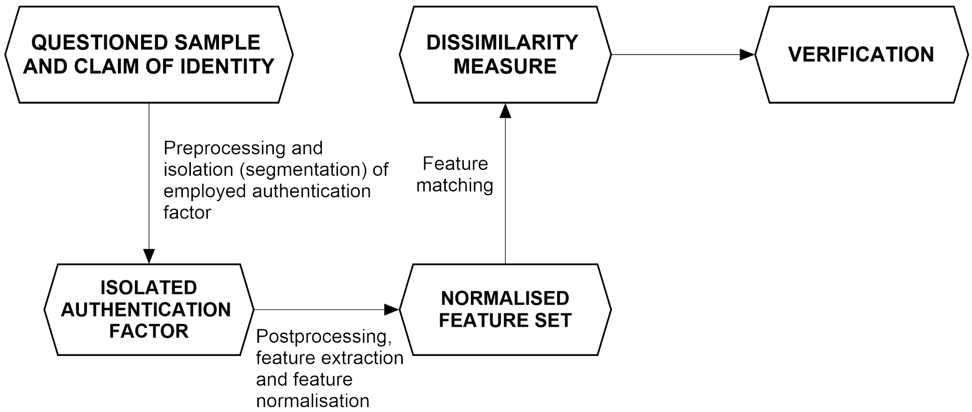


Figure 1.3: A conceptualisation of the components associated with an automated identity verification system.

It is important to note that the components conceptualised in Figure 1.3 constitute *only* the components of the automated identity verification systems developed in this study, and that *additional* components are typically required in order to implement said system in a real world scenario. Examples of these additional components include the authentication factor acquisition device, the component which sends the acquired authentication factor to the system and the component which delivers the authentication result produced by the system to the end user. The databases employed in this study are existing hand vein databases, which eliminates the need for acquiring hand vein samples, while the development of the input and output components are not included in the scope of this study. The components conceptualised in Figure 1.3 are discussed in the following sections within the context of an automated identity verification system in which the authentication factor constitutes an *image*.

1.2.4.1 Preprocessing

A certain level of preprocessing is typically required for the purpose of removing unwanted information from the acquired image, since it is infeasible in the majority of scenarios to acquire a machine-readable image of the object in question *without* any noise or artefacts. This is accomplished by employing a suitable preprocessing protocol, which typically comprises of (1) the selection of a suitable region of interest (ROI) and (2) an appropriate level of contrast enhancement during which the desired pattern is enhanced, while the background information is diminished. A categorisation of a number of popular contrast enhancement techniques is depicted in Figure 1.4.

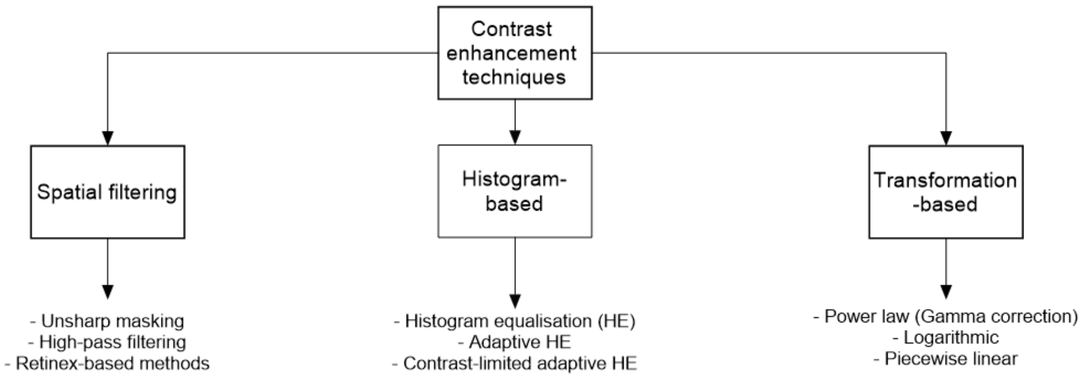


Figure 1.4: A categorisation of a number of popular contrast enhancement techniques.

It is important to note that careful consideration is necessary when selecting or developing a preprocessing protocol, since the aforementioned goal is rarely achievable in practice. There will in other words invariably be some disadvantages to the employed preprocessing protocol, for example the loss of desired information or the enhancement of irrelevant information.

Once the preprocessing component is complete, the preprocessed image is subjected to the feature extraction component, which is detailed in the following section.

1.2.4.2 Feature extraction

The feature extraction component of an automated identity verification system comprises of the application of one or more algorithms in order to obtain a set of features which describes the pattern in question. A categorisation of a number of popular feature extraction techniques is depicted in Figure 1.5.

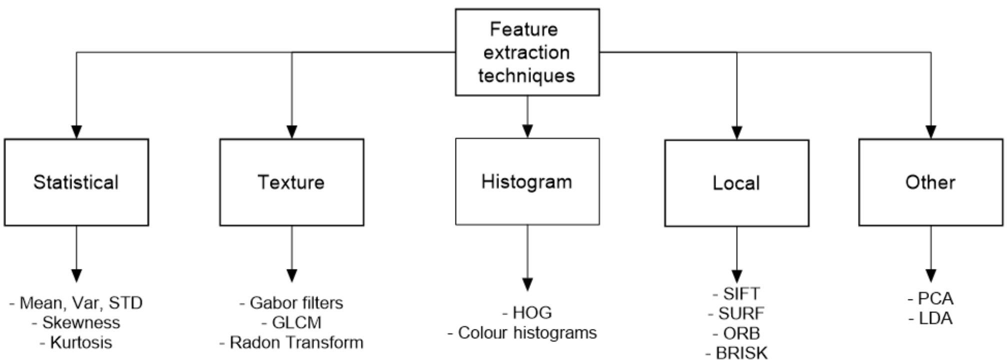


Figure 1.5: A categorisation of a number of popular feature extraction techniques.

It is important to note from Figure 1.5 that features may either be extracted manually or automatically. Manual feature extraction involves the manual selection of suitable feature extraction algorithms, while automatic feature extraction involves a certain algorithm that is able to *automatically* determine the optimal set of features for the image in question.

Recall from Section 1.2.2 that the features associated with the reference pattern must be compared to those features associated with the questioned pattern within the context of pattern *verification*. This is accomplished by the feature *matching* component, which is detailed in the following section.

1.2.4.3 Feature matching

The features associated with the reference pattern is compared with those features associated with the questioned pattern by the feature matching component, after which a so-called similarity measure is obtained. A categorisation of a number of popular feature matching algorithms is depicted in Figure 1.6.

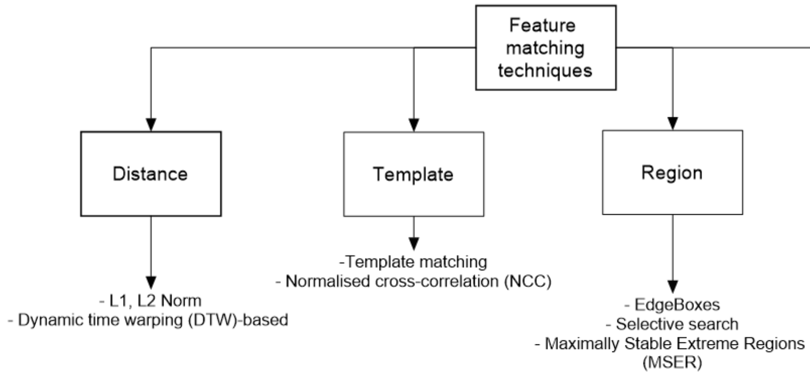


Figure 1.6: A categorisation of a number of popular feature matching algorithms.

The obtained similarity measure is used by the final component of the system, namely the verifier, for the purpose of achieving an authentication result. An overview of verification algorithms is provided in the following section.

1.2.4.4 Verification

Recall from Section 1.2.2 that a verifier is also referred to as a binary classifier. A verifier receives the similarity measure obtained from the feature matching component as input, and is tasked to decide whether or not to accept the questioned pattern. The way in which this decision is made depends on the employed classification technique, which may be (1) a predefined set of rules, (2) a statistical model or (3) a structural model. A categorisation of a number of popular classification techniques is depicted in Figure 1.7.

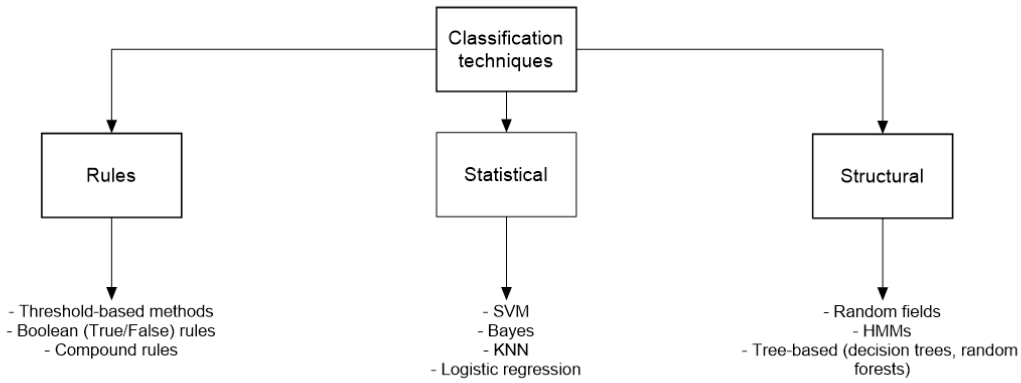


Figure 1.7: A categorisation of a number of popular classification techniques.

Statistical models typically have to undergo a so-called *training* stage, during which training *samples* are passed through the model in order to obtain predicted outputs. The predicted outputs are subsequently compared with the desired outputs, after which the so-called *cost* of the model is obtained. The purpose of the training stage is therefore to minimise the cost of the model within the context of the training samples by determining an optimal set of parameters for the model.

It is important to note that some algorithms are designed in such a way that they are able to *combine* the feature extraction component, the feature matching component *and* the verification component. One example of an algorithm that is able to combine all three aforementioned stages of an automated identity verification system is a statistical model referred to as an artificial neural network.

A number of artificial neural network architectures are investigated within the context of this study for the purpose of (1) *automatically* extracting suitable features from the preprocessed images *and* (2) *automatically* estimating the similarity between the features associated with the reference and questioned images. It is important to note that the output of the neural networks employed in this study is however not *automatically* accepted, but rather utilised by an *additional* verifier which is detailed in Section 1.4.1.4.

An overview of artificial neural networks is provided in the next section, since these algorithms constitute a fundamental component of the hand vein-based authentication systems proposed in this study.

1.2.5 Artificial neural networks

Artificial neural networks [10] are statistical algorithms of which the architecture is based on the structure and functions of biological neurons in the human brain. The fundamental building blocks of neural networks constitute nodes which are often simply referred to as *neurons*, which are organised into so-called *layers*. The connections between neurons associated with different layers determine the flow of information within the network.

It is important to note that *only* so-called *feed-forward* neural networks (FFNNs) are considered in this study in which the flow of information is unidirectional, starting from the so-called *input* layer, and ending at the so-called *output* layer. The input data is fed into the input layer and undergoes a number of *non-linear* transformations throughout the intermediary layers, which are typically known as the *hidden* layers. FFNNs may therefore be represented by a directed, acyclic graph (DAG), of which an example is depicted in Figure 1.8.

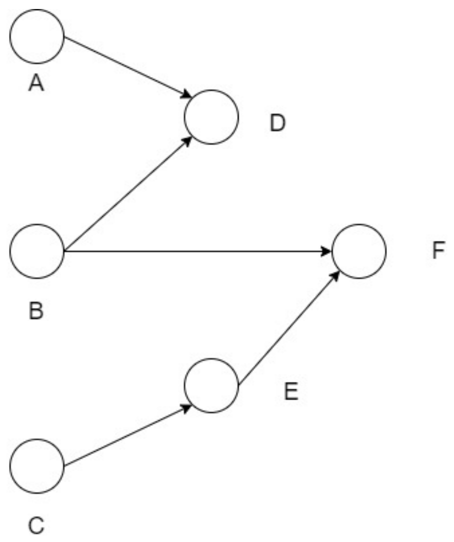


Figure 1.8: An example of a directed acyclic graph.

The non-linear transformations applied to the input data is controlled by (1) the connections between neurons, as well as (2) the so-called *weights* and (3) *biases* associated with the connections. These weights determine the "importance" of the connections, while the biases constitute constant values added to the transformation in order to avoid divergence during training. The weights and biases constitute the fundamental parameters in a neural network that need to be optimised. The optimisation process within the context of the neural networks considered in this study is called *backpropagation.* The steps of the backpropagation algorithm is summarised below for a single questioned sample.

1. A so-called *cost* vector of the network is first obtained by employing an appropriate cost function, which quantifies the difference between the output produced by the network and the desired output.

2. Once the cost vector is obtained, each element in said vector is expressed as a function of *all* the parameters (weights and biases) of the network which "produced" the cost element in question.

3. The chain rule is subsequently employed for the purpose of obtaining the direction and magnitude of the largest possible reduction in cost across all the parameters within the context of the sample in question. This enables the network to *automatically* adjust the weights and biases in such a way that the adjusted parameters produce an output which minimises the cost for the sample in question for the current *epoch* (iteration).

4. The previous three steps are repeated for *all* the training samples over a number of epochs, until the total cost of the network is minimised.

Neural networks have gained significant popularity in recent years due to their ability to *automatically* learn complex patterns and extract meaningful representations from large datasets. They have been successfully applied in a variety of real world scenarios, including computer vision, natural language processing and speech recognition.

This concludes the overview of the key concepts necessary to understand the research presented in this dissertation. The objectives of this study are provided in the following section.

1.3 Objectives of this study

The purpose of this research is to develop a number of automated hand vein-based authentication systems with the following properties:

1. **Novel architecture:** The architectures of the proposed systems should be *novel* in the sense that they (1) are sufficiently *different* when compared to existing hand vein-based authentication systems and (2) provide a significant contribution to the current literature.

2. **Implementation-ready:** The proposed systems should be *readily-implementable* in the sense that they do *not merely* serve as so-called proofs-of-concept (POCs), as is the case with many of the hand vein-based authentication systems proposed in recent literature.

3. **Seamless enrolment:** The proposed systems should be able to facilitate the enrolment of a new client in a *seamless* and *efficient* fashion.

1.4 Overview of this study

An overview of the research conducted in this study is provided in this section. It is important to note that, while this study focusses on near infra-red (NIR) hand vein images acquired from the *dorsal* surface of the hand, it is expected that *minimal* calibration is necessary in order for the proposed systems to be employed within the context of NIR hand vein images acquired from the *palmar* surface of the hand.

The proposed hand vein-based authentication systems are outlined in Section 1.4.1, while the experimental protocols and results are summarised in Section 1.4.2.

1.4.1 System design

A number of hand vein-based authentication systems are proposed in this study. The motivation for proposing multiple hand vein-based authentication systems constitutes the fact that it is virtually impossible to determine an optimal system design *prior* to experimentation, *especially* within the context of neural networks. The proficiency of *all* the proposed systems are therefore *first* evaluated, after which the top performing system is determined by ranking the proficiency of the proposed systems.

1.4.1.1 The employed databases

Subsets of two publicly available hand vein databases are employed in this study for the purpose of gauging the robustness and proficiency of the proposed systems. The applicable subset of the first database is henceforth referred to as the Bosphorus database [11], which contains 12 NIR images for each of 100 different individuals. The applicable subset of the second database is henceforth referred to as the Wilches database [12], which contains 4 NIR images for each of 100 different individuals. An example of a NIR image from the Bosphorus database is depicted on the left of Figure 1.9, while an example of a NIR image from the Wilches database is depicted on the right of Figure 1.9.

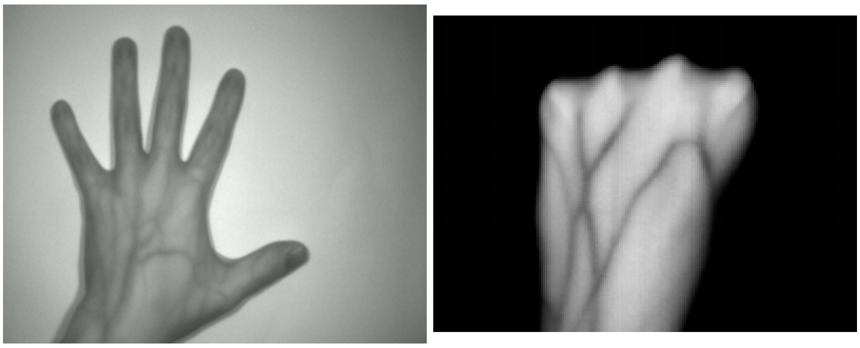


Figure 1.9: (Left) An example of a NIR image from the Bosphorus database. **(Right)** An example of a NIR image from the Wilches database.

The preprocessing protocols employed in this study are summarised in the following section.

1.4.1.2 Preprocessing

It is important to note from Figure 1.9 that the hands are open within the context of the Bosphorus database, while being closed within the context of the Wilches database. The proposed protocols for extracting a suitable region of interest (ROI) are therefore significantly different for the two aforementioned databases. A possible avenue for future work constitutes the development of a *general* ROI extraction protocol that can automatically extract a suitable ROI from multiple different databases.

The extracted ROIs are subjected to *three* different preprocessing protocols for the purpose of enhancing the desired hand vein pixels and diminishing the background pixels. These preprocessing protocols each apply different levels of contrast enhancement to the extracted ROI, and are summarised as follows:

1. **No preprocessing:** This protocol involves the utilisation of the extracted ROI *as is* without applying *any* contrast enhancement. This serves as the "benchmark" preprocessing protocol in order to establish whether system proficiency may be increased by employing higher levels of contrast enhancement.

2. **CLAHE:** The so-called Contrast-Limited Adaptive Histogram Equalisation (CLAHE) algorithm [13] is applied to the extracted ROIs within the context of the CLAHE preprocessing protocol. This popular algorithm is able to *locally* enhance the contrast of an image by utilising a so-called sliding window. This algorithm is employed in this study due to the fact that it is particularly well suited for contrast enhancement of images containing uneven illumination or dark regions [14], which is very common within the context of NIR images.

3. **Full binarisation:** The third and most extensive contrast enhancement protocol employed in this study constitutes the hand vein segmentation algorithm originally proposed by Beukes [2]. This algorithm is specifically equipped to segment *dark* and *nar-*

row structures from a grey-scale NIR hand vein image, which is shown to effectively mitigate the generation of unwanted artefacts in the segmentation result.

A summary of the specific type of neural networks employed in this study, namely *convolutional* neural networks, is provided in the following section.

1.4.1.3 Convolutional neural networks

A convolutional neural network (CNN) [15] is a type of neural network in which the non-linear transformations of the input samples are calculated with mathematical convolution, which is also known as the inverse of mathematical correlation. CNNs are most commonly employed within the context of computer vision due to their ability to recognise important regions in an image. CNNs typically comprise of three fundamental types of neural network *layers*, namely (1) convolutional layers, (2) pooling layers and (3) fully connected layers. These three layers are briefly explained below.

- **Convolutional layers:** The purpose of convolutional layers [15] is to automatically create so-called feature *embeddings* of the input images, which is achieved by iteratively performing mathematical convolution over the input image by means of a sliding window. This window is commonly known as the *receptive field*, and is based on the concept of the receptive field of the human eye.

- **Pooling layers:** The purpose of pooling layers [15] is to aggregate the output of the convolutional layers in such a way that *strong* responses are retained, while *weak* responses are ignored. Pooling layers therefore also serve as *dimensionality reduction* operators, which is necessary within the context of neural networks in order to avoid unnecessary computation.

- **Fully connected layers:** Fully connected layers [15] receive as input the aggregated N-dimensional feature embeddings produced by the convolutional and pooling layers, and output a one-dimensional feature *vector* which is finally used to represent the image in question for the purpose of classification.

Two different CNN *architectures* are employed within the context of this study for the purpose of automatically extracting suitable features from the reference and questioned images *and* to obtain a similarity measure between the two feature sets. These CNNs are henceforth referred to as two-channel (2CH) networks and Siamese networks. The aforementioned networks were originally proposed by Zagoruyko and Komodakis [16] for the purpose of obtaining a general similarity function that is able to learn image similarity *directly* from raw image data, and are the most commonly used similarity measure networks, which renders them feasible choices within the context of this study, since the objectives of this study do *not* include the development of a new similarity measure network. These networks are discussed in detail in Chapter 5.

A total of four different CNN *designs* which serve as automatic feature extractors are proposed in this study for *each* of the two aforementioned architectures . These designs comprise of different combinations of neural network *layers*, and are henceforth referred to as (1) the *standard* variation, (2) the *batch normalisation* variation, (3) the *dropout* variation and (4) the *batch normalisation and dropout* variation, and are summarised below.

- **The standard variation:** This variation constitutes a typical CNN design consisting of a specific arrangement of *only* convolutional, pooling and fully connected layers.

- **The batch normalisation variation:** This variation constitutes a specific arrangement of the standard variation with additional *batch normalisation* layers [17]. These layers normalise the output of the preceding layer in order to increase the generalisation ability of the network in order to prevent overfitting on the training data.

- **The dropout variation:** This variation constitutes a specific arrangement of the standard variation with additional *dropout* layers [18]. Dropout layers effectively *alter* the structure of the neural network by randomly dropping certain neurons and connections, and constitutes an alternative to batch normalisation layers for the purpose of increasing the generalisation ability of the network.

- **The batch normalisation and dropout variation:** This variation constitutes a specific arrangement of the standard variation with additional batch normalisation layers *and* additional dropout layers. The utilisation of *both* of these layers in a single neural network is motivated by Garbin *et al.* [19], where it is reported that batch normalisation layers may be used in conjunction with dropout layers in order to *further* increase system proficiency.

The design choice for the four aforementioned variations of the CNN component of the 2CH and Siamese networks is based on standard practice when designing a CNN. First, a "standard" CNN is considered which comprises of *only* the fundamental layers of a CNN, while the other three variations contain added regularisation layers for the purpose of determining whether or not the addition of these layers may improve the generalisation ability of the proposed systems within the context of the specific experimental protocols discussed in Section 1.4.2.1. It is important to note that, while many improved CNN designs have been proposed in recent literature such as Squeeze-and-Excitation networks [20] and CNNs that perform grouped convolution [21], the objectives of this study do *not* include the development of the most proficient CNN-based feature extractor within the context of hand vein-based authentication, but rather the evaluation of commonly used CNN designs within the context of very specific and *novel* experimental protocols that are designed to gauge the proficiency of the proposed hand vein-based authentication systems in *actual* real-world scenarios.

An overview of the verifiers employed in this study is provided in the following section.

1.4.1.4 Verification

The CNNs outlined in Section 1.4.1.3 are *all* configured to output a 2×1 *probability* vector, in which the first and second elements constitute the predicted probability that the questioned sample is *authentic* and *inauthentic* respectively.

A probabilistic threshold verifier is proposed in this study which accepts the predicted probability of the CNN if it is larger than the employed probabilistic threshold. The optimal probabilistic threshold is determined during the validation stage by gauging the proficiency of the proposed systems over a discrete set of probabilistic thresholds, and is based on exactly *one* of the following four criteria:

- The equal error rate (EER), that is the probabilistic threshold where FAR = FRR,

- the *zero* FAR (FAR_{zero}), that is the smallest probabilistic threshold where FAR = 0,

- the *zero* FRR (FRR_{zero}), that is the largest probabilistic threshold where FRR = 0 and

- the *minimum* AER (AER_{min}), that is the probabilistic threshold corresponding to the smallest AER.

The experimental protocols and results are outlined in the following section.

1.4.2 Experiments

1.4.2.1 Experimental setup

A total of four unique experiments are conducted in this study for the purpose of gauging the proficiency of the proposed systems within the context of *two* real world scenarios, namely the individual *dependent* scenario (IDS) and the individual *independent* scenario (IIS). *One* of the four experiments is associated with the IDS, while the other *three* are associated with the IIS. These experiments are henceforth referred to as (1) the individual dependent experiment (IDE), (2) the benchmark experiment (BME), (3) the first GenVeins experiment (FGE) and (4) the second GenVeins experiment (SGE).

It is important to note that there are certain challenges associated with the second and third objectives outlined in Section 1.3. The first major challenge constitutes the fact that a database consisting of a *sufficiently* large number of *different* individuals is required in order to successfully achieve the aforementioned objectives. The majority of publicly available *dorsal* hand vein databases however do *not* contain a sufficiently large number of different individuals. The IDE, FGE and SGE are therefore designed for the purpose of simulating possible solutions for this challenge. The four aforementioned experiments are briefly discussed below.

1. **The individual dependent experiment (IDE):** A tailor-made network is trained in *real time* for *each* client enrolled into the system in mere minutes. This effectively avoids the aforementioned challenge of database availability, and is shown to be extremely proficient within the context of *both* of the employed databases.

2. **The benchmark experiment (BME):** A *single* network is trained in a *once-off* fashion on a subset of *actual* individuals and evaluated on a *different* subset of *actual* individuals for *each* of the employed databases. The purpose of this experiment is to establish the severe impact on system proficiency when training on an *insufficient* number of *different* individuals.

3. **The first GenVeins experiment (FGE):** A *single* network is trained in a *once-off* fashion on a subset of *fictitious* individuals from the GenVeins database developed and proposed in this study (see Chapter 4) and evaluated on a subset of *actual* individuals for *each* of the employed databases. The purpose of this experiment is to establish the gain in system proficiency when training on a *sufficient* number of *different* individuals when compared to system proficiency within the context of the BME.

4. **The second GenVeins experiment (SGE):** This protocol is similar to the FGE, except for the fact that a questioned individual is granted a total of *three* attempts at authentication during testing. The purpose of this experiment is to establish the gain in system proficiency when *low-quality* images are discarded.

A summary of the results of the four experiments is provided in the following section.

1.4.2.2 Overview of results

The average error rates (AERs) of *only* the *top performing* systems within the context of the four experiments discussed in Section 1.4.2.1 and *both* of the employed databases are presented in Table 1.1. It is important to note that the details of the top performing systems are purposefully omitted in Table 1.1 for brevity. These details are however presented and discussed in detail in Chapter 8.

Experiment	Bosphorus (AER)	Wilches (AER)
IDE	1.708%	0.853%
BME	19.278%	5.486%
FGE	15.105%	4.271%
SGE	7.028%	1.614%

Table 1.1: The average error rate (AER) of the top performing systems within the context of the four experiments and both of the employed databases.

The following conclusions may be reached by considering the results depicted in Table 1.1:

1. The proposed systems are able to achieve *outstanding* proficiency within the context of the IDE and is robust for *both* of the employed databases. These results are also a *clear* indication of the proficiency of the proposed systems within the context of an *actual* real world scenario that utilises the proposed protocol associated with the IDS.

2. It is clear from the BME that the proficiency of the proposed systems is severely impacted by training on an *insufficient* number of *different* individuals.

3. The utilisation of the GenVeins database for the purpose of obtaining a *sufficient* number of *different* individuals for training purposes *significantly* increases the proficiency of the proposed systems when compared to the BME.

4. Granting a questioned individual a total of three opportunities for authentication *significantly* increases the proficiency of the proposed systems when compared to the BME and the FGE.

5. The detrimental impact of low-quality NIR images on system proficiency is significantly more substantial within the context of the Bosphorus database, which is clear from the large difference in system proficiency when compared to the results obtained within the context of the Wilches database for the BME, the FGE and the SGE.

It is also important to note from Table 1.1 that the reported proficiency of the proposed systems is considerably worse within the context of the IIS than is the case for the IDS. This is to be expected, since the IIS invariably constitutes a simulation of the *ideal* real world scenario in which case a *single* network is trained for *all prospective clients* in a *once-off* fashion *prior* to the enrolment of *any* client, as opposed to the IDS in which case a *tailor-made* network is trained for *each* client *during* enrolment. The advantages of the IIS over the IDS include the fact that, within the context of the IIS, a single network (1) is significantly easier to maintain, (2) is significantly more cost-efficient and (3) is able to generalise sufficiently well when presented with images from *unseen* individuals that were acquired by employing *multiple different* NIR acquisition devices. Such a single network is therefore more desirable than potentially *millions* of tailor-made networks. The IIS does therefore not require *any additional* training *or* calibration *prior* to implementation. Such a system has, to the best of the author's knowledge, *not yet* been developed within the context of *dorsal* hand vein-based authentication, which constitutes the primary motivation for gauging the proficiency of the proposed systems within the context of the IIS.

It is important to note that a statistical comparison between the proficiency of the systems proposed in this study and that of SOTA systems proposed in recent literature is *deliberately* omitted, due to the fact that the majority of SOTA systems discussed in Chapter 2 are trained *and* tested on the *same* set of individuals, which is *not* the case within the context of the systems proposed in this study. One of the recent papers discussed in Chapter 2 [22] however does train and test their proposed systems on *different* sets of *individuals*. The statistical performance measures and databases employed by Thapar *et al.* [22] and those employed in this study are however significantly different, which renders any comparison between the results reported by Thapar *et al.* [22] and the results reported in this study infeasible.

The contributions of this study are detailed in the following section.

1.5 Contributions of this study

1. **A number of novel, robust, highly proficient and implementation-ready hand vein-based authentication systems.**

   The hand vein-based authentication systems developed in this study are novel in a number of ways which are outlined below.

   - The protocol developed for the purpose of extracting a suitable ROI within the context of the Wilches database has not yet been developed to the best of the author's knowledge. Said protocol is extremely efficient in the sense that it comprises of *only* a handful of computationally *inexpensive* steps, and is shown to be robust across *all* the individuals in the Wilches database.
   - The full binarisation protocol employed in this study for the purpose of obtaining a binary hand vein image was originally proposed by Beukes [2] as an efficient and robust protocol for obtaining highly accurate segmentation results, *even* in low-quality NIR images. Said protocol has not yet been employed as a preprocessing protocol within the context of a deep learning-based authentication system. This protocol is also applied to a new database, namely the Wilches database, and is shown to achieve segmentation results of similar merit than those reported by Beukes [2].
   - The specific designs of the 2CH and Siamese networks developed in this study have, to the best of the author's knowledge, not yet been utilised for the purpose of hand vein-based authentication within the context of the *specific* experimental scenarios developed in this study. These two architectures were originally proposed by Zagoruyko and Komodakis [16], and are shown by Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23] to be highly proficient within the context of *general* image matching. The results obtained within the context of this study are consistent with the results reported by Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23], which shows that the utilisation of these

networks for the purpose of quantifying the similarity between pairs of hand vein images constitutes a feasible approach and a valid contribution to the current literature. It is however important to note that the proficiency of the proposed systems are *not* compared to existing SOTA systems within the context of this study, which constitutes a necessary avenue for future research for the purpose of gauging the extent of the contribution of the proposed systems to the current literature.

- The data augmentation protocol developed for the purpose of generating a sufficient number of samples within the context of the IDS constitutes a novel protocol specifically designed to simulate the *expected* extent of spatial variation that occurs during the acquisition of multiple hand vein samples from the same individual.

The proficiency of the majority of top performing systems proposed is this study is shown to be outstanding, *in addition* to the fact that these systems are *implementation-ready*, which is very rarely the case with the hand vein-based authentication systems proposed in recent literature. More specifically, the phrase "implementation-ready" refers to the fact that the proposed systems may be trained either in a (1) *tailor-made* fashion for *each* client enrolled into the system *in real time* or in a (2) *once-off* fashion on a set of individuals that is *sufficiently* representative of the *entire* population, which enables the proposed systems to *generalise* exceptionally well when presented with images from *unseen* individuals during implementation. The proposed systems do therefore *not merely* serve as so-called *proofs-of-concept* (POCs) in which a system is trained *and* tested on the *same* set of individuals, which is clearly not feasible within the context of *any* real world scenario. It is important to note from Table 1.1 that the proficiency of even the top performing systems within the context of the Bosphorus database, the FGE and the SGE is still arguably not sufficient to be considered "implementation-ready". Further research should therefore be conducted in order to improve the proficiency of the proposed systems within the aforementioned context.

2. **GenVeins: an artificially generated hand vein database.**

   The GenVeins database developed during the course of this study constitutes a hand vein database that comprises of a very large number of different *fictitious* individuals. The hand vein samples associated with these fictitious individuals are *automatically* generated by the protocol proposed in Chapter 4. The fact that this database is successfully utilised within the context of this study for the purpose of training the proposed systems in such a way that they are able to *generalise* exceptionally well when presented with images from *unseen* individuals during implementation constitutes a *major* contribution to the current SOTA, since a hand vein database of this magnitude has *not yet* been available prior to the existence of the GenVeins database. Said database and the corresponding source code may be made available to the public which will invariably enable a vast number of opportunities for improvement on existing hand vein-based authentication systems. he GenVeins database furthermore enables the seamless development and implementation of practical hand vein-based authentication systems by companies and organisations that need to develop and implement authentication systems that are much more resistant to fraud. The utilisation of the GenVeins database within the context of the proposed systems *additionally* contribute to the systems' "implementation-ready" characteristic in the sense that *no* training is required prior to implementation. The proposed networks are (1) trained *and* calibrated on images from *fictitious* individuals contained in the GenVeins database and (2) shown to achieve high proficiency when presented with images from *actual* individuals during implementation.

3. **The IDS: training a tailor-made network for each client.**

   The experimental protocol developed in this study in which a tailor-made network is trained for each client enrolled into the system constitutes a novel approach within the context of hand vein-based authentication systems. These networks are trained during enrolment, and are shown to be implementation-ready in a matter of minutes (see Appendix B). At the same time, these networks only require a handful of hand vein samples to be acquired from the client in question during enrolment, after which the novel data augmentation protocol developed in this study is utilised for the purpose of generating a sufficient number of training and validation samples. The hand vein-based authentication systems proposed in this study are shown to achieve *outstanding* proficiency within the context of the IDS, which further contributes to the current literature and their implementation-readiness.

4. **Seamless enrolment of new clients.**

   The two experimental scenarios proposed in this study, namely the individual *dependent* scenario (IDS) and the individual *independent* scenario (IIS) are designed in such a way that they facilitate *effortless* and *seamless* enrolment of prospective new clients into the system, which *additionally* contributes to the "implementation-ready" characteristic of the proposed systems. A *tailor-made* network is trained for *each* individual enrolled into the system *in real time* within the context of the IDS, which requires *only a handful* images from the prospective client which may be acquired *during* enrolment in a seamless and non-intrusive manner. The proposed systems can be trained in *mere minutes* within the context of the IDS (see Section B.2), which furthermore enables the seamless enrolment of prospective clients. A *single* network is trained *and* calibrated in a *once-off* fashion within the context of the IIS by utilising the proposed GenVeins database. A prospective client may therefore be enrolled in an *effortless* and *seamless* fashion in the sense that *only one* high-quality reference sample needs to be acquired *during* enrolment. The employed network does *not* need to be re-calibrated *or* updated in order for it to achieve high authentication accuracy within the context of the enrolled client, since it has *already* been trained to *generalise* exceptionally well when presented with images from *unseen actual* individuals. The seamless enrolment of new clients within the context of hand vein-based authentication systems enabled by the research conducted in this study therefore constitutes *another* major contribution to the current literature.

1.6 Outline of the dissertation

Chapter 2 (Literature study): This chapter provides a detailed review of relevant research conducted in recent years and current SOTA hand vein-based authentication systems.
Chapter 3 (Image processing): This chapter details the protocols developed for the purpose of preprocessing the NIR hand vein images associated with the databases employed in this study.
Chapter 4 (The GenVeins database): This chapter details the protocol developed in order to

generate hand vein samples of a sufficiently large number of different *fictitious* individuals that is used for the purpose of training the proposed systems.
Chapter 5 (Neural networks): This chapter provides a detailed overview of the history, mathematical foundations and relevant concepts associated with neural networks.
Chapter 6 (System design): This chapter details the hand vein-based authentication systems proposed in this study.
Chapter 7 (Experimental protocols): This chapter discusses the experimental protocols developed for the purpose of gauging the proficiency of the proposed systems within the context of a number of real world scenarios.
Chapter 8 (Experiments): This chapter presents and discusses the results of the proposed systems within the context of the two experimental scenarios.
Chapter 9 (Future work and conclusion): This chapter outlines possible avenues for future work in order to improve the proposed systems and also provides a detailed conclusion of the research conducted in this study.

1.7 Publications

During the course of this study, aspects of this research were published in a conference paper and a journal paper.

The conference paper entitled *Hand vein-based biometric authentication using two-channel similarity measure networks* was published in the *Proceedings of the Thirty-First Annual Symposium of the Pattern Recognition Association of South Africa* in 2020 [24].

The journal paper entitled *Hand vein-based biometric authentication with convolutional neural networks and support vector machines* was published in the *International Journal of Recent Research in Electrical and Electronics Engineering* in 2022 [25].

Another journal paper entitled *GenVeins: an artificially generated hand vein database* has also been accepted for publication in the *International Journal of Biometrics*.

Chapter 2

Literature study

2.1 Introduction

The most noteworthy advances in recent years within the context of vein-based biometric authentication and image matching techniques are reviewed in this chapter. The studies in which the proposed systems are based on manual feature extraction and matching techniques are discussed in Section 2.2, while recent literature involving traditional convolutional neural networks (CNNs) are presented in Section 2.3.1. Studies in which similarity measure networks (SMNs) are employed are reviewed in Section 2.3.2, while state-of-the-art (SOTA) transformer-based systems are discussed in Section 2.3.3. The differences between the systems proposed in this study and the current SOTA within the context of hand vein-based biometric authentication are finally discussed in Section 2.4.

2.2 Manual feature extraction and matching systems

As the name indicates, manual feature extraction and matching system comprises of two stages, namely feature extraction and matching. Features from images are *manually* extracted by employing a suitable algorithm, after which a similarity measure is *manually* calculated between the features associated with a reference and questioned sample. The popularity of these systems have been declining in recent years due to the advances in neural network-related research, as well as the increasing availability of computing power which is a requirement for deep neural networks. Some of these systems are however able to achieve SOTA-level proficiency within the context of biometric authentication, of which a relevant few are selected for discussion in this section.

The system proposed by Kumar and Prathyusha [26] achieves authentication by combining binarised hand vein image triangulation and knuckle shape extraction. The method is automated and utilises a near infra-red (NIR) imaging protocol that is both contactless and low-cost for the purpose of capturing palmar or dorsal hand vein images. The acquired images are first normalised, after which information about the knuckle tips are utilised for the purpose of extracting a suitable region of interest (ROI). The ROIs are subsequently binarised with an appropriate segmentation technique. The final weighted similarity measures are calculated in two stages, which involves (1) a hierarchical matching score from four topologies of triangulation within the binarised hand veins and (2) the perimeter distances between knuckle tips. An equal error rate of 1.14% is reported.

A robust dorsal hand vein authentication system is proposed by Narula *et al.* [27] that utilises a novel method based on fingertips and finger valley key points for the purpose of extracting a suitable ROI. The system employs a novel feature extraction algorithm which is based on information set theory, which includes so-called vein effective information, vein energy features, vein sigmoid features, Shannon transform features and composite transform features. A number of experiments are conducted on the GPDS and Bosphorus hand vein databases respectively. Encouraging results are reported.

A hand vein-based authentication system is proposed by Beukes [2] that is specifically adapted for scenarios in which a limited number of training samples are available. A novel protocol based on geometric properties is employed for the purpose of extracting a suitable ROI, after which the hand veins are isolated by employing a novel morphological reconstruction protocol. Feature vectors are subsequently extracted from the binarised hand veins by utilising the discrete Radon transform. The feature vectors are normalised in order to ensure rotational, translational, and scale invariance, after which a dissimilarity measure between normalised feature vectors is obtained by employing either the average Euclidean or a dynamic time warping-based distance measure. An average error rate (AER) of 2.85% is reported on the Bosphorus dataset when an individual is granted only one chance for authentication, while said AER may be reduced to 0.77% by granting an individual a total of three attempts for authentication.

A hand vein-based authentication system is proposed by Krishnaveni *et al.* [28] that is based on minutia extraction and triplet triangulation. Invariant triangles are formed from normalised images by utilising the angles and length between so-called "minutiae". The similarity between a reference and questioned hand vein sample is obtained by employing a novel triplet score. The proposed system outperforms systems which are based on independent component analysis and non-negative matrix factorisation algorithms at the time of publication, with an equal error rate (EER) of 1.26%.

Trabelsi *et al.* [29] proposed a novel multi-modal biometric authentication system in which hand and finger vein modalities are combined for enhanced recognition proficiency. The system utilises the monogenic local binary pattern algorithm for finger vein recognition and an improved Gaussian matched filter for hand vein recognition. An area under curve of 0.98 is reported.

Gourisankar and Sankareswaran [30] proposed a hand vein-based biometric authentication system in which the feature extraction stage involves the extraction of discriminative information by utilising the discrete wavelet transform together with adaptive thresholding techniques. The Hausdorff distance is subsequently employed in order to obtain a similarity measure between a reference and questioned hand vein sample. An accuracy of 99.5% is reported on the WASET database.

A computationally efficient multi-model hand vein-based biometric authentication system is proposed by Al-Roomi *et al.* [31] which employs a novel score-based fusion protocol of directional image derivatives. Said system obtains a feature representation of a given individual by utilising NIR images of *both* the palmar and wrist veins. Experimental results show that the proposed system surpasses SOTA systems (excluding deep learning) at the time of publication, with a 100% recognition rate on the PUT palm dataset. The fact that the proposed system does *not* rely on deep learning methods renders it a viable alternative in scenarios where an insufficient number of training samples is available.

A review of the recent advances within the context of image verification by employing neural networks is presented in the next section.

2.3 Neural networks

Deep learning methods has recently consistently demonstrated superior performance compared to standard machine learning models and statistical pattern recognition systems within the context of various real-world scenarios, including hand vein-based biometric authentication. A review of a number of vein-based authentication systems which employ traditional convolutional neural networks (CNNs) is presented in the next section.

2.3.1 Traditional neural networks

Two different hand vein-based authentication approaches which utilise CNNs are proposed by Abdullah and Elrefaei [32]. The first approach comprises of the utilisation of pre-trained CNN models, namely AlexNet, VGG16 and VGG19 for the purpose of automatically extracting features from input images. Authentication is subsequently achieved by utilising error-correcting output codes, together with either a support vector machine (SVM) or the k-nearest neighbour algorithm. The second approach comprises of the utilisation of transfer learning on AlexNet, VGG16 and VGG19, after which the fine-tuned models are employed for the purpose of feature extraction *and* classification. No preprocessing is applied to the input images within the context of *both* proposed approaches. A number of experiments are conducted in order to gauge the proficiency of the systems within the context of the Badawi and Bosphorus databases respectively. Experimental results for the first approach show that the pre-trained VGG19, together with ECOC and a SVM is the top performing system, with an accuracy of 100% and 98.5% within the context of the Badawi and Bosphorus databases respectively. Experimental results for the second approach indicate that VGG16 achieves the highest combined accuracy over both of the employed databases, with an accuracy of 100% and 99.0% within the context of the Badawi and Bosphorus databases respectively. The study finally concludes that the second approach significantly outperforms the first approach, indicating the benefit of utilising transfer learning within the context of hand vein-based biometric authentication.

Another system that is based on transfer learning with AlexNet is proposed by Laghari *et al.* [33], which involves a systematic fine-tuning approach that is based on the notion of a grid search in order to optimise the pre-trained AlexNet on the Bosphorus hand vein database. The proposed system is evaluated on the original images, as well as on cropped versions of the original images. An accuracy of 96% is reported when the model is trained on the cropped dataset, and is shown to significantly outperform the model when trained on the original images. The authors furthermore concluded that the proposed system outperforms the majority of SOTA systems at the time of publication, and that further improvements to the proposed system can be made by *automatically* searching for parameters, as opposed to manually *specifying* a parameter grid.

A novel approach for contactless palm vein-based biometric authentication, also based on transfer learning, is proposed by Chantaf *et al.* [34]. The proposed systems comprise of two pre-trained CNNs, namely Inception V3 and SmallerVggNet for the purpose of automatically extracting features from NIR palm vein images. These pre-trained CNNs are fine-tuned and tested on a new palm vein database acquired during the course of the study, which comprises of 20 different individuals and 200 images per individual. Accuracies of 93.2% and 91.4% are reported for experiments conducted on SmallerVGGNet and Inception V3 respectively.

Obayya *et al.* [35] proposed a novel CNN-based palm vein authentication system which incorporates Bayesian optimisation. The system first extracts a suitable ROI, after which the vein patterns are enhanced using a Jerman filter. The training protocol constitutes an iterative process during which Bayesian optimisation is utilised in order to determine the optimal set of hyper-parameters, as well as the optimal *structure* of the CNN in question. More specifically, a base CNN is defined which consists of three identical sequences of layers. The optimisation method starts by obtaining the validation error for the CNN in which only the first sequence of layers is present, after which the process is repeated for the CNN in question after adding the second sequence of layers. This process is either repeated three times, or until the proficiency of the augmented network starts declining. An accuracy of 99.4% and EER of 0.0683% are finally reported for the optimised CNN within the context of the CASIA Multi-Spectral Palmprint Image Database.

An iterative Deep Belief Network (DBN) is proposed by Qin *et al.* [36] for the purpose of automatically extracting palm vein features. A suitable ROI is first extracted, after which a novel vein segmentation technique is utilised for the purpose of hand vein segmentation. These binarised hand vein images are labelled on a pixel level in such a way that a vein pixel corresponds to the label 1, while a background pixel corresponds to the label 0. The proposed DBN is subsequently trained in an iterative fashion to predict whether each pixel is a vein pixel or not, while the predicted labels are corrected after each iteration. A similarity measure is obtained by calculating the Hamming distance between a reference sample (in which the vein pixels are labelled) and a questioned sample containing unlabelled pixels. EERs of 0.33% and 0.015% are reported for experiments conducted on the CASIA multi-spectral palmprint database and the PolyU multi-spectral palmprint database respectively, which are shown to outperform the SOTA at the time of publication.

A deep residual network with an attention mechanism called DRNAM is proposed by Shu *et al.* [37] for the purpose of addressing (1) large intraclass variation between dorsal hand vein samples acquired through NIR imaging and (2) the problem of the availability of a large number of training samples. An improved NIR image acquisition protocol based on the so-called "transmission type" of the NIR spectrum is proposed for the purpose of acquiring dorsal hand vein images with reduced intraclass variance. DRNAM is subsequently utilised for the purpose of automatically extracting compact and discriminative features from dorsal hand vein images, which is shown to improve the robustness of feature embeddings, since it incorporates *both* cross-channel and spatial information. The experimental results demonstrate that the dorsal hand vein images acquired by means of the transmission-type NIR spectrum is of higher quality than those acquired by means of the so-called "reflection-type" NIR imaging protocol, which ultimately leads to a lower EER of 0.5% compared to an EER of 1.1%.

A review of a number of image matching systems which employ similarity measure networks (SMNs) is presented in the next section.

2.3.2 Similarity measure networks

Recall from Chapter 1 that the problem addressed in this study is that of image *verification*, rather than classification, which involves a reference sample and a questioned sample. The employed system is presented with a questioned sample, together with a claim as to which class the questioned sample belongs, after which the system is prompted to decide whether the claim is true or not.

Similarity measure networks (SMNs) are neural networks that are trained in such a way so as to obtain a general similarity function that is able to perform verification. The majority of

SMNs rely on convolutional neural networks (CNNs) for the purpose of obtaining so-called feature embeddings, after which a suitable verifier is employed in order to either accept or reject a questioned sample.

Zagoruyko and Komodakis [16] proposed a method for learning a general similarity function directly from image data *without* the need for manually selected features. A number of different neural network architectures are investigated based on two-channel networks and Siamese networks. Experiments show that the proposed method outperforms existing methods within the context of various computer vision problems and benchmark datasets at the time of publication. The results of this study indicate that the proficiency of so-called two-channel networks are generally superior to the Siamese networks. Results also indicate that the two-stream multi-resolution Siamese networks and spatial pyramid pooling-based Siamese networks achieve superior performance to standard Siamese networks, which emphasises the importance of multi-resolution information within the context of the comparison of images.

Zagoruyko and Komodakis [16] conducted a follow-up study Zagoruyko and Komodakis [23] in which so-called normalised cross-correlation (NCC) layers are proposed. A NCC layer is a novel neural network layer which is similar to convolutional layers, but the input to the layer *and* the corresponding set of weights are normalised prior to the forward pass through the layer. SMNs in which NCC layers are utilised are shown to achieve superior proficiency when compared to the same SMNs in which standard convolutional layers are utilised. The top performing SMN, that is a two-channel NCC network, achieves an average false positive rate at 95% true positive rate (FPR95) of 3.84% over a number of *different* grey-scale image databases.

The general image matching systems based on deep learning methods proposed by Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23] are, to the best of the author's knowledge, the first of their kind in the sense that they are able to effectively quantify the similarity between two images containing objects that were *not* seen during training. The dataset used is referred to as the "PhotoTourism" dataset, and comprises of pairs of image patches that were sampled from 3D reconstructions of the Statue of Liberty in New York, Notre Dame in Paris and Half Dome in Yosemite. The experimental protocol proposed by Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23] invariably comprises of training and testing with image pairs which contain *different* objects. For example, a system may be trained on image patches from the Statue of Liberty, while tested on images patches from Notre Dame. The mean of the false acceptance rate at 95% true positive rate (FPR95) is reported for *all* six possible combinations of training and testing sets. This is closely related to the experimental protocol proposed in this study referred to as the "individual independent scenario (IIS)" (see Section 7.1), in which the proposed systems are trained and tested with image pairs associated with *mutually exclusive* sets of *individuals*.

The purpose of the system proposed by Bell and Bala [38] constitutes learning an embedding for visual search within the context of interior design that contains two domains of product images, namely internet scene crops and iconic product representations. The authors demonstrate the effectiveness of the proposed multi-domain embedding through various applications such as product identification in scenes and identifying stylistically similar products. The proposed multi-domain embedding is evaluated within the context of several different training architectures such as object classifiers and Siamese networks. The results indicate that the proposed systems are able to achieve high-quality search results across multiple visual domains.

A deep multi-scale Siamese network called SimNet is proposed by Appalaraju and Chaoji

[39], which is trained on pairs of positive and negative images by utilising a novel online pair mining strategy (OPMS) based on curriculum learning. SimNet incorporates both top and lower layer embeddings of images, where each embedding is associated with a different level of invariance. The proposed multi-scale SMN is shown to be more effective in the task of capturing fine-grained image similarities when compared to traditional CNNs. An accuracy of 92.6% is reported.

A general similarity function that is trained directly on raw image pairs is proposed by Zhang [40]. Different deep neural networks are evaluated, including double-channel and Siamese networks. This study is similar to the studies conducted by Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23], while the difference constitutes the fact that the employed databases comprise of colour images as opposed to grey-scale images. Experimental results indicate that the proposed systems significantly outperform the current SOTA at the time of publication.

Fang *et al.* [41] proposed a lightweight deep-learning system for finger vein verification. The system comprises of a two-channel two-stream network in which the two-channel architecture contains only three convolutional layers. The original image is passed through the first stream of the network, while a mini-ROI of the original image is passed through the second stream. The purpose of the mini-ROI is to mitigate the negative effect of spatial variation between samples of the same individual. EERs of 0.10% and 0.47% are reported within the context of the MMCBNU and SDUMLA databases respectively, which outperform existing methods at the time of publication.

An end-to-end pipeline for hand vein-based biometric recognition is proposed by Bagchi *et al.* [42]. The pipeline comprises of image enhancement, ROI extraction, and deep learning models for the purpose of recognition. A total of three deep learning models are proposed for the purpose of automatically extracting features from the ROIs, namely a customised CNN, a Siamese network, and a Triplet network. The Siamese network is trained by employing the contrastive loss function, while the Triplet network comprises of a pre-trained ResNet50 that is fine-tuned by employing the triplet loss function. The results indicate that the simple CNN model outpeforms the Siamese and Triplet networks, which may be due to the the limited availability of the training databases.

A palm vein authentication system called PVSNet is proposed by Thapar *et al.* [22], which involves two main steps. The first step comprises of an encoder-decoder network that learns domain-specific features from a suitable grey-scale ROI that is extracted from a palm vein image. The second step involves the utilisation of an auto-encoded Siamese network with pre-trained convolutional layers in conjunction with a triplet loss function in order to optimise feature embeddings. Thapar *et al.* [22] developed a novel experimental protocol which involves gauging the proficiency of the proposed systems on images from the same database, as well as on images from a *different* database that was employed for training. The second protocol is similar to the so-called "individual independent scenario" (IIS) developed in this study (see Section 7.1), which involves training and testing the proposed systems on *mutually exclusive* sets of individuals. It is clear from the results reported by Thapar *et al.* [22] that the proficiency of the proposed systems are *severely* impacted when testing on images from *unseen individuals*. The fact that the experimental protocol proposed by Thapar *et al.* [22] gauges the proficiency of their system by authenticating *unseen individuals* during testing provides a good indication of the proficiency of their system within the context of an actual real-world scenario, and may therefore be considered an implementation-ready multispectral palm-print authentication system. This constitutes a significant contribution to the current literature at the time of publication, and the author of this dissertation is of the opinion that the study by Thapar *et al.* [22] ranks among the top SOTA systems within the context of multi-spectral palm-print authentication systems based on deep learning. It is important to note that Thapar *et al.* [22] however employed *multi-spectral palm-print* databases, which is different to the *near infra-red dorsal* hand vein databases employed in this study.

A novel k-shot palm vein identification system is proposed by Swain *et al.* [43] which utilises Siamese neural networks. The purpose of this system is to propose a workaround in scenarios where a limited number of training samples are available. The proposed Siamese network employs images from the left *and* right palms of an individual, where each palm is fed into one of the two branches of the Siamese network. A combined feature embedding for the individual in question is subsequently obtained though the output of the Siamese network, which is used to authenticate the individual in question against a reference feature embedding. The proficiency of the proposed system is gauged within the context of the PolyU multi-spectral palm vein database. The systems in which $k = 5$ is reported to be the most proficient, and achieves a precision of 91.9%, a recall of 91.1%, a specificity of 92.2%, a F1-score of 91.5% and an accuracy of 90.5%.

A review of a number of transformer-based authentication systems is presented in the next section.

2.3.3 Transformers

The utilisation of transformers has drastically increased in recent years due to several advantages when compared to traditional CNNs. One of the main advantages of transformers constitutes the fact that they excel in capturing long-range dependencies in sequential data by employing self-attention mechanisms. Secondly, transformers eliminate the need for sequential processing, enabling parallelism and faster training, which in turn enables increased scalability and efficiency within the context of large-scale datasets.

Gao *et al.* [44] address certain limitations of traditional CNNs by proposing a label enhancement-based multi-scale vein transformer called LE-MSVT. These limitations constitute (1) the inability to embed long-range pixel dependencies and (2) capturing label correlation. LE-MSVT utilises a multi-scale vein transformer for the purpose of extracting robust features, together with a novel label enhancement approach that is based on a graph convolutional network for the purpose of capturing label correlations. LE-MSVT is shown to significantly outperform the current SOTA at the time of publication within the context of *three* hand vein databases, namely the PolyU multi-spectral palm print database, the Tongji University palmprint database and the VERA palm vein database. Accuracies of 99.70%, 98.73% and 95.45% are reported within the context of the aforementioned databases respectively. Gao *et al.* [44] acknowledges that their system requires *all* classes (individuals) to be known during training in order to authenticate during testing, and that their system would have to be retrained upon the enrolment of new clients. It was stated that, for future work, a so-called "class-incremental" approach for hand vein recognition should be investigated.

A study by El-Yacoubi *et al.* [45] proposes a full-view finger vein identification system called FV-LT that utilises a local attention transformer. A prototype device that is capable of capturing full-view finger vein images is designed, which rotates a LED group along the finger during image acquisition. The prototype device is shown to capture concealed vein patterns that are not typically captured with standard finger vein acquisition protocols, which produces a more comprehensive representation of finger vein information. The proposed local attention transformer is utilised for the purpose of automatically extracting so-called "dependency features" among image patches *and* full-view images, which is shown to effectively normalise positional variations caused by finger rolls during the acquisition of multiple samples of the same individual. The proficiency of FV-LT is gauged by employing two databases, namely a full-view finger-vein image database acquired by employing the proposed prototype device, as well as a single-view database. The experimental results invariably show that FV-LT outperforms the current SOTA at the time of publication by employing the full-view acquisition protocol.

Garcia-Martin and Sanchez-Reillo [46] proposed a novel approach to vascular-based biometric authentication that involves multiple pre-trained and fine-tuned vision transformers. The objective of the study is to address the challenge of the availability of a limited number of samples in publicly accessible vascular datasets. The proposed protocol involves pre-training the vision transformers on distinctive image features from the ImageNet-1k and ImageNet-21k databases. These pre-trained models are subsequently fine-tuned for the four primary vascular variants, namely finger veins, palm veins, dorsal veins and wrist veins. The proficiency of the proposed systems is gauged on a number of existing vascular datasets, as well as on a novel wrist vein database called UC3M-CV3. Experimental results indicate the efficiency and robustness of the proposed systems across multiple different vascular databases, and is shown to be comparable to the current SOTA at the time of publication.

Another study by Li *et al.* [47] introduces a novel model called ViT-Cap, which comprises of a combination of two different neural network architectures, namely vision transformers and capsule networks, for the purpose of finger vein recognition. The model utilises both global and local attention for the purpose of automatic feature extraction. The proposed approach comprises of first splitting the input images into patches, after which each patch is passed through the vision transformer in order to obtain a feature embedding. These features are subsequently processed by the capsule network in order to achieve a recognition result. The proficiency of the proposed system is gauged on four publicly available finger vein databases, and is shown to achieve an average recognition accuracy which exceeds 96%. The proposed ensemble-based system is shown to outperform each separate component of the system, as well as other advanced finger vein recognition systems. It is reported that the proposed system achieves an 0.3% EER within the context of the FV-USM dataset, which is comparable to the current SOTA at the time of publication. Within the context of identity verification, the author of this dissertation is of the opinion that the system proposed by Li *et al.* [47] called ViT-Cap, constitutes one of the most advanced authentication systems proposed in recent literature. ViT-Cap combines two of the most sophisticated neural network architectures within the context of image processing, which should be further investigated within the context of alternative authentication factors such as hand veins.

The fundamental differences between the current SOTA and the systems proposed in this study are highlighted in the following section.

2.4 The systems proposed in this study

The main difference between the current SOTA and the systems proposed in this study constitutes the fact that the SOTA dorsal hand vein-based authentication systems discussed in this chapter are *not* readily implementable within the context of a real-world scenario. The reason for this is the fact that the SOTA dorsal hand vein-based authentication systems discussed in this chapter are trained *and* tested on the available images that belong to *all* individuals in the employed databases, which invariably renders the systems biased towards the samples of the individuals utilised for training. In other words, the reported results of the SOTA dorsal hand vein-based authentication systems discussed in this chapter are *not* indicative of the proficiency of the proposed systems when presented with images from *unseen individuals*.

The novel systems proposed in this study on the other hand are shown through rigorous experimentation (see Chapter 7) to be efficient and readily implementable within the context of many real world scenarios. This is accomplished by either (1) training a tailor-made model for *each* individual enrolled into the system in mere minutes or (2) training a single model in a *once-off* fashion on a *sufficient* number of *different individuals*, which renders the system *unbiased* towards the individuals utilised during training. This constitutes a major contribution of this study to the current SOTA.

2.5 Concluding remarks

A comprehensive review of recent studies conducted within the context of vein-based biometric authentication was presented in this chapter. The review included manual feature extraction and matching techniques, traditional CNNs, SMNs and transformer-based systems. The differences between the current SOTA and the systems proposed in this study were also highlighted. The various preprocessing protocols developed during the course of this study is presented in the next chapter.

Chapter 3

Image processing

3.1 Introduction

The image processing techniques developed in this study for the purpose of obtaining input images for the employed networks are detailed in this chapter. Recall from Chapter 1 that a total of three databases are employed in this study, namely the Bosphorus database [11], the Wilches database [12] and the GenVeins database. The first two are actual hand vein databases acquired from real individuals, while the GenVeins database is an artificially generated hand vein database developed during the course of this study (see Chapter 4).

Within the context of the two actual hand vein databases, it is necessary to *first* extract a region of interest (ROI), *prior* to the application of the image processing techniques proposed in this study. Within the context of the GenVeins database however, the generated images already constitute *only* the required ROIs that will serve as the input for the employed networks.

This chapter therefore *only* contains the details pertaining to (1) the extraction of the ROIs within the context of the actual hand vein databases and (2) the employed image processing techniques within the context of the Bosphorus database, since these techniques are identical for the ROIs associated with all three employed databases.

3.2 Extracting the region of interest

The protocols developed for extracting a suitable region of interest (ROI) within the context of the two actual hand vein databases are detailed in this section. These protocols differ substantially between the Bosphorus and Wilches databases due to a significant difference in the respective image acquisition techniques.

3.2.1 Extracting the region of interest: The Bosphorus database

It is important to note that the ROI extraction protocol for the Bosphorus database has been developed and extensively detailed in a previous study [2] and is therefore relayed in this section for the purpose of convenience, with references where necessary. A minor change has however been made to the final component of the aforementioned protocol within the context of *this* study in order to obtain a ROI of shape 50×40 pixels. These are the expected input dimensions for the feature extractors employed in this study (see Chapter 6). The aforementioned change is detailed in Section 3.2.1.5.

Two typical near infra-red (NIR) images from the Bosphorus database of size 300 × 240 pixels belonging to different individuals are depicted in Figure 3.1, in which approximate boundaries of the desired ROIs are *manually* annotated in yellow.

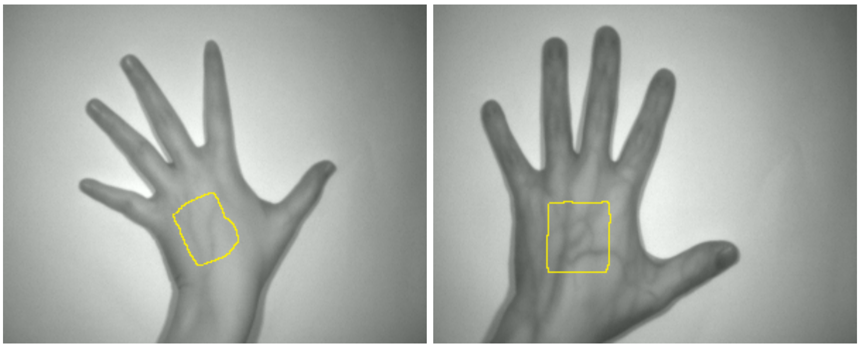


Figure 3.1: Two images from the Bosphorus database belonging to two different individuals. Approximate boundaries of the desired ROIs are *manually* annotated in yellow.

The ROI extraction protocol within the context of the Bosphorus database developed by Beukes [2] and implemented in this study may be summarised as follows:

1. The detection of the hand (see Section 3.2.1.1).

2. The pruning of the four fingers (see Section 3.2.1.2).

3. The vertical alignment of the hand (see Section 3.2.1.3).

4. The pruning of the thumb and wrist (see Section 3.2.1.5).

It is important to note that the extraction of the ROI is accomplished by systematically eliminating pixels in the input image which are deemed *not* to belong to the set of pixels associated with the ROI.

3.2.1.1 Detecting the hand

The first step towards the extraction of a suitable ROI involves the detection of the hand. More specifically, the phrase "detection of the hand" refers to obtaining the set of pixels in the input image associated with the hand. In order to obtain the aforementioned set of pixels, a suitable *thresholding* strategy is employed.

Thresholding involves the binarisation of a greyscale image, where all greyscale values (intensities) in the image *smaller* than the employed threshold intensity are rendered black (assigned a value of 0), while all greyscale values in the image *larger* than or equal to the employed threshold intensity are rendered white (assigned a value of 255). The thresholded image is subsequently normalised so that all black pixels are 0-valued and all white pixels are 1-valued. In the remainder of this dissertation, a set of connected white pixels within a binary image will be referred to as an *object*, while all black pixels, regardless of whether they are connected or not, will collectively be referred to as the *background*.

Otsu's thresholding method [48] is employed in this study for the purpose of obtaining a binarised image, and constitutes the first step towards the detection of the hand. Otsu's method evaluates every greyscale intensity $t \in [0, 255]$ as a possible threshold intensity, after which it selects the one that maximizes the *inter-class variance*. This ensures maximal separation between the objects and the background. The principles involved in Otsu's method are conceptualised in Figure 3.2, where the optimal threshold is located roughly in the middle of the two peaks of a bimodal histogram.

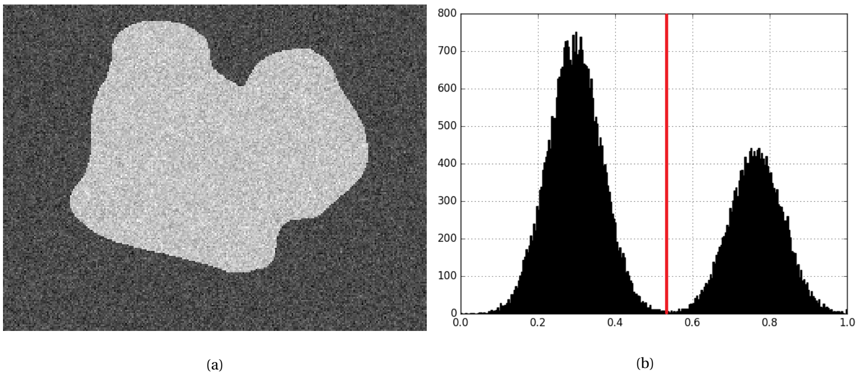


Figure 3.2: (a) A grey-scale image corrupted by Gaussian noise. **(b)** The histogram of the image in (a), where the location of the appropriate threshold value as determined by Otsu's algorithm is denoted by the red vertical line. These images are adopted from Beukes [2].

The results of applying Otsu's method to the images depicted in Figure 3.1 are shown in Figure 3.3. Due to the contrast between the hand and its immediate boundary, the application of Otsu's method to the original images will at least result in a well-defined hand boundary. Note that a Gaussian smoothing filter is applied to *all* grey-scale images prior to the application of a thresholding operation within the context of this study, which is a standard approach to soften object boundaries.

Figure 3.3: The results of applying Otsu's method to the images depicted in Figure 3.1.

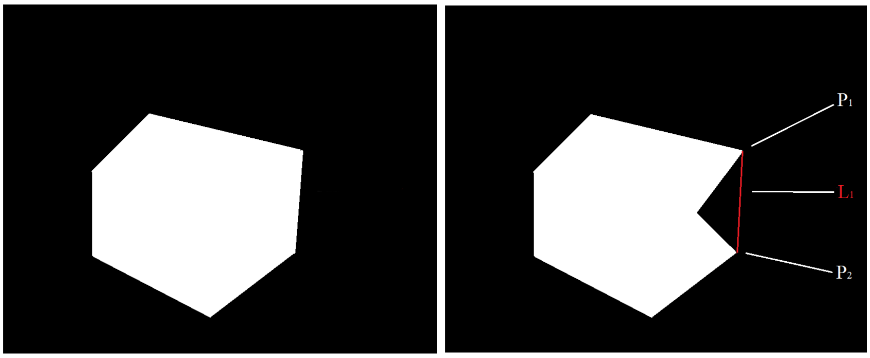
It is important to note that, since the hand is *darker* than its immediate surroundings, the application of Otsu's method produces a black, hand-shaped region instead of a white, hand-shaped object. The corner regions of the images are also black, while the immediate area surrounding the black hand-shaped region is white. This is a result of the NIR imaging equipment used during acquisition within the context of the Bosphorus database. The following steps are therefore necessary in order to obtain an image containing *only one* set of white pixels associated with the hand:

1. The union of the actual background of the hand, that is the union of the circular white object surrounding the black hand-shaped region and the four black corner regions, is first obtained.

2. The arbitrary white objects located near the center of the black hand-shaped region are subsequently removed.

3. The complement of the resulting image is finally obtained in order to produce a hand-shaped object on a black background.

In order to unify the actual background of the hand, the pixels associated with the large black corner regions are first identified by employing the morphological concept of a *convex hull*. A convex object is defined as an object for which *no* two pixels can be connected with a line that does *not also* form part of said object. A convex and a non-convex object is depicted in Figure 3.4 (a) and (b) respectively.



(a) (b)

Figure 3.4: (a) A convex object. **(b)** A non-convex object. These images are adopted from Beukes [2].

The convex *hull* of an object is subsequently defined as the *smallest* solid convex object that contains the object in question. For example, the convex hull of the object depicted in Figure 3.4 (b) may be obtained by adding the line L_1 to the boundary of said object and rendering the black area inside the new object white. The convex hulls of the objects depicted in Figure 3.3 are depicted in Figure 3.5.

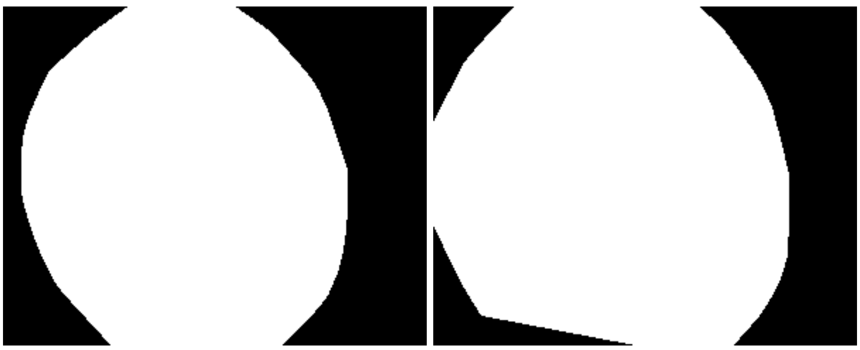


Figure 3.5: The convex hulls of the images depicted in Figure 3.3.

The results when pixels associated with the large black corner regions are negated and merged with the large white object surrounding the black, hand-shaped region depicted in Figure 3.3 are depicted in Figure 3.6.

All the morphologically connected components in the images depicted in Figure 3.6 are subsequently identified for the purpose of unifying the white artefacts within the black,

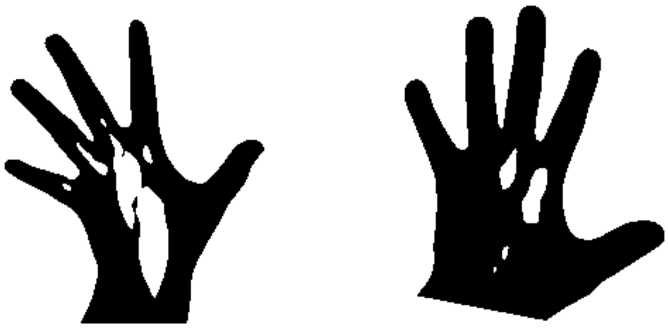


Figure 3.6: The images depicted in Figure 3.3 in which the actual background of the black, hand-shaped region is unified.

hand-shaped region, after which *only* the largest *two* are retained. These two retained components are associated with the black, hand-shaped region and the surrounding white object. The results are finally negated and depicted in Figure 3.7.

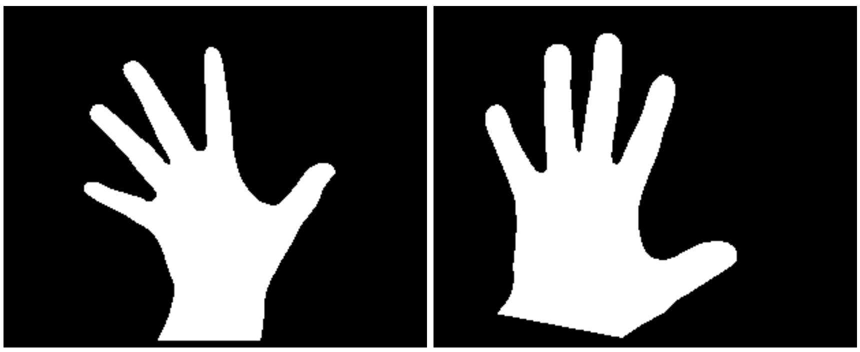


Figure 3.7: The negative of the images depicted in Figure 3.6 in which the white artefacts within the remaining black, hand-shaped region are removed.

3.2.1.2 Pruning the four fingers

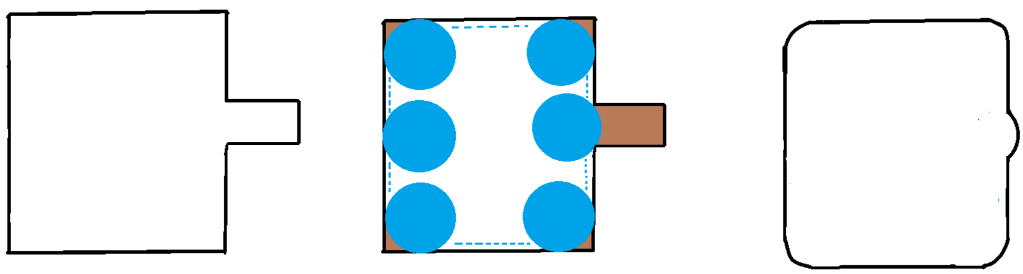
Image thresholding is a feasible approach for the purpose of detecting the hand, due to the fact that the hand and its immediate surroundings are well-contrasted. This is *not* the case within the context of the four fingers and the rest of the hand. *Morphological* image processing techniques are therefore employed, which consider the morphological shape and size of the desired regions, rather than their grey-scale intensities.

Morphological image processing techniques are based on set theory, in conjunction with a so-called *structuring element* (SE). An SE fundamentally constitutes a set of pixels that is shaped and sized in such a way that it is appropriate for the morphological operation and the region in question. In other words, the morphology of the employed SE, together with the employed morphological operation, determine the retention and/or removal of certain regions in a grey-scale or binary image. This is accomplished by iteratively centering the SE on every *white* pixel within the image, and subsequently applying the morphological operation. Note that only *binary* morphological processing is employed within the context of this chapter.

When comparing the morphology of the four fingers with the rest of the hand, it is clear that the four fingers are significantly narrower. The morphological operation known as *open-*

ing is therefore employed, which facilitates the removal of regions in which the employed SE cannot fit into in its entirety. The aforementioned operation comprises of two stages. During the first stage, the intersection between the SE and every region in the input image is iteratively calculated, which is called morphological *erosion*. During the second stage, the union between the SE and the regions in the output of the first stage is iteratively calculated, which is called morphological *dilation*. In this way, the morphology of regions that had not been entirely removed during the first stage remain unchanged after the operation.

Morphological opening is conceptualised by means of a simple example in Figure 3.8, where the employed SE is disc-shaped and depicted in blue. The brown regions in Figure 3.8 (b) may therefore be removed (see Figure 3.8 (c)) by applying morphological opening with the aforementioned SE to the object depicted in Figure 3.8 (b).



(a) (b) (c)

Figure 3.8: (a) The boundary of an object. **(b)** Conceptualisation of the morphological opening of the object in (a) with the blue disc-shaped SE. **(c)** The result of opening (a) with the disc-shaped SE depicted in (b). These illustrations are adopted from Beukes [2].

In order to successfully prune the four fingers, morphological opening with a disc-shaped SE of which the diameter slightly exceeds the estimated width of a finger is applied to the objects depicted in Figure 3.7. Note that the estimated width of a finger has been experimentally determined within the context of the Bosphorus database, and more specifically denotes the estimated width of the largest finger in the entire dataset. This is a feasible approach, since (1) opening does not affect the morphology of any regions *unless* a region is entirely removed and (2) the fingers are *significantly* narrower than the rest of the hand, which mitigates the risk of erroneously removing desired parts of the hand. The results obtained when applying the aforementioned operation to the images depicted in Figure 3.7 are shown in Figure 3.9.

3.2.1.3 The vertical alignment of the hand

When the images depicted in Figure 3.1 are compared, it is clear that the orientation of the two hands differ substantially. This is the case for all the samples in the Bosphorus database. The stages of the ROI extraction protocol up to this point are not affected by the orientation of the hand, since the SEs in the aforementioned stages are *isotropic*. The SEs in the following stages are however *not* isotropic, and are in fact affected by the orientation of the hand. It is therefore necessary to *first* ensure that all the hands are vertically aligned *before* commencing with the following stages pertaining to the acquisition of the desired ROIs within the context of the Bosphorus database.

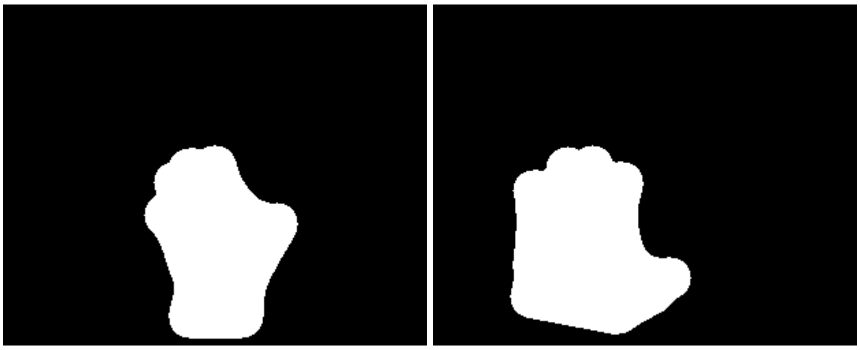
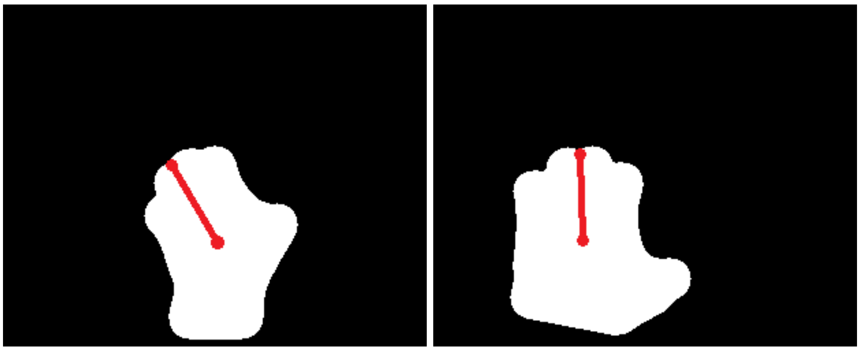
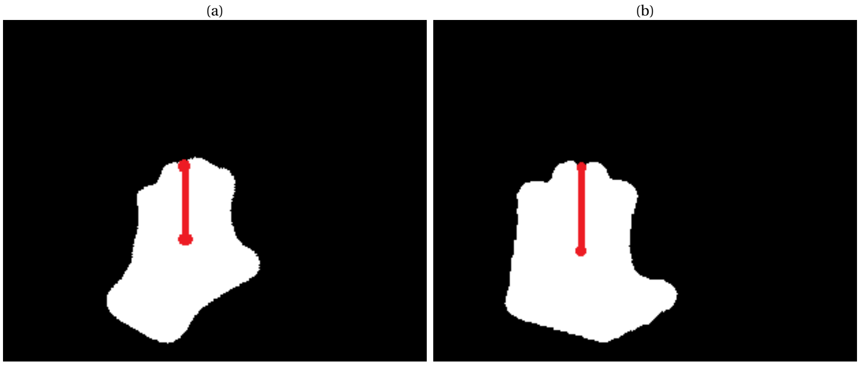


Figure 3.9: The images in Figure 3.7 in which the four fingers are removed through morphological opening with an appropriately sized and shaped SE.

The orientation of the hand is defined by the angle of the line connecting the center of mass and the webbing between the middle and ring fingers of the objects depicted in Figure 3.9. This line is manually annotated in red on the aforementioned objects and shown in Figure 3.10.



(a) (b)



(c) (d)

Figure 3.10: (a) and (b) The images depicted in Figure 3.9, where the line that is used to approximate the orientation of the body of the hand is manually annotated in red. **(c) and (d)** The vertically oriented versions of (a) and (b) respectively.

The aforementioned re-orientation procedure is subsequently applied to *all* images in the Bosphorus database during this stage of the image preprocessing phase. The protocol developed by Beukes [2] for identifying the angle of the red lines depicted in Figures 3.10 (a) and (b) is detailed in the following section.

3.2.1.4 Estimating the orientation of the hand

In order to estimate the angle of the red lines depicted in Figures 3.10 (a) and (b), the center of mass of these objects and the location of the webbing between the middle and ring fingers have to be identified. The center of mass is obtained by calculating the morphological center of mass of the objects depicted in Figures 3.10 (a) and (b). The protocol for identifying the location (coordinates) of the webbing between the middle and ring fingers is as follows. Note that for brevity, the aforementioned protocol is only demonstrated for the image depicted in Figure 3.10 (a).

First, the boundary of the binary hand is obtained through morphological boundary extraction, which is defined as the difference between a given object and its morphological erosion by a 3×3 square-shaped SE (see Figure 3.11 (a)).

The intersection of the extracted boundary and the object depicted on the left of Figure 3.9 is subsequently obtained. This invariably produces approximately 7 disconnected objects (see Figure 3.11 (b)); four large ones and three small ones. The four large ones are associated with the boundary of the hand (1) between the thumb and index fingers, (2) along the wrist, (3) between the wrist and little finger, and (4) between the wrist and the thumb. The three small objects are associated with the webbing between the four fingers.

In order to ensure that each of the aforementioned objects are *morphologically connected*, said objects are enlarged through morphological dilation. Dilation is the inverse operation of erosion, and produces the union of the employed SE and the object in question, thereby enlarging the object in question by approximately half of the width of the employed SE.

Only the three small objects are subsequently retained (see Figure 3.11 (c)), after which the one in the middle may be identified by ranking their horizontal coordinates in an ascending fashion.

Finally, in order to identify the angle in question, the straight line between the center of mass of the entire pruned hand and the centrally located small object is considered, which may be visually illustrated by a right-angled triangle (see Figure 3.11 (d)). The angle in question, θ, is therefore defined as the inverse of the tangent function of the ratio of the opposite and adjacent sides of the aforementioned triangle.

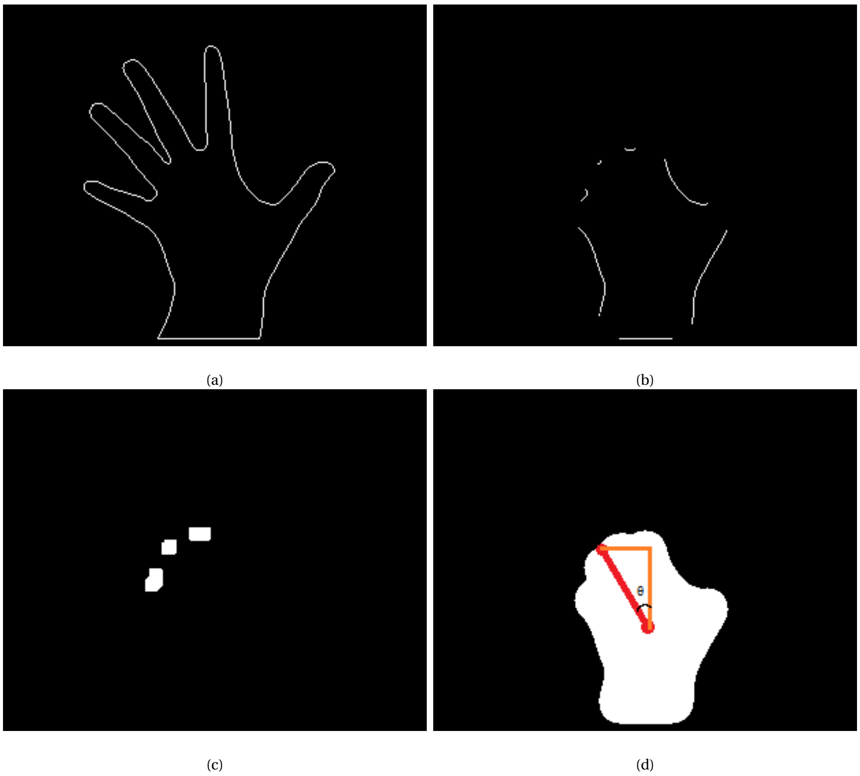


Figure 3.11: (a) The boundary of the detected hand. **(b)** The intersection of (a) and the object depicted on the left of Figure 3.9. **(c)** The three objects associated with the webbing between the four fingers that does not include the thumb. **(d)** A conceptualisation of the desired angle θ using a right-angled triangle.

3.2.1.5 Pruning the thumb and wrist

Recall that all SEs employed up to this stage in the ROI extraction protocol within the context of the Bosphorus database are isotropic. The removal of the thumb and wrist however depends on the notion that all objects in the Bosphorus database like the ones depicted in Figure 3.9 are vertically oriented (upright).

The removal of the thumb is accomplished by opening the objects depicted in Figure 3.9 by a vertical, one-dimensional SE of which the length is defined as 80% of the estimated height of said objects. This height is defined to be equal to twice the number of pixels between the center of mass and the webbing between the middle and ring fingers. The results of the aforementioned operation are depicted in Figure 3.12.

In order to remove the wrist and the remainder of the four fingers, the objects depicted in Figure 3.12 are opened by a circular SE with a diameter equal to 80% of the estimated width of the aforementioned objects. This width is defined as twice the length of the red line connecting the center of mass and the pixel located on the right boundary of said objects at the same vertical coordinate as the center of mass (see Figure 3.13 (a) and (b)). The results of the aforementioned protocol are shown in Figure 3.13 (c) and (d) respectively.

The final step towards obtaining the desired ROIs manually annotated in Figure 3.1 involves the selection of a bounding box of size 50×40 around the center of mass of the objects depicted in Figure 3.13 (c) and (d). This step constitutes the minor change to the original ROI extraction protocol proposed by Beukes [2]. The results of said operation are depicted in Figure 3.14 (a) and (b), while the vertically oriented version of the images depicted in Figure 3.1 are shown in Figure 3.14 (c) and (d) for reference. In Figure 3.14 (e) and (f), the

Figure 3.12: The result of opening the objects depicted in Figure 3.9 with a vertical, one-dimensional SE of which the length is defined as 80% of the estimated height of said objects.

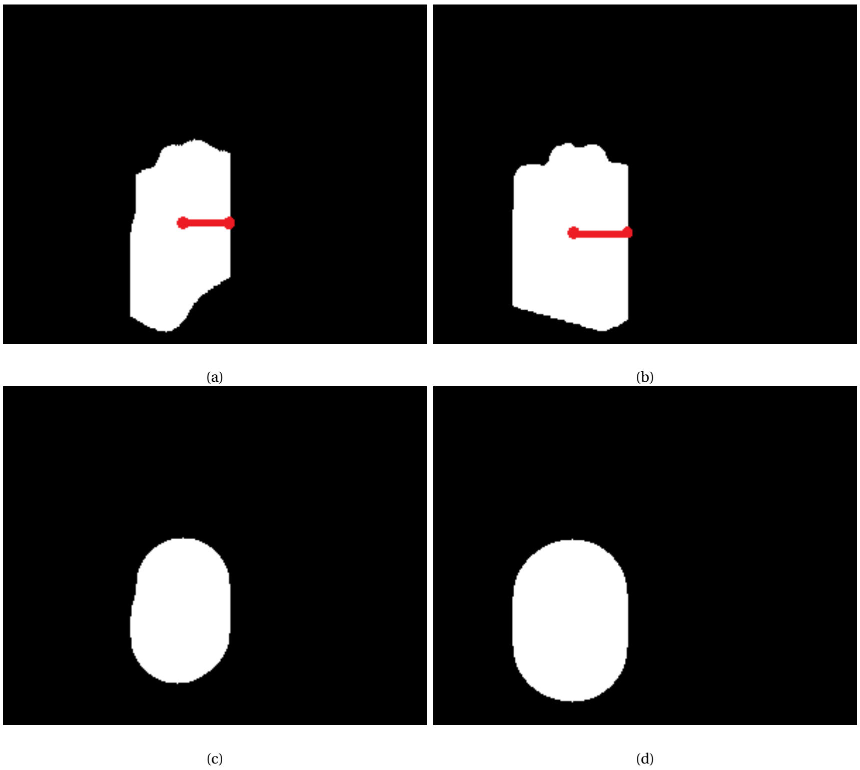


Figure 3.13: (a and b) Conceptualisation of the proposed protocol for estimating the width of the objects depicted in Figure 3.12. **(c and d)** The results of opening the objects in (a) and (b) by a circular SE with a diameter equal to 80% of the estimated width of said objects.

extracted ROIs are highlighted while being superimposed onto the images depicted in Figures 3.14 (c) and (d) in order to illustrate the accuracy of the ROI extraction protocol developed by Beukes [2] within the context of the Bosphorus dataset.

3.2.2 Extracting the region of interest: The Wilches database

A novel protocol for extracting a suitable ROI within the context of the Wilches database is developed in this study. Said protocol is significantly simpler than the one required within the context of the Bosphorus database. This is a result of major differences in the image acquisition protocols associated with the two databases. In Figure 3.15, two NIR images belonging to different individuals from the Wilches database are depicted, in which approx-

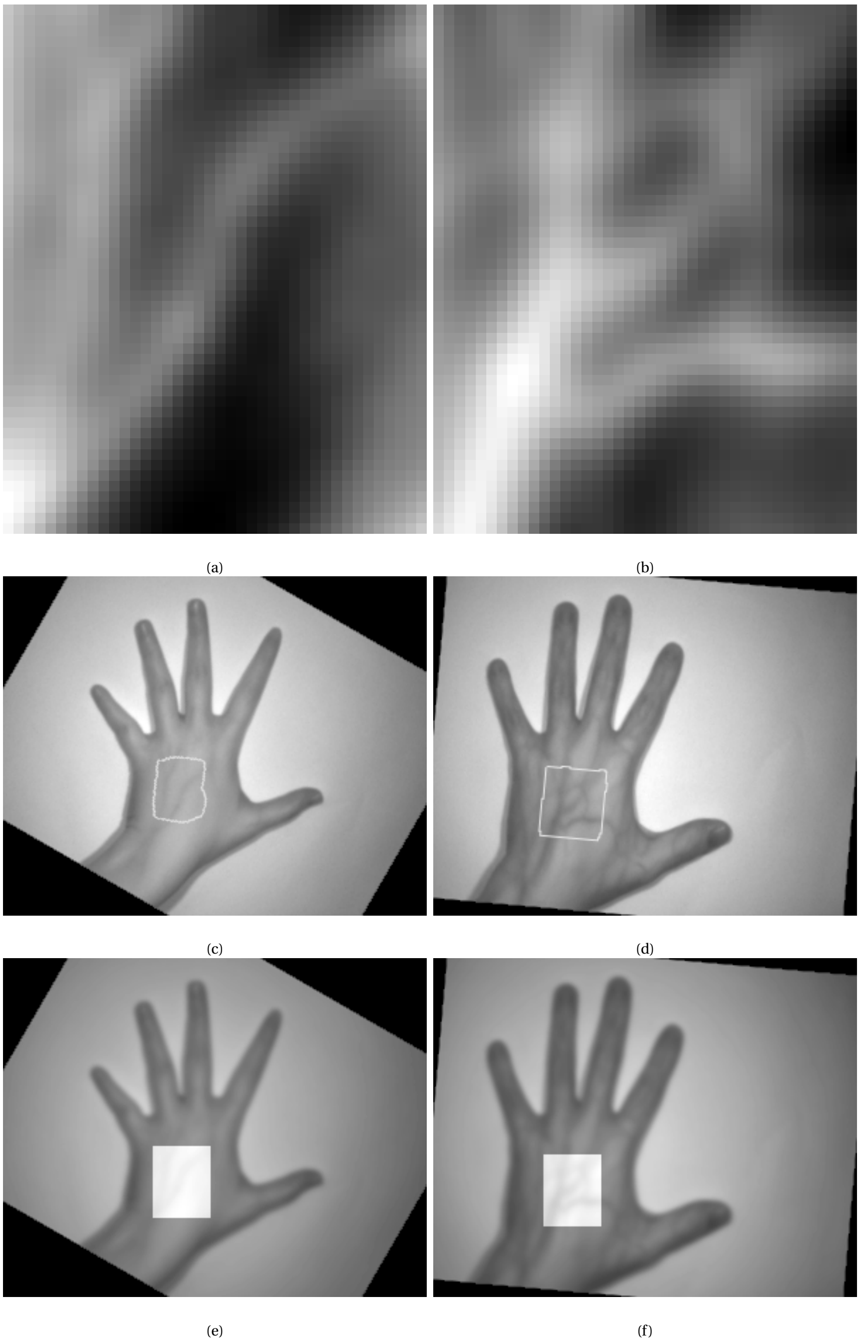


Figure 3.14: (a and b) The final ROIs of size 50 × 40 for the images depicted in Figure 3.1. **(c and d)** The vertically oriented versions of the images depicted in Figure 3.1. **(e and f)** The images depicted in (a) and (b) highlighted while being superimposed onto the images shown in (c) and (d).

imate boundaries of the desired ROIs are *manually* annotated in yellow.

The first step in extracting the desired ROI as illustrated in Figure 3.15 involves obtaining a thresholded image through Otsu's method. The results obtained when said method is applied to the images in Figure 3.15 are shown in Figure 3.16. Note that the ROIs within the context of the Wilches database are extracted in such a way that they have the same dimensions as the ROIs obtained within the context of the Bosphorus database, which is 50 × 40 pixels.

Next, the convex hulls of the objects depicted in Figure 3.16 are obtained and depicted in Figure 3.17.

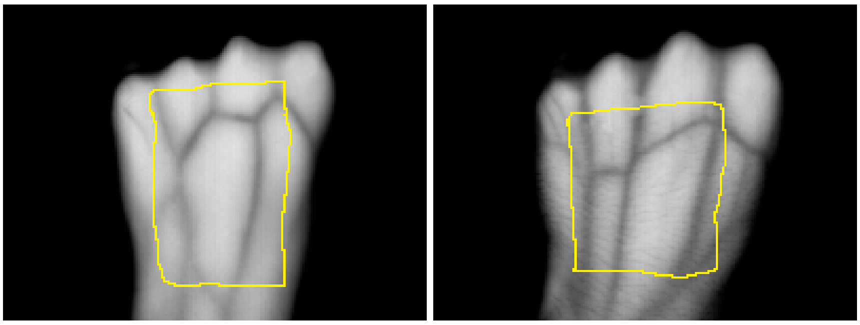


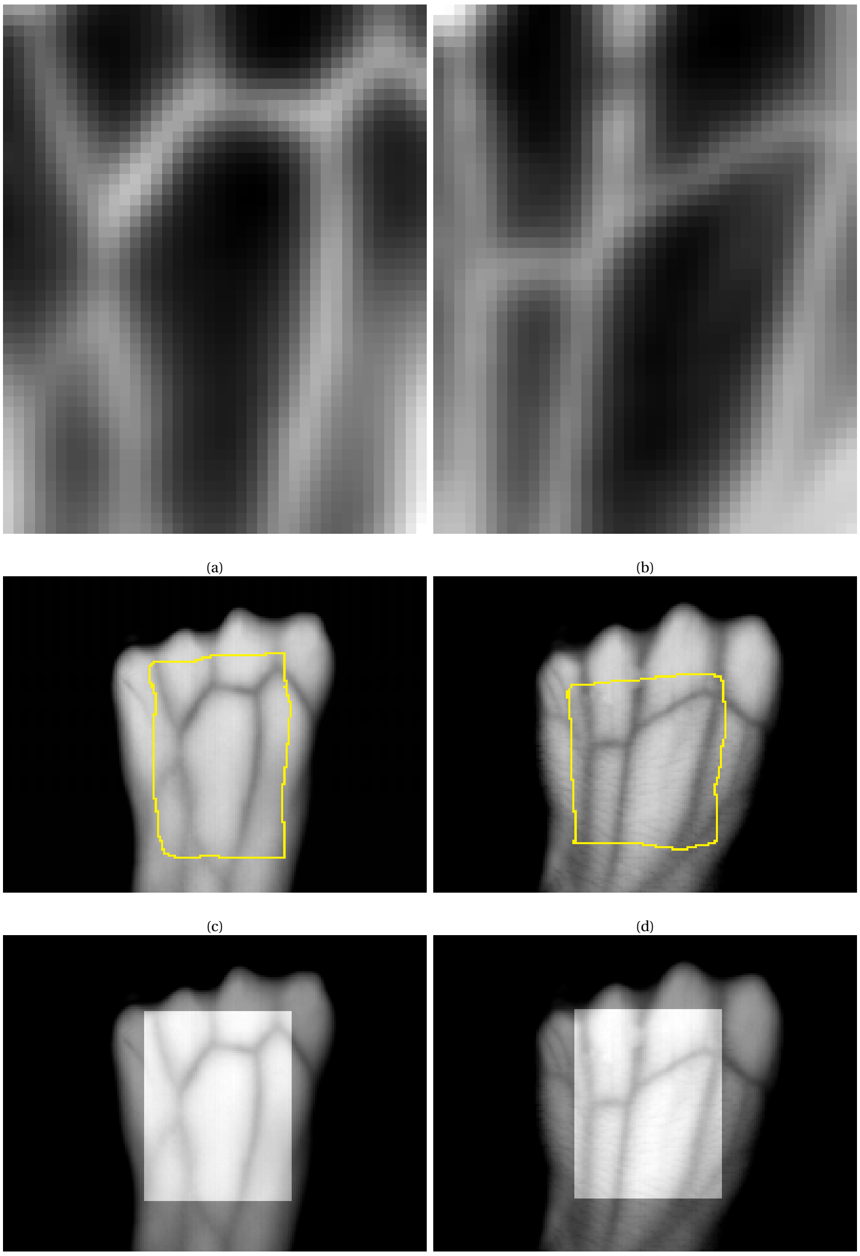
Figure 3.15: Two images from the Wilches database belonging to two different individuals. Approximate boundaries of the desired ROIs are *manually* annotated in yellow.

Figure 3.16: The images depicted in Figure 3.15 after the application of Otsu's method.

Figure 3.17: The convex hulls of the objects depicted in Figure 3.16.

In order to obtain the desired ROIs manually annotated in Figure 3.15, the following iterative protocol is developed: first, a rectangle centered on the center of mass of the objects depicted in Figure 3.17 with the same *aspect ratio* as the ROIs within the context of the Bosphorus database, that is $\frac{40}{50} = 0.8$, is considered. This rectangle is then iteratively enlarged while maintaining the same aspect ratio until at least one pixel on the boundary of said rectangle falls *outside* the convex hulls shown in Figure 3.17. Finally, since these ROIs will differ in dimensions across all the images in the Wilches database, they are resized to 50×40 pixels.

The results when the aforementioned protocol is applied to the images depicted in Figure 3.17 are shown in Figure 3.18 (a) and (b). In Figure 3.18 (c) and (d), the images depicted in Figure 3.15 are reproduced for reference. In Figure 3.18 (e) and (f), the images depicted in Figure 3.18 (a) and (b) are highlighted while being superimposed onto the images depicted in Figure 3.18 (c) and (d).



(a) (b) (c) (d) (e) (f)

Figure 3.18: (a and b) The final ROIs of size 50 × 40 for the images depicted in Figure 3.15. **(c and d)** Reproductions of the images depicted in Figure 3.15. **(e and f)** The images depicted in (a) and (b) highlighted and superimposed onto the images shown in (c) and (d).

3.3 Preprocessing the region of interest

The three levels of preprocessing that are applied to the extracted regions of interest (ROIs) are detailed in this section. These preprocessing protocols are identical for both databases and are therefore only discussed within the context of the Bosphorus database.

In summary: the three levels of preprocessing constitute (1) no preprocessing (see Section 3.3.1), (2) contrast enhancement (see Section 3.3.2) and (3) full binarisation (see Section 3.3.3). Note that *all images* from this point forward are of size 50 × 40 pixels, which is the fixed input size for the models developed in this study.

3.3.1 No preprocessing

When *No preprocessing* is performed, the extracted ROIs are fed *as is* into the models developed in this study. For reference, the examples of extracted ROIs provided in Sections 3.2.1 and 3.2.2 are depicted in Figure 3.19. It is important to note that, within the context of image processing, information of interest is typically represented by lighter pixels, while unwanted information is typically represented by darker pixels. The inverse of the extracted ROIs (see Figure 3.14 (e) and (f) and Figure 3.18 (e) and (f)) is therefore first obtained, since the hand veins constitute the information of interest.

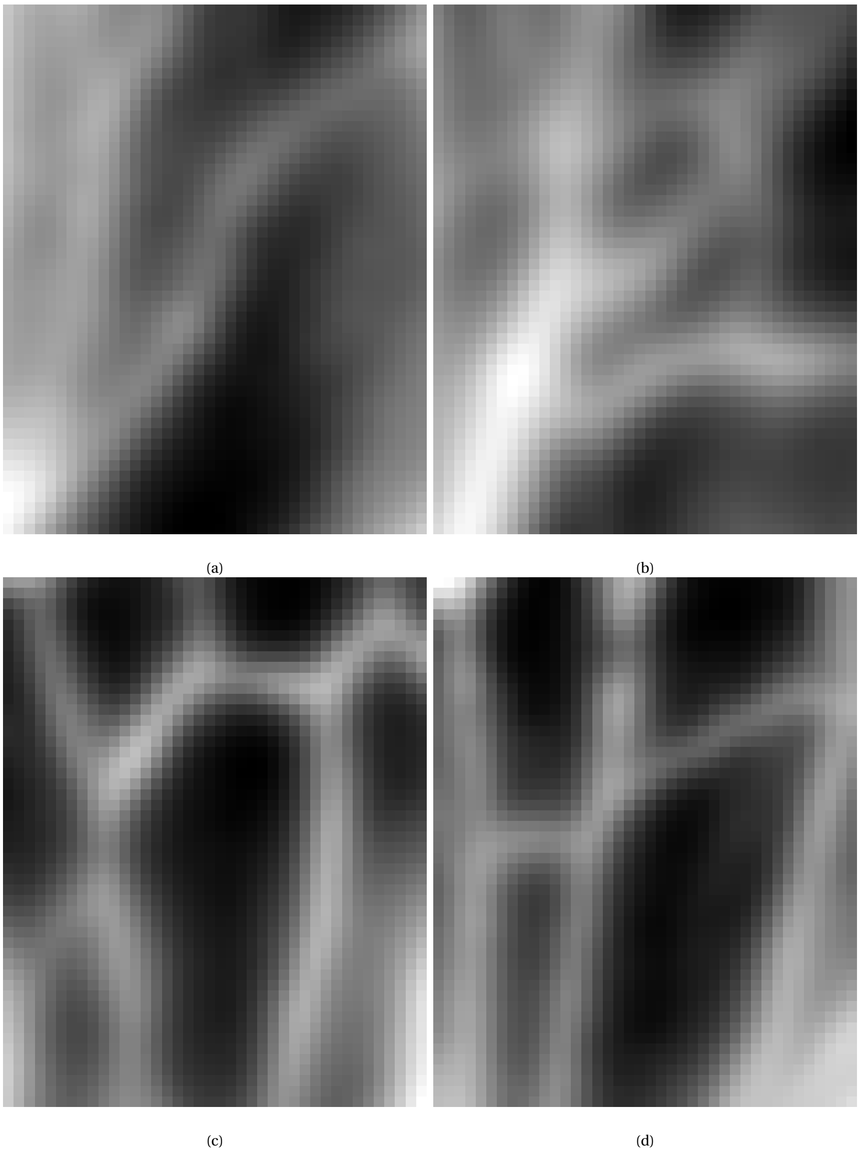


Figure 3.19: (a and b) The inverse of the final ROIs of size 50 × 40 for the images from the Bosphorus database depicted in Figure 3.1. **(c and d)** The inverse of the final ROIs of size 50 × 40 for the images from the Wilches database depicted in Figure 3.15.

This strategy serves as a benchmark in order to determine whether different levels of contrast enhancement may have a positive effect on the proficiency of the systems proposed in this study.

3.3.2 Contrast enhancement

Contrast enhancement is a common practice in the field of image preprocessing that accentuates the difference in brightness between objects and their background, thereby enhancing the *perceptibility* of said objects.

A number of contrast enhancement algorithms have been developed, which include histogram equalisation, adaptive histogram equalisation and contrast stretching. The contrast enhancement algorithm utilised in this study within the context of the extracted ROIs (see Figure 3.19) is called contrast-limited adaptive histogram equalisation (CLAHE), and was originally developed by Zuiderveld [13]. Reportedly, CLAHE produces less noise than the typical adaptive histogram equalisation algorithm, and is able to prevent so-called brightness saturation that is a common issue with standard histogram equalisation [49].

CLAHE is a local adaptive histogram equalisation algorithm which enhances contrast by iteratively applying histogram equalisation over different regions in the input image using a sliding window (kernel). The extent of contrast enhancement is controlled by a so-called clip limit $c \in [0,1]$, where larger values produce more contrast. In this study, the size of the kernel is set to 5×5 pixels, which is roughly 10% of the image size, while the clip limit is experimentally determined to be optimal at 0.03.

The results of applying CLAHE with a 5×5 kernel and a clip limit of 0.03 to the images depicted in Figure 3.19 are shown in Figure 3.20. Note that the ROIs are *first* smoothed by employing a Gaussian filter with $\sigma = 0.4$ in order to soften the boundaries.

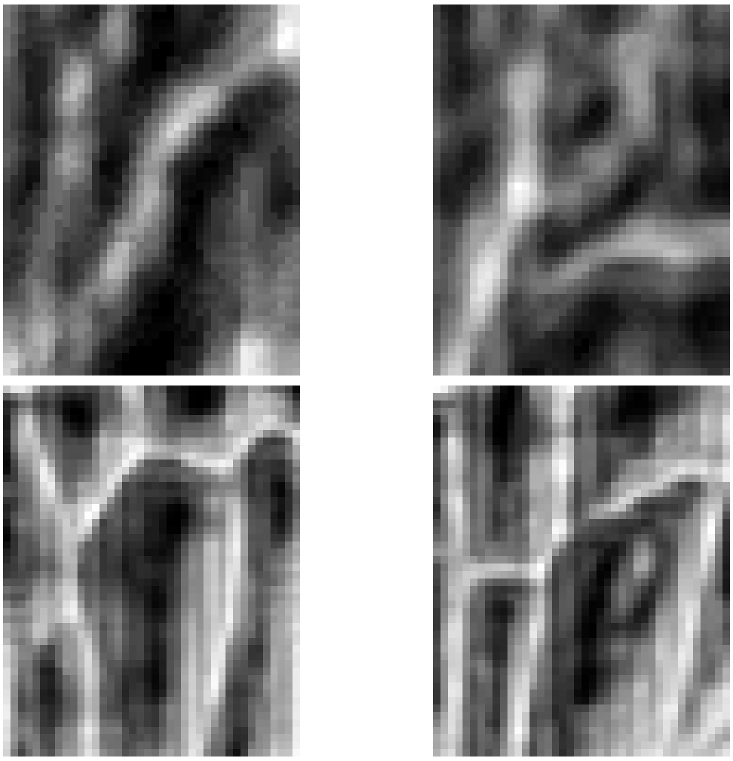


Figure 3.20: The images depicted in Figure 3.19 after the application of Gaussian smoothing with $\sigma = 0.4$ and CLAHE with a 5×5 kernel and a clip limit of 0.03.

3.3.3 Full binarisation

An overview of the novel protocol developed by Beukes [2] in order to identify the locations of vein pixels in the extracted ROIs within the context of the Bosphorus database is presented in this section for the purpose of convenience. It is demonstrated in this study that the aforementioned protocol is robust when applied to *multiple* databases, and is shown to achieve results of similar merit when utilised to identify the locations of vein pixels within the context of the Wilches *and* GenVeins database. This protocol is therefore *only* discussed within the context of the Bosphorus database.

The desired hand vein pixels for the images depicted in Figure 3.19 are manually annotated in yellow and shown in Figure 3.21.

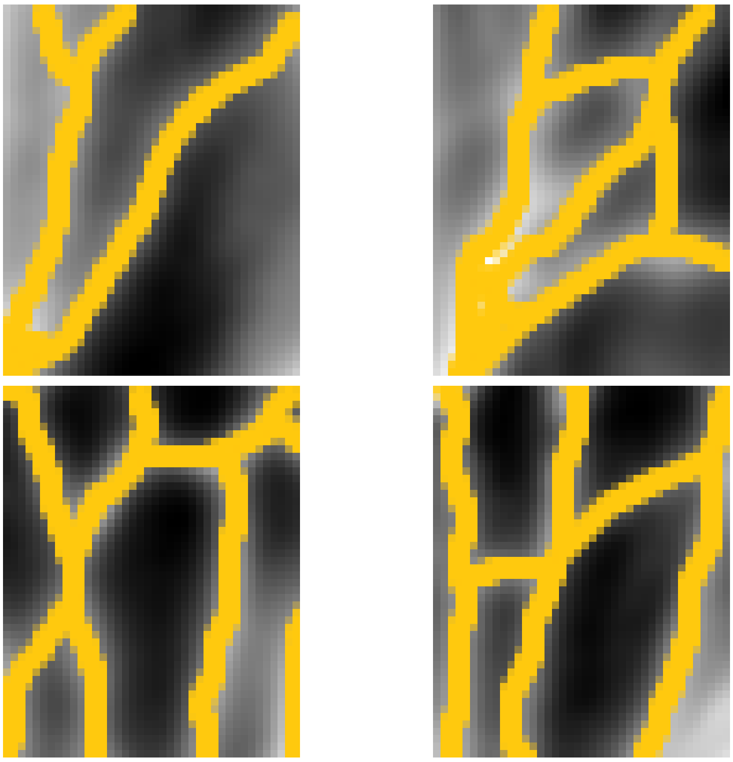


Figure 3.21: The images depicted in Figure 3.19 with the desired hand vein pixels manually annotated in yellow.

It is important to note that the employed hand vein segmentation protocol acts on the extracted ROIs (see Figure 3.14 (e) and (f) and Figure 3.18 (e) and (f)), and *not* on their inverse (see Figure 3.19). By investigating the properties of the hand veins within the images depicted in Figure 3.14 (e) and (f) and Figure 3.18 (e) and (f) respectively, the following three key criteria are defined which form the basis for the novel hand vein segmentation protocol proposed by Beukes [2] and employed in this study:

1. Hand veins are *darker* than their immediate surroundings.

2. Hand veins are *narrow* structures.

3. Hand veins are in the immediate vicinity of *fast-varying* greyscale intensities.

Separate algorithms are first employed in order to exploit each of the aforementioned criteria. The results of said algorithms are subsequently utilised to create a *seed* image for morphological reconstruction by dilation (see Section 3.3.3.5), which produces the final segmented image.

The protocol developed for the purpose of extracting dark and narrow regions in a greyscale image is detailed in Section 3.3.3.1, while the algorithm employed for the purpose of identifying regions that contain fast-varying greyscale intensities is discussed in Section 3.3.3.2.

3.3.3.1 Extracting dark and narrow regions

Dark and narrow regions are identified within a greyscale image by employing a morphological image processing method called the *black top-hat* (BTH) transform [50]. Recall that morphological image processing is fundamentally based on set theory, and uses a structuring element (SE) in order to select desired regions within an image. The BTH transform facilitates the retention of *dark* regions that fit entirely into the employed SE.

By selecting a disc-shaped SE of which the diameter slightly exceeds the estimated thickness of a typical hand vein, only dark *and* narrow regions within the extracted ROIs depicted in Figure 3.19 are retained. It is however important to note that, since the BTH is applied to a greyscale image, the resultant image is *also* greyscale, in which the pixel values constitute weighted responses of how well these values adhere to the targeted attributes (dark and narrow regions). Otsu's method is therefore applied to the resultant image produced by the BTH transform so as to (1) retain only the *strongest* responses and (2) acquire the first seed image for the purpose of morphological reconstruction.

The results of the BTH transform, when applied to the images depicted in Figure 3.19 and binarised through Otsu's method, are depicted in Figure 3.22.

Figure 3.22: The images depicted in Figure 3.19 after the application of the BTH transform and Otsu's method.

It is clear from Figure 3.22 that the BTH transform is well-suited for the purpose of identifying dark and narrow regions within a given greyscale image. It is however important to note that a significant number of vein pixels are *not* contained in the images depicted in Figure 3.22, since Otsu's method has been applied for the purpose of retaining only the strongest responses associated with the BTH transform. The reason for this approach is to maximise the likelihood that the white pixels contained in the result are in fact associated with a vein pixel, since these images are utilised as seed images for morphological reconstruction. In other words, it is not advisable to morphologically reconstruct regions that are *not* associated with the targeted hand veins (see Figure 3.21).

3.3.3.2 Extracting regions that contain fast-varying intensities

The well-known *Laplacian-of-Gaussian* (LoG) transform is employed for the purpose of identifying regions that contain fast-varying intensities. Said transform is well-suited for the aforementioned task, since it calculates the second derivative of an image, which results in strong responses in the vicinity of fast-varying intensities.

Only the strongest responses are again retained by applying Otsu's method to the result of the LoG transform. The results of the LoG transform, when applied to the images depicted in Figure 3.19 and binarised through Otsu's method, are depicted in Figure 3.23.

Figure 3.23: The images depicted in Figure 3.19 after the application of the LoG transform and Otsu's method.

It is clear from Figure 3.23 that the application of the LoG transform yields significantly more pixels that satisfy the criteria of being located in the *vicinity* of fast-varying intensities. The black pixels within the images depicted in Figure 3.23 are associated with so-called *inflection points* where the sign of the curvature changes, and the LoG transform response is zero-valued.

3.3.3.3 Seed image for morphological reconstruction

In order to obtain the final seed image for morphological reconstruction, the images depicted in Figures 3.22 and 3.23 are combined by calculating the morphological *intersection*. This ensures that the regions in the final seed image adhere to *all* three key criteria outlined in the beginning of Section 3.3.3. The resultant images obtained by calculating the morphological intersection between the images depicted in Figures 3.22 and 3.23 are shown in Figure 3.24.

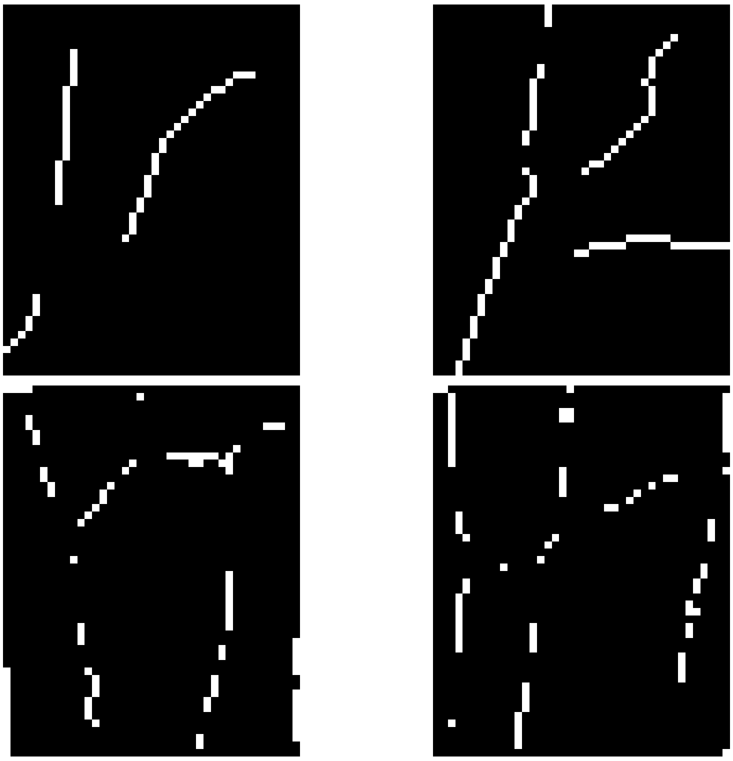


Figure 3.24: The morphological intersection of the images depicted in Figures 3.22 and 3.23.

The protocol employed for the purpose of obtaining a suitable *mask* image for morphological reconstruction is discussed in the following section.

3.3.3.4 Mask image for morphological reconstruction

The purpose of the mask image within the context of morphological reconstruction by dilation is to restrict the dilation of the seed image to the boundaries associated with the targeted hand veins. Said image should therefore contain as many of the desired hand vein pixels as possible, while also having well-defined boundaries of the desired hand veins. It is therefore proposed by Beukes [2] that a textural segmentation method called moving average thresholding [50] is employed for the purpose of obtaining a mask image that satisfies both of the aforementioned criteria.

Moving average thresholding is a local thresholding method whereby the threshold value of each pixel in the input image is determined by calculating the average pixel intensity in a given neighbourhood of the pixel in question. For example, if the pixel in question has a larger intensity than the average pixel intensity of the neighbourhood, said pixel is assigned a value of 1.

The result of applying moving average thresholding with a 10 × 10 pixel neighbourhood to the images depicted in Figure 3.19 are shown in Figure 3.25.

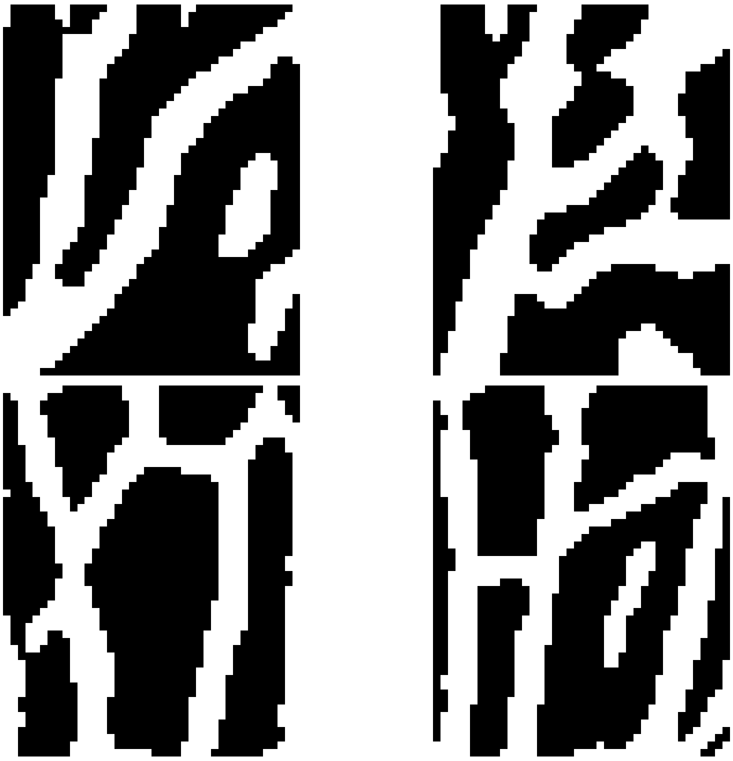


Figure 3.25: The images depicted in Figure 3.19 after the application of moving average thresholding with a 10 × 10 pixel neighbourhood.

3.3.3.5 Morphological reconstruction by dilation

Morphological reconstruction by dilation comprises of the *iterative* application of morphological dilation to the seed image until the boundaries of the objects within the mask image are reached. Recall that the seed image is obtained by meticulously selecting certain regions in the input greyscale image that has the highest likelihood of being associated with the regions targeted for reconstruction (the hand veins), while the purpose of the *mask* image is to restrict the enlargement of the regions within the seed image so as to achieve the desired morphology of the reconstructed regions.

The results obtained after the application of morphological reconstruction by dilation to the seed images depicted in Figure 3.24, while being restricted by the mask images shown in Figure 3.25, are presented in Figure 3.26. The images depicted in Figure 3.21, in which the desired hand veins are manually annotated in yellow, are also shown in the left column of Figure 3.26 in order to illustrate the proficiency of the employed binarisation protocol originally proposed and developed by Beukes [2].

3.4 Concluding remarks

In this chapter it is illustrated that the proficiency and robustness of the novel hand vein binarisation protocol originally proposed by Beukes [2] is retained when applied to another

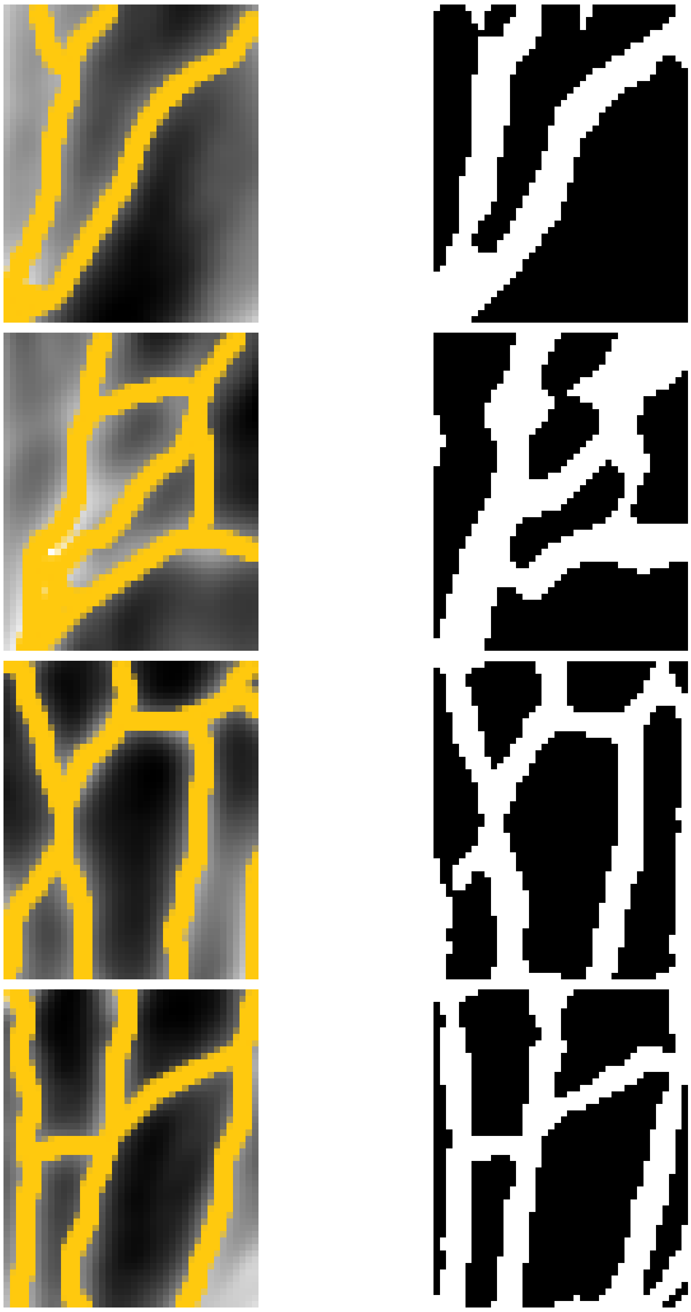


Figure 3.26: (Left) The images depicted in Figure 3.21, in which the desired hand veins are manually annotated in yellow. **(Right)** The images depicted in Figure 3.19 after the application of morphological reconstruction by dilation with the seed and mask images depicted in Figures 3.24 and 3.25 respectively.

dorsal hand vein database, namely the Wilches database. The protocol developed in this study for the purpose of generating the GenVeins database is detailed in the following chapter.

Chapter 4

The GenVeins database

4.1 Introduction

The protocol developed in this study for the purpose of generating an artificial hand vein database, called GenVeins, is detailed in this chapter. The purpose of the aforementioned protocol is to obtain a hand vein database that contains images of a sufficiently *large* number of *different* fictitious individuals, since the available *actual* hand vein databases contain images from only a limited number of different individuals. The development of the aforementioned protocol is motivated by experimental results of the so-called "benchmark experiment" (BME) (see Section 8.6.1) and the so-called "augmented BME" (see Section A.4), which indicate that the networks proposed in this study are not able to generalise well during testing when trained, validated and tested on mutually exclusive sets of an *insufficient* number of *different* individuals, regardless of the number of available samples for each individual. The proposed protocol may be outlined as follows:

1. One unique sample of so-called *base veins* (see Section 4.2.1) is first generated, which represents the main vein structure of a fictitious individual.

2. Four copies of each unique sample are subsequently created, after which each copy is supplemented by randomly adding so-called *auxiliary veins* (see Section 4.2.2). The purpose of this step is to simulate minor structural differences that arise during the acquisition of multiple samples of the hand veins belonging to an actual individual.

3. Each supplemented copy, which constitutes a binary image, is then "*greyed*" in order to obtain four non-binary regions of interest (ROIs) for each artificial individual (see Section 4.3.1). The purpose of this step is to obtain samples similar to the ROIs of dimensions 50×40 pixels obtained in Section 3.3.1.

4. Each artificial non-binary ROI is finally subjected to the same preprocessing protocol as the one proposed in Section 3.3.

In this study a set of four artificial hand vein samples are generated for a total of 20 000 fictitious individuals by employing the aforementioned protocol, resulting in a total of 80 000 artificial hand vein samples in the GenVeins database.

4.2 Generating binary hand veins

The algorithm proposed for the purpose of generating binary hand veins is fundamentally similar for each type of vein. A novel tree generation algorithm is proposed, which consists of the following steps:

1. The root and tip of each vein constitute two random *seed* points in a black image with dimensions of 50×40 pixels.

2. A number of so-called *connection* points are subsequently obtained through randomisation. These connection points serve as points in a piecewise linear function associated with the hand vein in question.

3. The vertical spacing between the connection points is uniform, while the horizontal spacing between successive connection points is randomised within a suitable margin. This facilitates a unique structure for each hand vein.

4. The connection points are finally iteratively connected by obtaining line segments between successive connection points, starting from the root seed point and ending at the tip seed point.

The constraints imposed on the two random seed points associated with each vein determine the specific vein type that is generated. It is important to note that the proposed hand vein generation algorithm employs a number of random values that are controlled by *experimentally calibrated* constraints in order to achieve the desired output.

4.2.1 Generating the base veins

The term *base veins* refers to the basic structure of the dorsal veins of the individual and comprises of so-called *trunk* veins and *branch* veins. Trunk veins are typically the thickest veins which span most of the vertical distance of the ROI, while the branch veins are slightly thinner and serve as so-called *connector* veins between the trunk veins. The constraints associated with the generation of the trunk veins are provided below.

1. The number of trunk veins is randomly selected from a discrete set of values defined by $n \in [2,3]$ with weighted probabilities of $p \in [0.65, 0.35]$ respectively.

2. The seed points associated with trunk veins invariably lie on the borders of the ROI, where the two borders associated with a seed point pair are selected at random, but favouring the top and bottom borders with a 0.6 probability, since trunk veins typically run towards the fingers of the hand.

3. The squared Euclidean distance between each pair of seed points must be greater than or equal to 55% of the length of the diagonal of the ROI.

4. The vertical distance between each pair of seed points must be greater than or equal to 80% of the height of the ROI.

5. The vertical angle between any pair of trunk veins is defined by $\theta \in [15°, 45°]$. This ensures that any pair of trunk veins are neither excessively parallel nor orthogonal to each other.

The constraints associated with the generation of the branch veins are as follows:

1. The number of branch veins is randomly selected from a discrete set of values defined by $n \in [0, 1, 2, 3]$ with weighted probabilities of $p \in [0.2, 0.35, 0.3, 0.15]$ respectively.

2. The seed points associated with a branch vein invariably lie on two different trunk veins.

3. The vertical distance between each pair of seed points must be greater than or equal to 40% of the height of the ROI. This ensures that branch veins are not excessively horizontal.

A constraint on the percentage of white pixels is also placed on the final base vein image which contains the trunk and branch veins, in order to further ensure sufficiently sparse hand vein structures. The acceptable range is defined as $s \in (0.35, 0.7)$.

If *any* of the aforementioned constraints are *not* met while generating a trunk or branch vein, the process is restarted. It is again important to note that, since the aforementioned protocol may result in infinite restarts, the specific values associated with the constraints were *experimentally calibrated* in order to ensure that this will not occur. The main constraints that required careful calibration constitute the vertical angle between any pair of trunk veins defined by $\theta \in [15°, 45°]$ and the acceptable range of white pixels in the final base vein image defined as $s \in (0.35, 0.7)$. Various feasible ranges were considered and tested on a number of significantly smaller versions of the GenVeins database (a total of 50 fictitious individuals each) in order to establish whether or not any infinite restarts occur while maintaining high quality generated hand vein samples.

Four binary base vein samples that were generated by the aforementioned algorithm are depicted in Figure 4.1. Each sample is associated with a *different* fictitious individual.

The protocol for creating four copies of the base veins associated with a *specific* fictitious individual and randomly adding so-called *auxiliary* veins to each copy is explained in the following section.

4.2.2 Adding the auxiliary veins

Auxiliary veins comprise of so-called *twig* veins and *unconnected* veins. Twig veins constitute "expansion" veins of the basic hand vein structure, while the term "unconnected veins" refers to those structures that appear on a NIR image which *look* like veins and have the same grey-scale intensity as veins, but may or may not be associated with the actual hand vein structure. Twig veins and unconnected veins may therefore be employed within the context of the proposed algorithm in order to obtain four *similar* hand vein samples of a *specific* fictitious individual, since they may or may not be detected by the preprocessing strategies presented in Chapter 3. The constraints associated with the generation of the twig veins are provided below.

1. The number of twig veins is randomly selected from a discrete set of values defined by $n \in [0, 1, 2]$ with weighted probabilities of $p \in [0.3, 0.5, 0.2]$ respectively.

2. At least one of the seed points associated with twig veins invariably lie on one of the trunk veins.

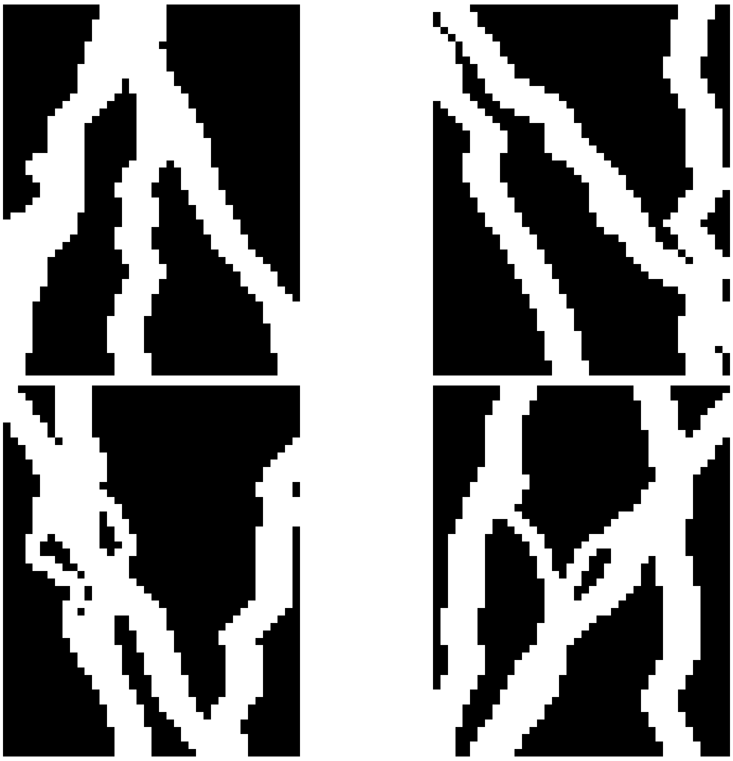


Figure 4.1: Four binary base vein samples associated with *different* fictitious individuals.

3. The number of vertical *and* horizontal pixels between each pair of seed points are defined by $d_i \in [10,20]$ and $d_j \in [10,20]$. This ensures a well-defined ratio between the length of the trunk veins and twig veins.

The constraints associated with the generation of the unconnected veins are as follows:

1. The number of unconnected veins is randomly selected from a discrete set of values defined by $n \in [0,1,2]$ with weighted probabilities of $p \in [0.3,0.5,0.2]$ respectively.

2. There are *no* constraints on the locations of the seed points associated with an unconnected vein.

3. The number of vertical *and* horizontal pixels between each pair of seed points are defined as $d_i \in [10,20]$ and $d_j \in [10,20]$.

If *any* of the aforementioned constraints are *not* met during the generation of an auxiliary vein, the auxiliary vein generation process is restarted, as is the case within the context of the generation of the base veins.

It is important to note that minor spatial variation is also incorporated while creating four supplemented copies of each fictitious hand vein sample. The purpose of this is to simulate possible variations in orientation between multiple hand vein samples associated with a specific *actual* individual that may occur during acquisition. The algorithm for adding spatial variation to each of the four supplemented copies associated with a specific *fictitious* individual is outlined below.

- A unique rotation through a random angle θ, where $\theta \in [-1°,1°], \theta \in R$, followed by

- A unique translation of the pixel values in the x and y directions, where $x, y \in \{-1, 0, 1\}$.

The images depicted in Figure 4.1 are reproduced for convenience in the left column of Figure 4.2, while *one* of the four supplemented copies of the corresponding image in the left column, that is obtained after adding auxiliary veins, is shown in the right column of Figure 4.2.

The algorithm developed in this study for the purpose of greying the final binary hand vein samples associated with a fictitious individual is detailed in the next section. The results obtained when subjecting the non-binary versions of the aforementioned samples to the preprocessing strategies explained in Chapter 3 are subsequently presented.

4.3 Preprocessing

In order to properly evaluate the proficiency of the systems proposed in this study, the binary hand vein images depicted in the right column of Figure 4.2 need to be subjected to the same preprocessing protocols presented in Chapter 3. The aforementioned images therefore first need to be *greyed* so as to obtain images similar to the non-binary ROIs obtained in Section 3.3.1. The non-binary version of the greyed fictitious hand vein images is subsequently subjected to the CLAHE-based and full binarisation preprocessing protocols proposed in Sections 3.3.2 and 3.3.3 respectively.

4.3.1 Greying the artificially generated binary hand veins

The proposed protocol for greying the binary hand vein images such as those depicted in the right column of Figure 4.2 involve two initial greying algorithms, followed by the iterative application of smoothing algorithms and morphological erosion. It is important to note that these protocols were *experimentally calibrated* in order to achieve the desired output, that is non-binary ROIs similar to those obtained in Section 3.3.1, within the context of the Bosphorus and Wilches databases respectively. The main constraint that required careful calibration constitutes the combination of the smoothing factor and the extent of morphological erosion during each iteration. Various feasible ranges were considered and tested on a number of significantly smaller versions of the GenVeins database (a total of 50 fictitious individuals each) in order to establish whether or not the output appears feasible for all the greyed images. The quality of the greyed images were visually compared to the extracted ROIs of the Bosphorus and Wilches databases (see Figure 3.19) in order to ensure the artificially generated grey-scale images resemble actual hand vein images.

The generated binary hand vein images are first *slightly* thinned by utilising binary erosion with a disc-shaped SE of appropriate size *before* they are subjected to the greying protocol. The purpose of the aforementioned step is to mitigate the effect of so-called "over-smoothing" that may produce blob-like structures as opposed to vein-like structures.

The first stage of the greying protocol is based on altering the intensity of the vein and background pixels in order to produce an initial non-binary image. Each pixel in the eroded hand vein image is assigned a completely new intensity, which is randomly selected from an acceptable grey-scale intensity range based on whether the pixel is associated with a vein or the background.

The first stage of the greying protocol however invariably produces significant discontinuities between the intensities of neighbouring vein or background pixels. This effect is mitigated by the second stage of the greying protocol, which involves a series of three ordered combinations of two local smoothing algorithms, namely the local *mean* and *median*. Local smoothing algorithms iterate over the input image and calculate the local intensity based on the employed mathematical function in a defined neighbourhood of each pixel, and assigns the resulting intensity to the pixel in question. The result of each of the three ordered combinations of smoothing algorithms is then subjected to Gaussian smoothing with $\sigma = 2.5$, after which the result is subjected to grey-scale morphological erosion with a disc-shaped SE of which the diameter is equal to the approximate thickness of the desired hand veins. This effectively mitigates the so-called *overflow* of vein pixels across their initial boundaries. The thickness of the greyed veins should in other words remain more or less the same as the thickness of the generated binary hand veins for a given image.

The results when the images depicted in the right column of Figure 4.2 are subjected to the greying protocol detailed in this section are presented in the right column of Figure 4.3, while the original images depicted in the right column of Figure 4.2 are again presented in the left column of Figure 4.3 for convenience.

4.3.2 Final results

The images depicted in the right column of Figure 4.3 constitute the simulated ROIs which contain the dorsal hand veins of four *different* fictitious individuals. These images therefore represent the ROIs after being subjected to *no* preprocessing, similar to those images acquired in Section 3.3.1 within the context of the Bosphorus and Wilches databases respectively.

In the left column of Figure 4.4, the aforementioned images are shown again for reference purposes. The contrast enhanced and full binarised versions of these images obtained after being subjected to the proposed contrast enhancement and binarisation protocols (see Sections 3.3.2 and 3.3.3) are depicted in the centre and right column of Figure 4.4 respectively.

4.4 Concluding remarks

A novel protocol developed in this study for the purpose of generating artificial hand vein samples of fictitious individuals is detailed in this chapter. It is clear from the results obtained in Section 4.3.2 that the aforementioned protocol is proficient in achieving images that successfully *mimic* actual hand vein samples. These images are subsequently utilised in order to train and validate the proposed systems within the context of the individual *independent* scenario (see Section 8.6) developed in this study for the purpose of authenticating a hand vein sample associated with *any* unseen *actual* individual. The theory associated with the neural network architectures employed in this study is detailed in the following chapter.

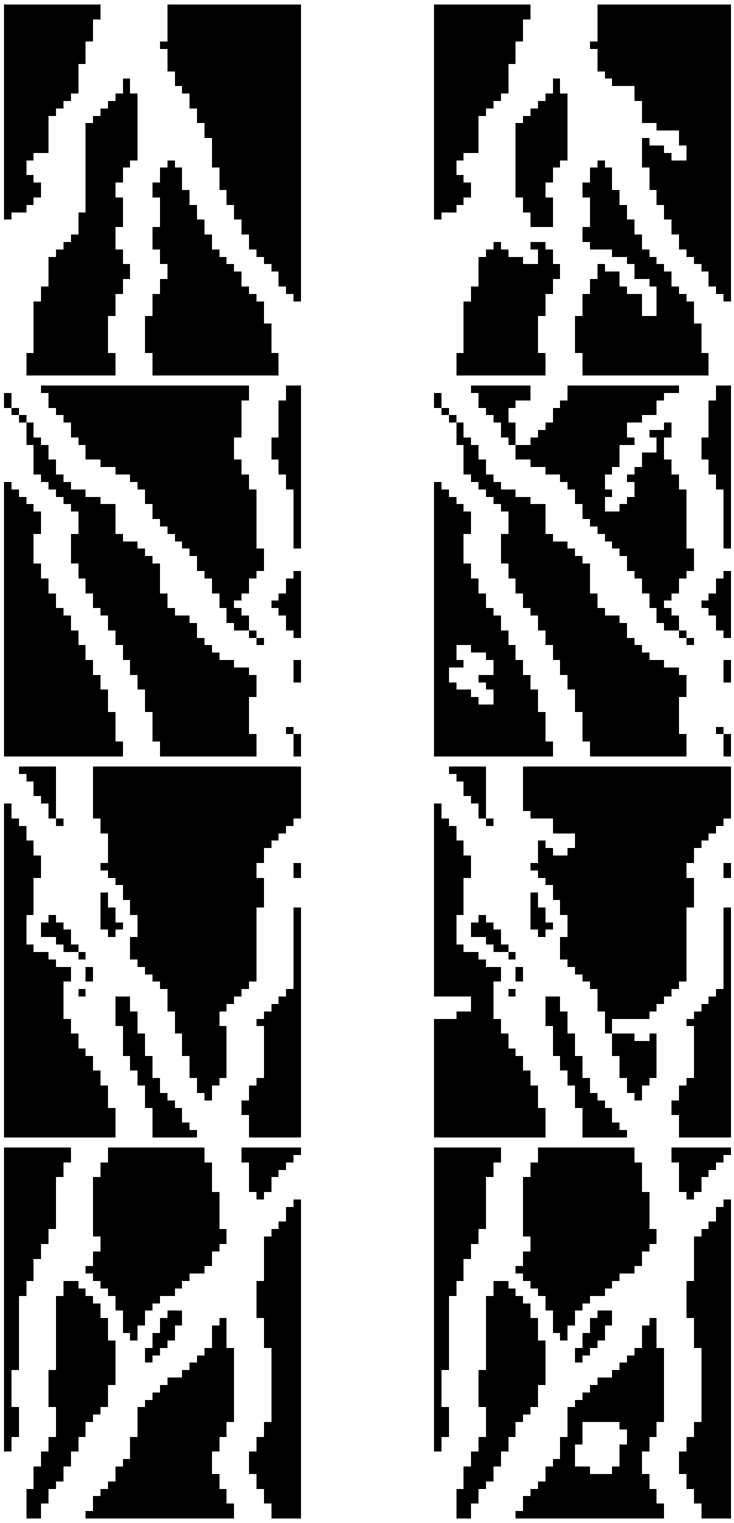


Figure 4.2: (Left) The images depicted in Figure 4.1. **(Right)** One of the four supplemented copies of the corresponding image to the left that is obtained after adding auxiliary veins.

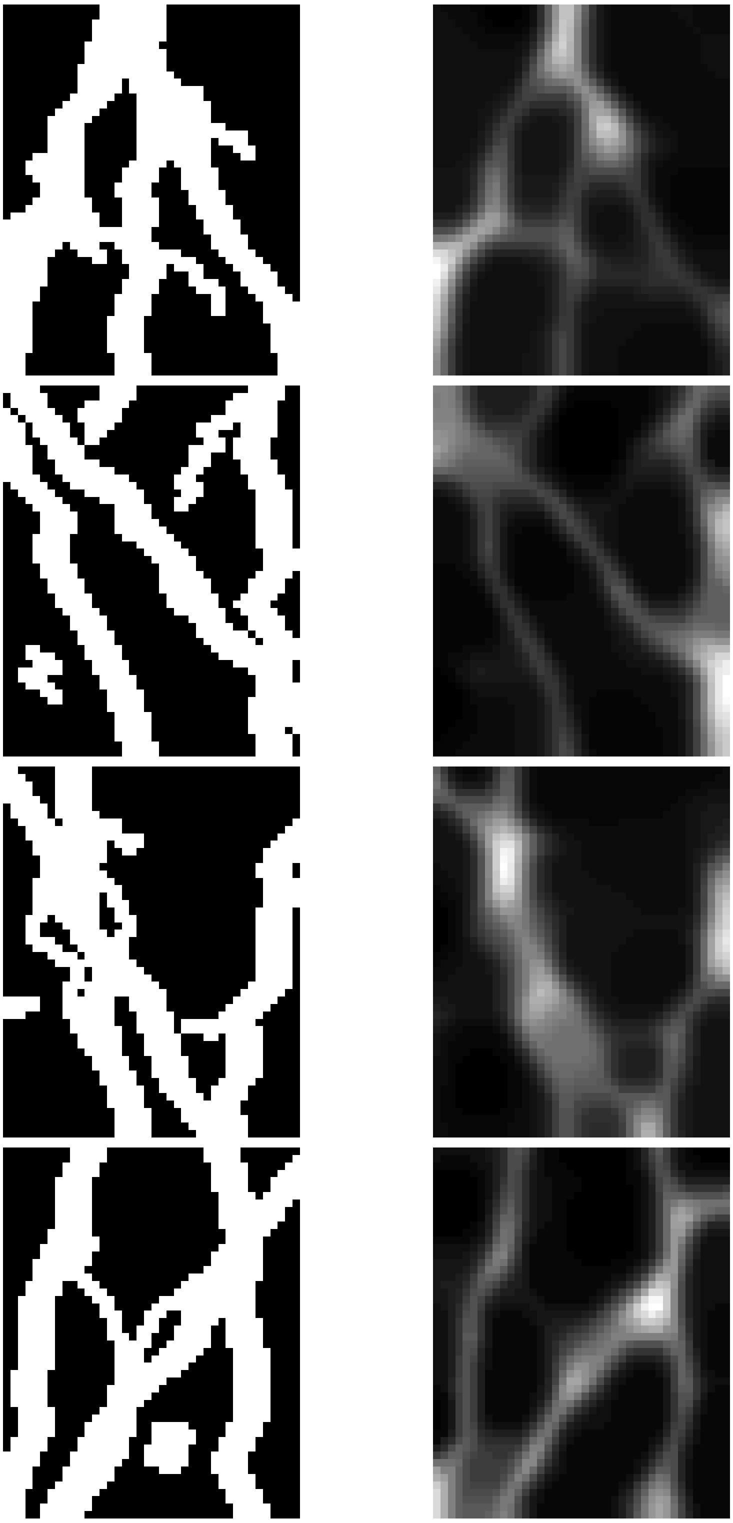


Figure 4.3: (Left) The images depicted in the right column of Figure 4.2. **(Right)** The images depicted on the left after being subjected to the proposed greying protocol.

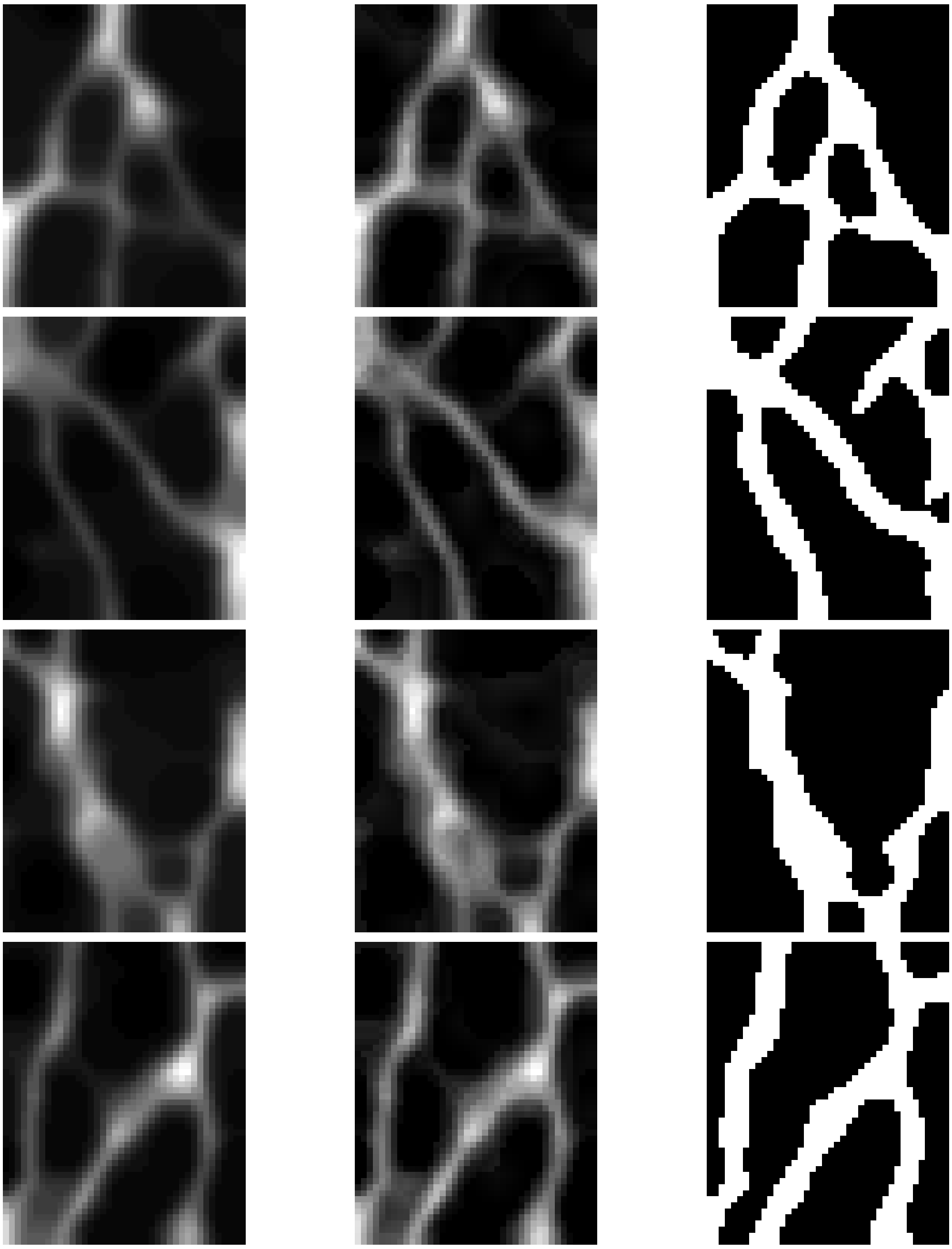


Figure 4.4: (Left) The images depicted in the right column of Figure 4.3. **(Center)** The images depicted in the right column of Figure 4.3 after the application of the contrast enhancement protocol proposed in Section 3.3.2. **(Right)** The images depicted in the right column of Figure 4.3 after the application of the full binarisation protocol proposed in Section 3.3.3.

Chapter 5

Neural networks

5.1 Introduction

The theory associated with the neural network architectures employed in this study is detailed in this chapter. A brief history of the evolution of artificial neural networks (ANNs) is first provided in Section 5.2, after which the relevant theory associated with the ANNs developed in this study is explained in Section 5.3. The group of neural network architectures employed in this study, namely *convolutional* neural networks (CNNs), is subsequently discussed in Section 5.4. A specific subset of CNNs which constitutes the basis of the majority of systems proposed in this study, namely similarity measure networks (SMNs), is finally detailed in Section 5.5.

5.2 A brief history of artificial neural networks

The concept of neural networks as computational algorithms, more commonly known as artificial neural networks (ANNs), was first introduced in 1943 by McCulloch and Pitts [51], in which the authors designed an electrical circuit that is able to simulate the functioning of neurons in the human brain. A specific branch of calculus referred to as propositional logic is utilised for the purpose of simulating the activation pattern of the artificial neurons, and is deemed to be a feasible approach due to the so-called "all-or-nothing" characteristic of neurons in the human brain. Neurons that are modelled on the aforementioned characteristic later became known as perceptrons.

The next contribution to ANNs was made in 1949 by Donald Hebb in his book "The Organisation of Behaviour" [52], in which he concluded that (1) neurons in the human brain are strengthened upon successive activation and (2) the connection between neurons is strengthened if they are active at the same time. Hebb's theory of learning [52] later became known as Hebbian Learning, and is commonly phrased as "neurons that fire together, wire together".

In 1955 at the Institute of Radio Electronics National Convention, the Professional Group on Electronic Computers sponsored a symposium titled "The Design of Machines to Simulate the Behavior of the Human Brain", of which the transcript is published by McCulloch *et al.* [53]. One of the panel members of the aforementioned symposium was an IBM researcher named Nathanial Rochester, who later lead the first attempt at simulating an ANN on a computer during a summer research project at Dartmouth College, New Hampshire in 1956. A summary of the research proposal is published by Rochester *et al.* [54], while the full proposal is currently housed at Dartmouth College and Stanford University. The motivation for the project, as presented by Rochester *et al.* [54], was the notion that any characteristic of human intelligence can, in principle, be described in sufficient detail so as to enable a computer simulation thereof. This project is widely considered as the birthplace of artificial intelligence.

A psychologist by the name of Frank Rosenblatt was the first to create a computer-based implementation of a perceptron in 1958 at Cornell Aeronautic Laboratory. The aforementioned machine is called the Mark I Perceptron, and constitutes a simple input-output system which calculates a weighted sum of the input vector and a weight vector. The weighted sum is finally converted to either a 0 or 1 by employing a suitable threshold. The application of the threshold in this scenario is referred to as an activation function (see Section 5.3.2), which is used to map the weighted sum to an output value that determines whether the neuron is active or not. The Mark I Perceptron was trained by passing in multiple input vectors and minimising the difference between the machine output and the desired output by adjusting the weight vector. This training protocol is commonly known as the perceptron learning rule (PLR), and is based on Hebbian Learning. A major drawback of the PLR however is that it is unable to train on linearly non-separable input vectors, due to the error being restricted to a discrete value $\mathrm{e} \in \{-1, 0, 1\}$.

In 1959, two scientists at Stanford University by the names of Bernard Widrow and Marcian Hoff proposed the concept of a so-called ADALINE neuron [55] which comprises of a weight vector, a threshold and so-called "adaptive linear elements" which are utilised for the purpose of automatically adjusting the weight vector during training. The training protocol associated with the ADALINE neuron is termed the Widrow-Hoff or delta learning rule (DLR). Two major differences between the DLR and the PLR is that (1) the error in the DLR is not restricted and (2) the DLR is differentiable for *any* activation function, whereas PLR is only differentiable when a threshold-based activation function is employed. The ADALINE neuron was developed in an effort to predict the next binary value (bit) in a series of bits, and was subsequently utilised in a system called MADALINE. The MADALINE system constitutes the first ANN that has been successfully applied in a real world scenario for the purpose of removing noisy echoes from telephonic conversations, and is still in use to this day.

During the 1960s, ANNs had lost their popularity due to a number of reasons, of which the main one was a book by Marvin Minsky [56]. In his book, Minsky concluded amongst other things that the extension of single-layer ANNs to multi-layer ANNs (see Section 5.3.3) is not possible, since it would take an infinite number of training iterations in order to determine the optimal set of weights. In addition to the aforementioned critique, many learning functions which were used to train ANNs at that time were fundamentally flawed.

The aforementioned issues were first addressed in 1974 by an American scientist named Paul Werbos [57]. He was the first to show that a well-known mathematical concept called backpropagation is ideal for the purpose of training multi-layer ANNs, since it reduces the significance of each error depending on when it occurred in the chain of activations. This had been the first major breakthrough in the field of artificial intelligence in more than a decade.

In 1982, an individual by the name John Hopfield of Caltech devised an ANN in which the connections between neurons are asynchronous [58], whereas connections between neurons before his invention were one-directional, also known as feed-forward neural networks (FFNNs). In the same year, a hybrid ANN was developed by Reilly *et al.* [59] which uses multiple learning rules across the network.

The major contribution to the field of artificial intelligence that forms the basis of all

ANNs to this day was published in 1986 by Rumelhart *et al.* [60], in which the authors successfully illustrated how backpropagation can be used to effectively train multi-layer ANNs (see Section 5.3.4.4). One drawback of backpropagating errors in a multi-layer ANN is that it requires a significant amount of computational power. Research into artificial intelligence and ANNs has since been on the rise due to the supply in computational power becoming more easily available.

As of 2023, ANNs are by far the most popular architectures within the context of machine learning and artificial intelligence, and are being successfully employed in a wide variety of real world scenarios with outstanding proficiency. The theoretical foundations of modern ANNs are extensively detailed in the following section.

5.3 Foundations of artificial neural networks

The theory upon which modern ANNs are founded is explained in this section. It is important to note that, while ANNs may be unidirectional *or* bidirectional, the systems proposed in this study utilise *only* unidirectional ANNs, also called feed-forward neural networks (FFNNs). The scope of the theory presented in this section is therefore limited to that of FFNNs.

FFNNs refer to ANNs in which the flow of information is one directional, starting with an input and ending with an output, and may be conceptualised as a directed acyclic graph (DAG) (see Figure 5.1).

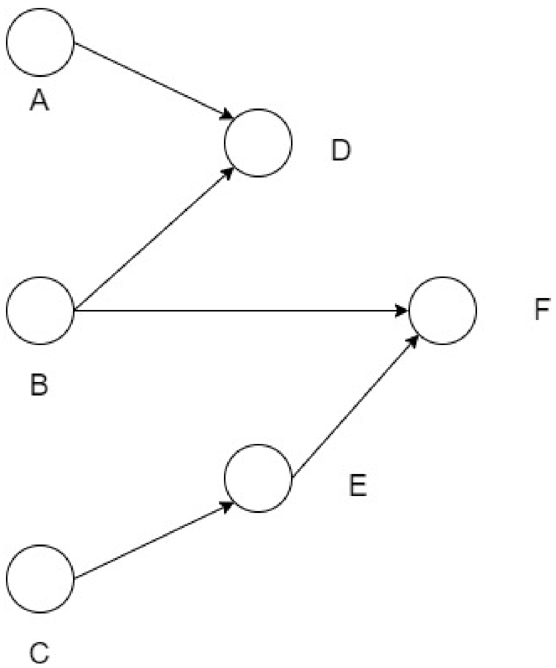


Figure 5.1: An example of a directed acyclic graph.

It is clear from Figure 5.1 that the lines (edges) connecting the nodes (vertices) are *directed* from left to right only. For example, the fact that information flows from node $\mathbb{A}$ to node $\mathbb{D}$ directly implies that information *cannot* flow from node $\mathbb{D}$ to node $\mathbb{A}$. This is the fundamental difference between FFNNs and bidirectional NNs, since the latter may contain cyclic flow of information between nodes (neurons). An example of a bidirectional ANN is a recurrent neural network [61], in which the neurons are known as *memory cells*. FFNNs may therefore be thought of as input-output systems in which input data enters the left side of the network, undergoes some processing and exits the right side of the network.

The fundamental components of FFNNs are introduced in the next sections. The concept of neurons and the connections between them are introduced in Section 5.3.1, while activation functions are detailed in Section 5.3.2. The concept of layers are finally discussed in Section 5.3.3.

5.3.1 Neurons

A neuron in an ANN may be conceptualised as a node in a DAG that holds a value, which is referred to as an activation. The activation of a neuron denotes the importance of the neuron within the context of the entire network. For example, a neuron with a larger activation contributes more to the output produced by the network than a neuron with a smaller activation. The activation a_j of a neuron is determined by the weighted sum of the activations of all neurons that are connected to the neuron in question, plus a bias term, and may be expressed by the following equation,

$$a_j = \sum_{i=0}^{n} w_{ij} a_i + b_i, \tag{5.1}$$

where

- w_{ij} denotes the weight of the connection between neurons i and j,
- a_j and a_i denote the activations of neurons i and j and
- b_i denotes the bias term added after the weighted sum is calculated.

The purpose of the bias term is to either increase or decrease the activation of the neuron in question, and forms part of the parameters of the network which are calibrated during training, and are discussed in detail in Section 5.3.4. It is important to note that Equation 5.1 is *not* applied for certain types of ANN *layers*, which is discussed in Section 5.3.3.

5.3.2 Activation functions

Activation functions form a crucial part of ANNs. Recall that the fundamental operation that a neuron performs inside an ANN constitutes the calculation of a weighted sum of input values received by the neuron, after which an output value is produced (see Equation 5.1). This operation constitutes a simple linear transformation of the input values received by the neuron. It is important to note that, by definition, a composition of linear transformations is also a linear transformation. An ANN is therefore simply a linear transformation of the input data, no matter how many connected neurons it comprises of.

The purpose of activation functions is therefore to transform the output of each neuron in such a way that *non-linearity* is introduced into the network. In so doing, more flexibility in the behaviour of neurons is facilitated, which enables the network to learn non-linear patterns in the input data.

The activation functions employed within the context of this study, that is the ReLU and Softmax activation functions, are detailed in Sections 5.3.2.1 and 5.3.2.2 respectively.

5.3.2.1 The ReLU activation function

The Rectified Linear Unit (ReLU) activation function is defined by the following equation,

$$a_j = \max(0.0, a_i), i = [0, 1, ..., n], \tag{5.2}$$

where a_i denotes the activation of neuron i (see Equation 5.1). The ReLU activation function may be conceptualised by the curve depicted in Figure 5.2.

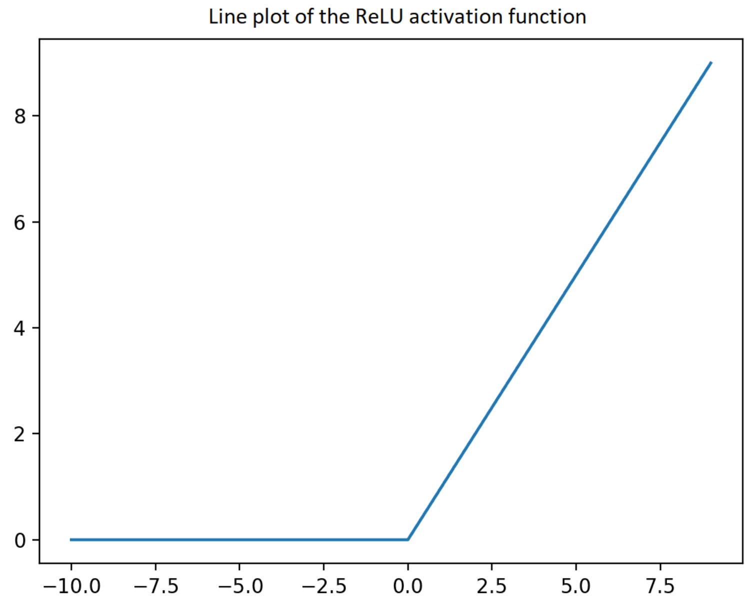


Figure 5.2: A conceptualisation of the ReLU activation function.

The ReLU activation function is based on a standard linear function, where a standard linear function simply maps every input value to itself. It is shown by Goodfellow *et al.* [62] that deep ANNs are easier to train if their behaviour *resembles* that of a composite linear function. In other words, the more complex the employed activation function is, the more difficult it becomes to train and optimise the ANN in question. The ReLU activation function is therefore one of the most commonly used activation functions, since it introduces non-linearity while minimising training complexity. The ReLU activation function is employed within the context of the majority of ANNs proposed in this study.

5.3.2.2 The Softmax activation function

The Softmax activation function, also known as the normalised exponential function, is defined as follows,

$$a_j = \frac{e^{a_i}}{\sum_{i=0}^{n} e^{a_i}}, i = [0, 1, ..., n]. \tag{5.3}$$

The softmax activation function may be conceptualised by the curve depicted in Figure 5.3.

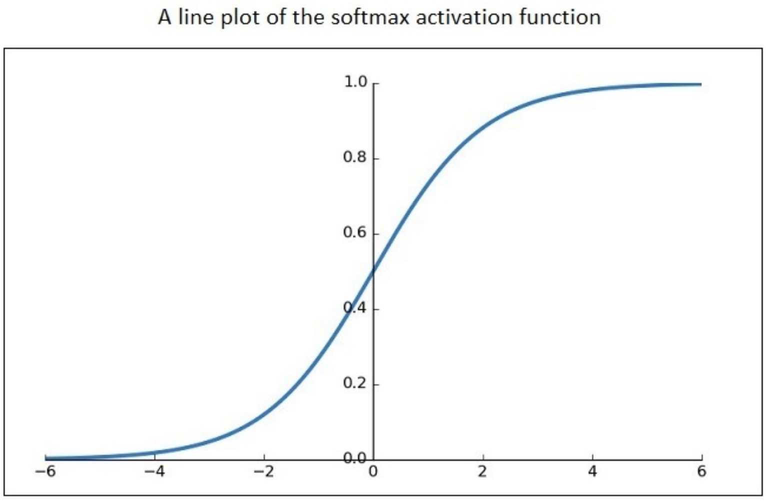


Figure 5.3: A conceptualisation of the softmax activation function.

Due to the normalising effect of the Softmax activation function, the output a_j invariably constitutes a probability, similar to the output of a linear regression model. The Softmax activation function is therefore often used to activate the *final* neurons in an ANN, and by doing so, mapping the output produced by the network to a vector of probabilities that can subsequently be used for classification. These probabilities are also a fundamental component within the context of the training process associated with ANNs (see Section 5.3.4). The Softmax activation function is utilised to activate the final neurons in *all* the networks proposed in this study.

The concept of layers within the context of the ANNs developed in this study are discussed in the following section.

5.3.3 Neural network layers

A layer constitutes a group of neurons which receives input information at the same time and performs a specific operation, for example the calculation of the activations, which produces an output for each neuron within the layer. A single-layer ANN therefore refers to an ANN that comprises of *only* an input layer and an output. An illustration of a single-layer FFNN (perceptron) is depicted in Figure 5.4.

It is important to note that the *same* activation function is *typically* employed by *all* the neurons in a specific layer in order to avoid unnecessary computational complexity. Harmon and Klabjan [63] however reported that, by employing *different* activation functions in specific layers, superior results are achieved when compared to the scenario in which the same activation function is employed for *all* neurons in a specific layer. The networks in this study however employ the *same* activation functions across entire layers.

It is also important to note that activation functions are typically *only* employed within the context of layers in which a weighted sum is calculated for each neuron, for example fully connected layers (see Section 5.3.3.1) and convolutional layers (see Section 5.3.3.2). On the other hand, activation functions are typically *not* employed in layers that do *not* calculate a weighted sum, for example pooling layers (see Section 5.3.3.3) and dropout layers (see Section 5.3.3.4), which renders the use of an activation function redundant.

Modern ANNs often consist of *more* than one layer, and are called multi-layer ANNs.

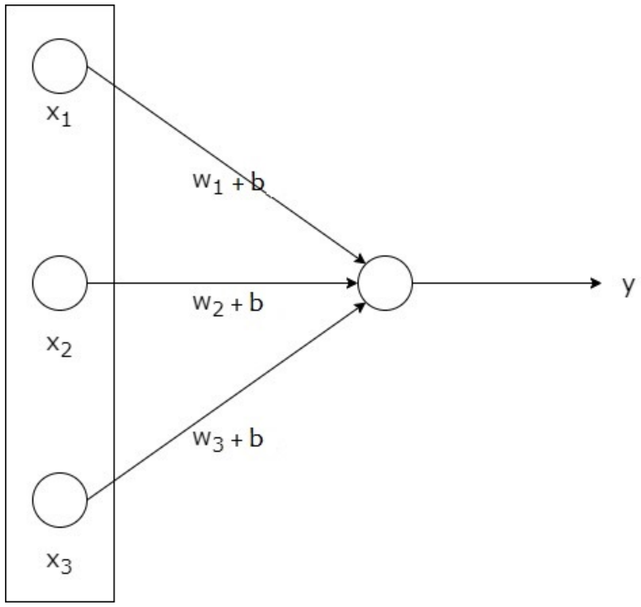


Figure 5.4: An illustration of a single-layer FFNN.

These networks contain one or more so-called *hidden* layers that are located in between the input layer and the output. Hidden layers produce n-dimensional *feature maps*, which are fed directly into the next layer. The purpose of adding hidden layers to a ANN is to obtain high-dimensional feature representations of the input data, and is useful in many scenarios where the input data contains complicated patterns such as objects in colour images. ANNs that comprise of at least one hidden layer are often referred to as *deep* ANNs. An illustration of a deep FFNN which comprises of an input layer, one hidden layer and an output is depicted in Figure 5.5.

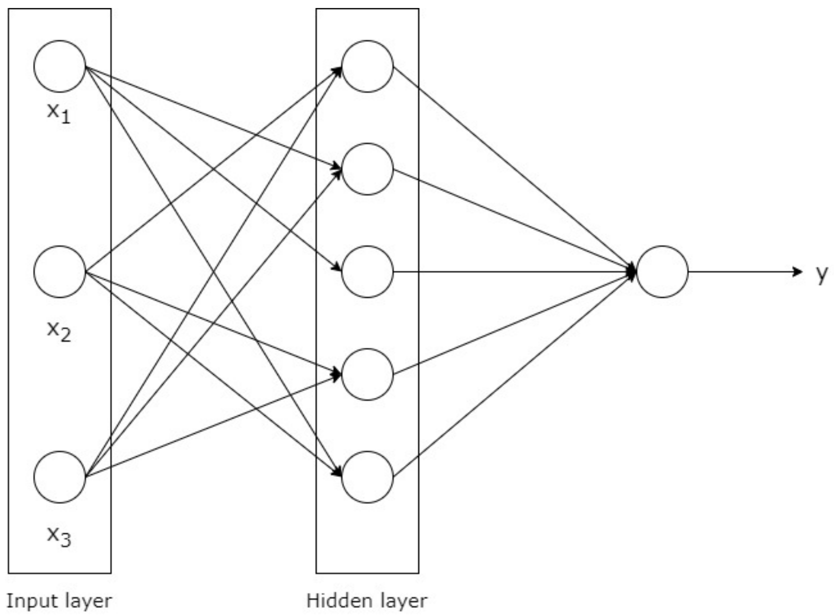


Figure 5.5: An illustration of a multi-layer FFNN.

The connections between neurons in different layers determine the flow of information between layers. It is important to note that, within the context of FFNNs, the *order* of the layers determine which neurons may or may not be connected: the flow of information is *invariably* restricted between *successive* layers. Suppose for example that a certain FFNN comprises of an input layer, three hidden layers and an output. In this example, neurons in the first hidden layer may *only* be connected to neurons in the *second* hidden layer.

A wide variety of different types of layers have been developed in recent years, while only a select few that are employed in this study is discussed in the next few sections.

5.3.3.1 Fully connected layers

In fully connected or "dense" layers, each neuron in one layer is connected to *all* neurons in the *next* layer. A FFNN in which all layers are fully connected is called a fully connected FFNN. Figure 5.6 depicts a fully connected FFNN with one hidden layer.

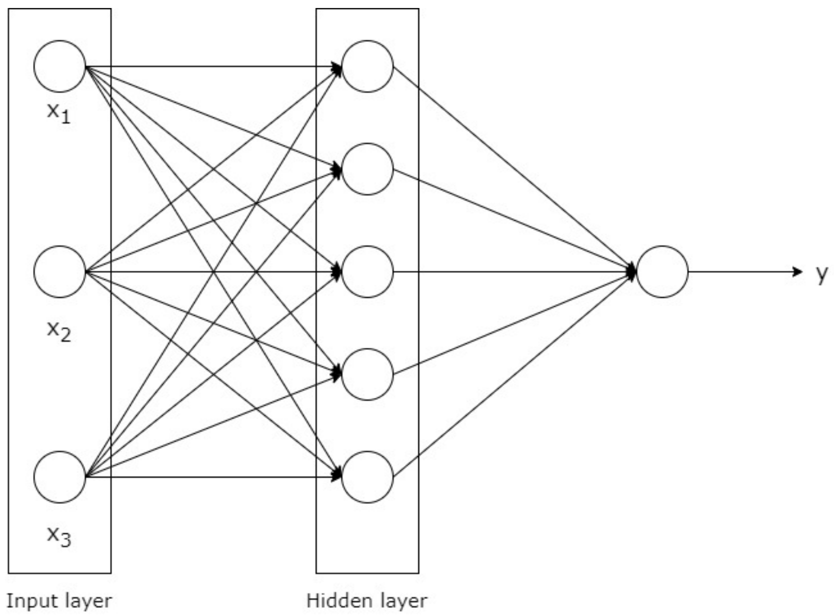


Figure 5.6: An illustration of a fully connected FFNN.

Fully connected layers are often added as the final layers within a FFNN in order to map the high-dimensional feature maps produced by prior layers to a lower-dimensional space for the purpose of regression or classification. Fully connected FFNNs are extremely computationally expensive due to the number of calculations that have to occur to move from one layer to the next, and often leads to an over-estimation of the data distribution without the proper regularisation tools in place.

5.3.3.2 Convolutional layers

The notion on which convolutional layers is based is called a *receptive field.* This notion is derived from the way in which human eyes perceive important information in the environment. The receptive field of a convolutional layer, commonly known as a *kernel,* therefore constitutes a *subset* of neurons inside said layer at a given time. Equation 5.1 is performed on each *subset* of neurons by utilising the corresponding *subset* of weights in order to produce an output. The aforementioned operation is then performed over the entire layer by iteratively centering the kernel on the neurons in the layer and calculating the output as determined by Equation 5.1. Common parameters associated with the kernel include the shape and stride. The shape of the kernel determines the size of the subset of neurons during each iteration, while the stride determines the center of the kernel during each iteration. For example, if the stride is equal to 1, the kernel is iteratively centered on *all* the neurons in the layer, and if the stride is equal to 2, the kernel is iteratively centered on every *second* neuron in the layer, and so forth. Figure 5.7 illustrates the iterative calculations associated with a convolutional layer in which the kernel is of shape 3×1 with a stride equal to 1.

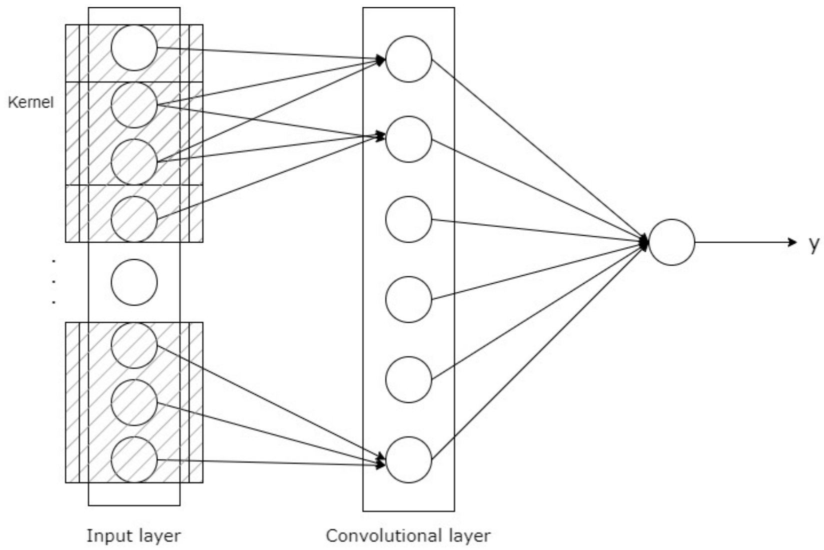


Figure 5.7: An illustration of the operations performed inside a convolutional layer with a kernel of shape 3×1 and stride equal to 1.

The layer upon which the convolutional layer acts may also be *padded* around the boundary in order to ensure that the kernel always fits into the layer when centered on boundary neurons. A suitable kernel shape and stride is typically determined through hyperparameter tuning and a good understanding of the input data. These parameters should be carefully tuned in order to avoid poor network performance. For example, if the kernel shape and stride are both small compared to the size of the layer, the network may become biased on the training set and deliver poor performance on unseen data.

5.3.3.3 Pooling layers

The purpose of pooling layers is to perform subsampling on the output of the preceding layer inside a FFNN, thereby reducing the dimension of the output of the preceding layer before the information is passed to the next layer. Pooling layers have the same parameters as convolutional layers (see Section 5.3.3.2) since the operations in pooling layers are also performed by employing a suitable kernel. Commonly employed pooling operations include max pooling and average pooling. In a max pooling layer, the kernel is iteratively centered on neurons inside the previous layer, after which *only* the maximum value of the neurons in the kernel is retained. Similarly, in an average pooling layer, *only* the average value of the neurons inside the kernel is retained. Figure 5.8 illustrates max pooling with a kernel of shape 3×1 and stride equal to 1.

It is important to note that one iteration of pooling invariably produces a scalar or a single activation, which is how the layer facilitates subsampling for the purpose of dimensionality

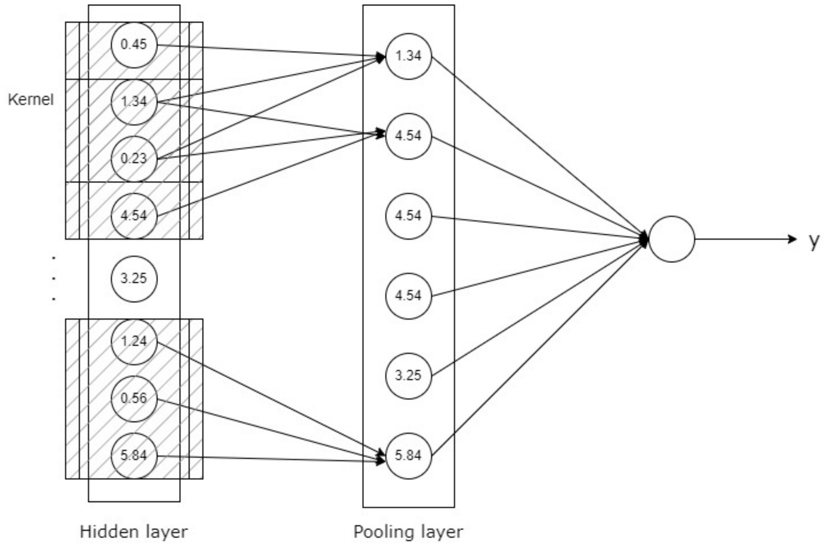


Figure 5.8: An illustration of max pooling with a kernel of shape 3 × 1 and stride equal to 1.

reduction. The pooling layer parameters should also be selected with care, as is the case within the context of convolutional layers. If the kernel is too large for example, valuable information may be discarded which will negatively affect the proficiency of the network.

5.3.3.4 Dropout layers

A dropout layer effectively ignores the contribution of a random subset of neurons across *all* preceding layers, before passing the information on to the next layer. Dropout layers were first proposed by Srivastava *et al.* [18] as a simple method to ensure the network is able to generalise well on unseen data. Figure 5.9 illustrates the effect of applying dropout on a given neural network.

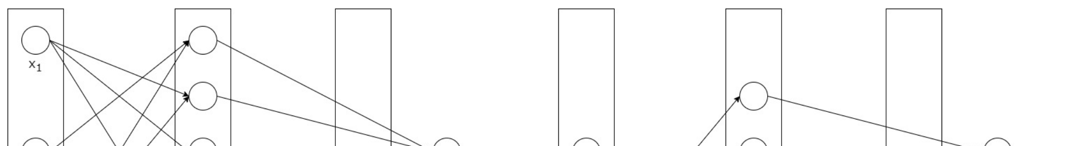


Figure 5.9: (Left) A neural network before dropout is applied. **(Right)** A neural network after dropout is applied.

The most important parameter within the context of a dropout layer constitutes the probability p by which the layer ignores or retains the contribution of a neuron in the preceding layers. An optimal value for p within the context of a specific neural network may be empirically determined by evaluating the performance of the network for a discrete set of probabilities. Srivastava *et al.* [18] however conclude that, based on their experimental results, the optimal p value is typically less than 0.5, which confirms the intuition that discarding the majority of neurons in a network will most likely have a negative effect on the proficiency of the network.

It is important to note that, as stated by Srivastava *et al.* [18], dropout layers are *only* to be active during *training*. The reason for this is that, if dropout layers were to be active during validation and testing as well, the consistency of the network would be adversely affected. In other words, the output produced by such a network for a given test sample would be significantly different over multiple iterations, which is disadvantageous due to a number of reasons, of which the major one constitutes the trustworthiness of the network. It is important to note that, when dropout is applied to a certain neuron during training, the set of weights associated with the remaining active neurons are automatically scaled up by a factor of $s_t = \frac{1}{1-p}$ in order to maintain consistency of the activations. The aforementioned set of weights must therefore be scaled with $\frac{1}{s_t}$ during validation or testing when the dropout layers are inactive in order to maintain consistency of the activations.

5.3.3.5 Batch normalisation layers

Batch normalisation layers were first introduced by Ioffe and Szegedy [17] as a suitable method for accelerating the training time of a deep neural network, as well as to introduce some regularisation into the network. These layers do *not* employ a kernel as is the case within the context of pooling and convolutional layers, and collectively re-scales and re-centers the output of *all* the neurons in the preceding layer by removing their empirical mean and variance for *each* mini-batch during training as follows,

$$X_j = \frac{X_i - \mu_i}{\sigma_i}, i = [0, 1, ..., n], \tag{5.4}$$

where

- X_i denotes the output (activation of all neurons) in layer i,
- μ_i denotes the mean of X_i,
- σ_i denotes the standard deviation of X_i and
- X_j denotes the output of the batch normalisation layer.

In addition to the aforementioned operation, the batch normalisation layer keeps track of the *global* mean and variance for the entire training set, which is then used *directly* during validation and testing in order to re-scale and re-center the output of the neurons in the preceding layer, since no mini-batches are available during validation and testing from which to compute batch-specific statistics.
A conceptualisation of a batch normalisation layer is depicted in Figure 5.10.

Batch normalisation layers have become a common addition to many deep ANNs in recent times due to their proven positive effect on training time and network proficiency. In Section 5.3.4, the reader is introduced to the necessary concepts and mathematical tools employed for the purpose of training ANNs.

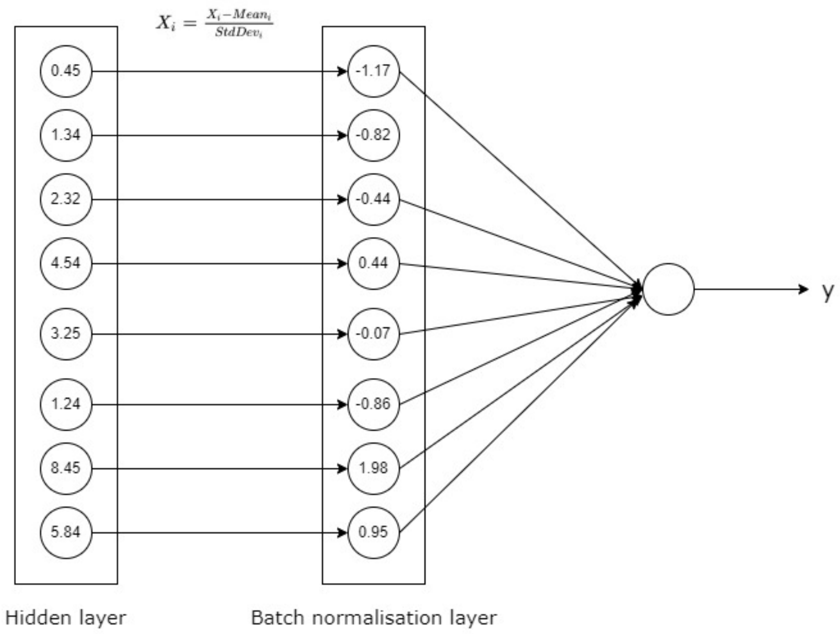


Figure 5.10: A conceptualisation of a batch normalisation layer.

5.3.4 Training artificial neural networks

All the networks proposed in this study are trained in a *supervised* fashion, where input data samples *and* their desired output are provided to the network. The supervised training of artificial neural networks is an iterative process which comprises of a number of components, namely

1. an initialisation step (see Section 5.3.4.1),
2. a forward pass (see Section 5.3.4.2),
3. a cost function (see Section 5.3.4.3),
4. a backward pass (see Section 5.3.4.4) and
5. an optimiser (see Section 5.3.4.5).

For simplicity, the fully connected FFNN depicted in Figure 5.11 with ReLU activation functions is employed for the purpose of explaining one iteration of a training step.

It is important to note that, while the bias terms b_1 and b_2 are typically also vectors that contain different bias *values* associated with each of the neurons in a given layer (see Equation 5.1), the bias terms utilised within the context of the network depicted in Figure 5.11 are set as *constant* values for the purpose of simplifying the calculations.

5.3.4.1 The initialisation step

The first step towards training the network depicted in Figure 5.11 is to initialise the weights associated with the connections between the neurons. The initial set of weights are typically obtained by randomly drawing samples from either a uniform or Gaussian distribution, since there is no algorithm that may aid in distinguishing between a “bad” and a “good” set of

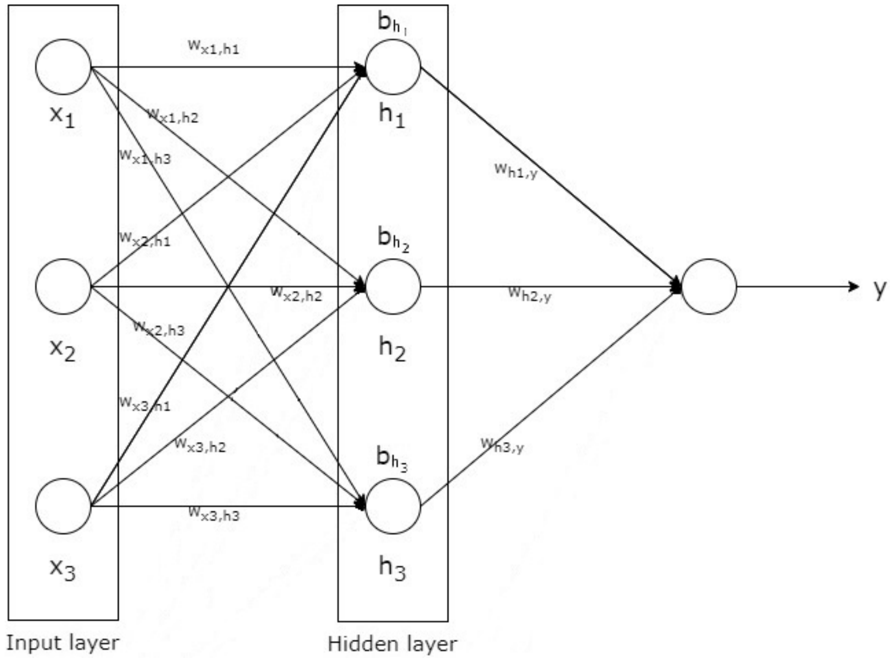


Figure 5.11: A simple fully connected FFNN.

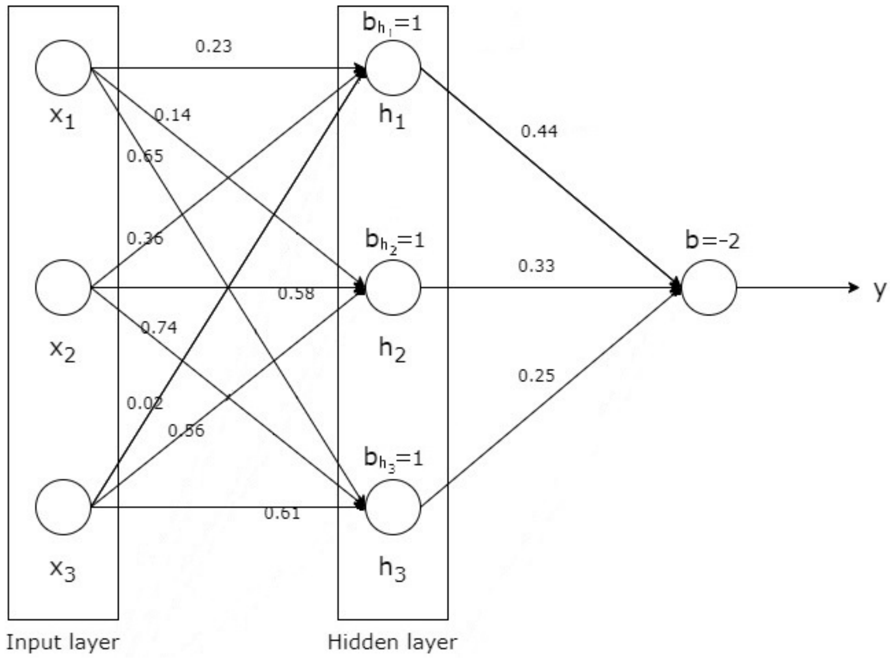


Figure 5.12: The network depicted in Figure 5.11, after being initialised with a Gaussian distribution.

starting weights. The network depicted in Figure 5.11, after being initialised with a Gaussian distribution, is depicted in Figure 5.12.

Due to the random initialisation strategy, some sets of starting weights may cause the network to perform poorly and make it very difficult to train. There have been attempts to address this issue, one of which is called Xavier Initialisation [64], which places an upper bound on the random weight values, after which the weight values are normalised based on the number of incoming and outgoing connections to the layer in question. The networks proposed in this study however address the problem of "bad" initialisation in a different way by simply *re-initialising* the network when poor performance is detected based on certain criteria.

5.3.4.2 The forward pass

The forward pass involves the training samples being fed into the input layer and sequentially transformed by all the hidden layers in order to obtain an output.

Given a training sample $x = [0.51, 2.31, -4.24]$, the activations of the neurons in the hidden layer of the initialised network depicted in Figure 5.12 may be calculated by utilising Equations 5.1 and 5.2:

$$\begin{aligned} h_1 &= \max(0, w_{x_1,h_1}x_1 + w_{x_2,h_1}x_2 + w_{x_3,h_1}x_3 + b_{h_1}) = 1.8641 \\ h_2 &= \max(0, w_{x_1,h_2}x_1 + w_{x_2,h_2}x_2 + w_{x_3,h_2}x_3 + b_{h_2}) = 0.0368 \\ h_3 &= \max(0, w_{x_1,h_3}x_1 + w_{x_2,h_3}x_2 + w_{x_3,h_3}x_3 + b_{h_3}) = 0.4545 \end{aligned} \tag{5.5}$$

The output of the hidden layer may be expressed as another input vector H_1, such that $H_1 = [1.8641, 0.0368, 0.4545]$. H_1 is subsequently utilised to calculate the predicted output $y_{\text{predicted}}$ in the same manner as Equation 5.5, which results in $y_{\text{predicted}} = 0.0$. This concludes the first forward pass within the context of the network depicted in Figure 5.12 and the training sample x. The forward pass is executed for *all* the training samples before *any* of the other steps associated with training a neural network are executed.

5.3.4.3 The cost function

The purpose of the cost function is to sum the difference between the desired output y_{true} and the predicted output $y_{\text{predicted}}$ over *all* training samples. There are a number of different functions that are able to perform the aforementioned task, of which the most common one constitutes the mean squared error (MSE) function:

$$C_x = \frac{1}{N}\sum_{i=0}^{N}(y_{\text{predicted}_i} - y_{\text{true}_i})^2 \tag{5.6}$$

Suppose for example that, within the context of the training sample x defined in Section 5.3.4.2, $y_{\text{true}} = 1.0$. The cost of the network for the training example in question may be calculated with Equation 5.6 as follows,

$$C_x = (0.0 - 1.0)^2 = 1. \tag{5.7}$$

The calculation in Equation 5.7 is repeated over *all* training samples, which produces a cost vector of length K, where K is the number of training samples.

It is important to note that the cost function may be written as a function of all the weights and biases in the network, since these are the parameters which determine the output of the network. The cost function and corresponding cost vector for one training iteration are subsequently utilised during the backward pass in order to determine how the parameters (weights and biases) should be adjusted to *minimise* the total cost over all training samples during the next iteration.

5.3.4.4 The backward pass

Recall that the derivative of any function determines the gradient of the function, while the sign of the derivative determines the direction of the gradient. In order to determine the necessary adjustments to the parameters which lead to the minimum of the cost function, the derivative of said function must be calculated with respect to all the parameters of the network during each training iteration.

It is important to note that the cost function is a *composite* function due to the grouping of the weights and biases in different layers of the network. This renders the direct differentiation of the cost function implausible, and warrants the utilisation of the well known chain rule for differentiation. The chain rule for a composite function is defined as follows using Leibniz' notation,

$$\frac{dy}{dx} = \frac{dy}{du}\frac{du}{dx}, \tag{5.8}$$

where $y = f(x)$ and $u = g(x)$. In order to illustrate the chain rule within the context of the backward pass, the network depicted in Figure 5.12 is simplified to the network shown in Figure 5.13.

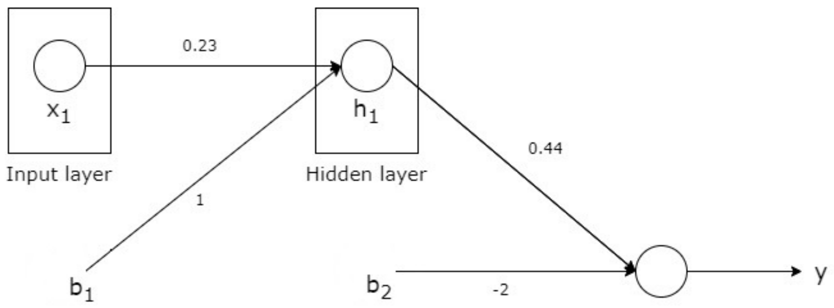


Figure 5.13: A simplified version of the network depicted in Figure 5.12.

The following notation is used to express the chain rule for the network depicted in Figure 5.13:

- C_x: The cost of the network associated with training sample x.
- z_L: The weighted sum as defined in Equation 5.1.
- a_L: The activation of z_L as defined in Equation 5.2.
- a_{L-1}: The activation of the last hidden layer.
- w_L and w_{L-1}: The weights involved in the calculation of a_L and a_{L-1}.
- b_L and b_{L-1}: The biases involved in the calculation of a_L and a_{L-1}.

By employing the notation defined above and Equation 5.1, C_x may be expressed as follows,

$$\begin{aligned} C_x &= (a_L - y_{\text{true}})^2 \\ &= ((\max(z_L, 0.0) - y_{\text{true}})^2 \\ &= ((\max(w_L a_{L-1} + b_L), 0.0) - y_{\text{true}})^2. \end{aligned} \tag{5.9}$$

The derivative of C_x with respect to w_L may therefore be calculated as follows by employing the chain rule,

$$\begin{aligned}\frac{dC_x}{dw_L} &= \frac{dz_L}{dw_L}\frac{da_L}{dz_L}\frac{dC_x}{da_L} \\ &= a_{L-1}\frac{da_L}{dz_L}2(a_L - y_{\text{true}}) \\ &= 2a_{L-1}(a_L - y_{\text{true}}) \text{ if } x > 0 \text{ and } 0 \text{ if } x \le 0.\end{aligned} \tag{5.10}$$

Likewise, the derivative of C_x with respect to b_L can be calculated as follows,

$$\begin{aligned}\frac{dC_x}{db_L} &= \frac{dz_L}{db_L}\frac{da_L}{dz_L}\frac{dC_x}{da_L} \\ &= 1\frac{da_L}{dz_L}2(a_L - y_{\text{true}}) \\ &= 2(a_L - y_{\text{true}}) \text{ if } x > 0 \text{ and } 0 \text{ if } x \le 0.\end{aligned} \tag{5.11}$$

Equations 5.10 and 5.11 express the change in cost within the context of *one* training sample and parameters w_L and b_L. The expression for the total change in cost for *all* of the training samples within the context of w_L and b_L respectively may therefore be expressed as follows,

$$\begin{aligned}\frac{dC}{dw_L} &= \frac{1}{N}\sum_{i=0}^{N}\frac{dC_i}{dw_L} \\ \frac{dC}{db_L} &= \frac{1}{N}\sum_{i=0}^{N}\frac{dC_i}{db_L}.\end{aligned} \tag{5.12}$$

These two equations form two components of the total change in cost for *all* training samples and *all* parameters, denoted by ΔC, which can be expressed as a column vector,

$$\Delta C = \begin{bmatrix} \frac{dC}{dw_0} \\ \frac{dC}{db_0} \\ \frac{dC}{dw_1} \\ \frac{dC}{db_1} \\ \vdots \\ \frac{dC}{dw_L} \\ \frac{dC}{db_L} \end{bmatrix}. \tag{5.13}$$

The quantity $-\Delta C$ therefore denotes the change in parameters w_i and b_i that leads to the largest possible reduction in cost for the next iteration, commonly known as steepest *gradient descent.* It is important to note that Equation 5.13 is only expressed for a fully connected FFNN where all L layers contain *only* one neuron each. In practice however, layers typically contain many neurons with multiple connections to neurons from previous layers. Equation 5.13 should therefore be calculated for *all* of these connections. This constitutes one iteration of the backward pass, also called back propagation, which constitutes the calculation of the change in parameters necessary to minimise the cost function for the next training iteration.

5.3.4.5 The optimiser

In practice, the calculation of Equation 5.13 is extremely computationally expensive, especially for neural networks with thousands of interdependent parameters. The calculation of the "true" $-\Delta C$ value is therefore infeasible in the majority of scenarios. The purpose of the optimiser is to calculate $-\Delta C$ in a *stochastic* manner, which means that the training samples are typically grouped into so-called mini-batches for which $-\Delta C$ is obtained, by which the parameters are *incrementally* updated after $-\Delta C$ is obtained for each mini-batch. The so-called stochastic gradient descent (SGD) algorithm is therefore utilised for the purpose of calculating the steepest gradient descent of the cost function within the context of each mini-batch, which provides a drastic reduction in computational complexity and training time, while still being able to minimise the cost function across all training samples.

The difference between a "true" and stochastic gradient descent step may be conceptualised by the analogy of a ball rolling down a hill versus a drunk man stumbling down the same hill. The ball will invariably find the fastest way to the bottom of the hill (the minimum of the cost landscape), whereas the assumption is that the drunk man should "eventually" arrive at the bottom of the hill.

It is important to note that, in practice, the cost landscape typically comprises of a number of so-called *local* minima, and depending on the starting position of the optimiser, there is no guarantee that the obtained minimum is in fact the global minimum. There are however ways to increase the probability of reaching the global minimum as opposed to getting stuck in local optima. One of these ways constitutes the introduction of a so-called *momentum* term into the SGD algorithm, which essentially assigns more prominence to steeper negative gradients, and might assist by simply "stepping over" the local optima. The principle of SGD with momentum is employed for the purpose of training *all* the networks developed in this study.

The two most important parameters that need to be calibrated within the context of SGD constitute (1) the learning rate and (2) the size of the mini-batches. The learning rate controls the magnitude of the adjustments made to the weights after the loss has been back propagated for a given mini-batch. In other words, it defines the size of the "step" towards a minimum of the cost function. The larger the step size, the greater the risk of over-stepping the minimum of the cost function, or in other words successively "landing" on opposite sides of the hill. The smaller the step size, the more computationally expensive the algorithm becomes, and the greater the risk of ending up in local minima. The size of the mini-batches on the other hand determines the accuracy of the SGD algorithm within the context of all the training data. The smaller the batch size, the more erratic the steps toward a minimum becomes and the greater the risk of not reaching any minimum at all. Larger batch sizes increase computational complexity, but is typically a safer choice within the context of reaching a minimum. The process of utilising SGD to obtain a minimum of the cost function is commonly known as *convergence*. If the network is unable to reach a minimum, it is said that the network diverges.

This concludes the relevant theory with regards to the training process associated with neural networks. The main type of neural network architecture employed in this study, namely convolutional neural networks (CNNs), is detailed in the following section.

5.4 Convolutional neural networks

Convolutional neural networks (CNNs) are networks that primarily employ convolutional layers (see Section 5.3.3.2) for the purpose of processing the input data, and are therefore a very suitable choice within the context of image processing and classification. Recall that convolutional layers employ a kernel which is related to the receptive field within the context of the human eye, and are therefore particularly efficient for the purpose of learning distinct visual features in images.

It has been shown that CNNs are able to achieve state-of-the-art (SOTA) level proficiency in many popular image classification tasks, and is therefore a suitable neural network architecture within the context of hand vein-based biometric authentication. Due to the increasing availability of computational power, there is virtually no limit to the number of layers and parameters that a CNN may comprise of. The number of layers and parameters mainly depend on the extent of the detail in the images to be classified. In other words, the more complicated the detail in the input images, the more layers and parameters are typically required in order to achieve sufficient proficiency. The images to be classified within the context of this study (see Chapter 3) however do *not* contain highly complicated patterns when compared to colour images of, say, animals, machinery or vehicles. The CNNs developed in this study therefore comprise of only a small number of layers when compared to CNNs such as ResNet [65] or AlexNet [66].

The base architecture of the CNNs developed in this study is depicted in Figure 5.14.

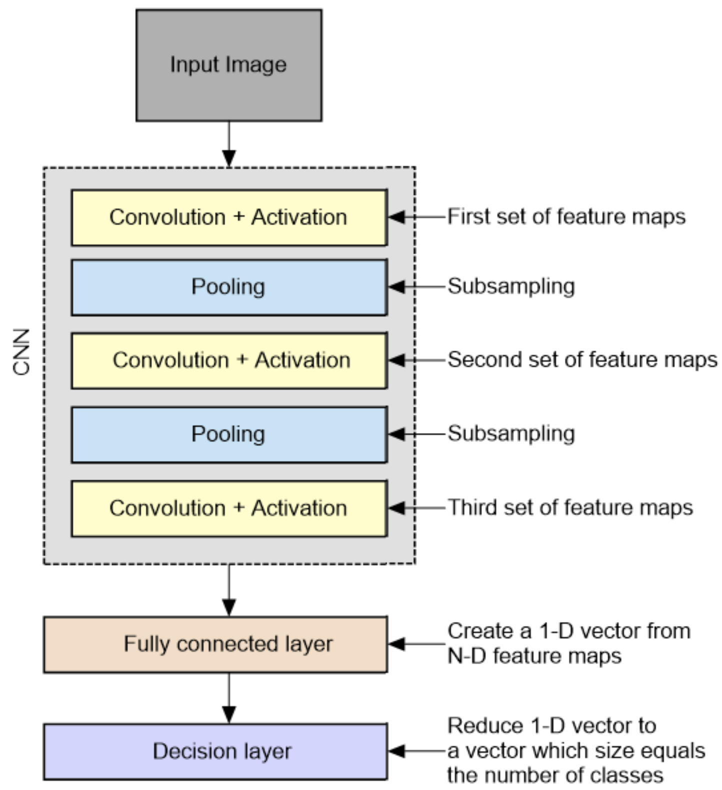


Figure 5.14: The base architecture of the CNNs developed in this study.

It is important to note that a pooling layer (see Section 5.3.3.3) is employed in between the convolutional layers for the purpose of selecting the most prominent activations, while two fully connected layers (see Section 5.3.3.1) are added at the end of the network in order to reduce the dimensions of the convolutional feature maps to a vector that is used for classification. It is also important to note that a ReLU activation function (see Section 5.3.2.1) is employed after each convolutional layer, as well as after the second-to-last fully connected layer for the purpose of truncating negative activations to zero and to introduce non-linearity into the network. A softmax activation function (see Section 5.3.2.2) is employed after the final fully connected layer for the purpose of obtaining a vector of probabilities, each of which indicate the probability that a given data sample belongs to a particular class (authentic or imposter).

Although CNNs are mainly used for image *classification*, the main problem addressed in this study is that of image *verification*. Within the context of image *verification*, a *comparison* needs to occur between two images: an *authentic* image and a *questioned* image. The network is therefore supposed to compare the two images and outputs a probability that denotes whether or not the questioned image contains the same structure as the one contained in the authentic image.

A specific class of CNNs, called similarity measure networks (SMNs), are therefore utilised within the context of this study for the aforementioned purpose, and is detailed in the following section.

5.5 Similarity measure networks

The purpose of similarity measure networks (SMNs) is to utilise the basic architecture of CNNs in order to compare two images: the reference (authentic) image and the questioned image, after which an output is produced that denotes the probability that the questioned image belongs to the same class as the authentic image. The final fully connected (decision) layer of a SMN therefore typically comprises of two classes: the so-called positive (authentic) class and the so-called negative (imposter) class.

The main difference between an SMN and a standard CNN is the way in which the input data is fed into the network and the way the output is produced. Recall that the purpose of an SMN is to obtain a feature map that represents the *difference* between an authentic and questioned image. An SMN therefore needs to be structured in such a way that it is able to compare an authentic and questioned image.

The concept of utilising CNNs to compare two images was first introduced by Zagoruyko and Komodakis [16]. It is important to note that Zagoruyko and Komodakis [16] proposed that SMNs be used to iteratively compare corresponding *patches* in the two input images, after which a decision is made by considering the probabilities associated with the comparison between *all* the image patches. The SMNs proposed in this study do however *not* consider image patches; the two images are instead fed into the networks in their entirety, since it is desired to compare the entire hand vein structures between the authentic and questioned images. The two SMN architectures employed in this study are called two-channel (2CH) networks (see Section 5.5.1) and Siamese networks (see Section 5.5.2).

5.5.1 Two-channel networks

Two-channel (2CH) networks derive their name from the way the authentic and questioned images are fed into the network. The authentic and questioned images are *stacked* in a depth-wise fashion in order to produce a two-channel image, where the top channel contains the authentic image, while the bottom channel contains the questioned image. The aforementioned stacking protocol is illustrated in Figure 5.15 within the context of two grey-scale images, where a green square indicates an authentic sample and a red square indicates an imposter sample.

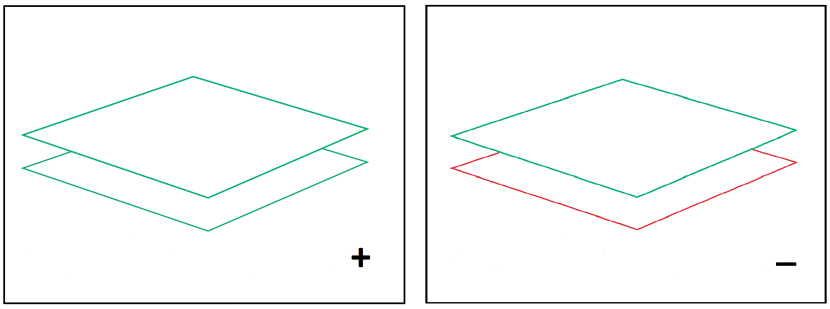


Figure 5.15: An illustration of the stacking protocol within the context of 2CH SMNs. A green square indicates an authentic sample and a red square indicates an imposter sample.

It is important to note that the authentic and questioned images are not *explicitly* compared within the context of a two-channel network. The motivation behind the utilisation of a 2CH network for the purpose of comparing two images rather lies in the fact that, since CNNs are particularly well-suited for the purpose of learning high-dimensional feature representations of images, the difference between the channels in the stacked two-channel images may be learned *implicitly*. In other words, the feature representations contain in themselves an implicit difference measurement between the two images. The architecture of a typical 2CH network is depicted in Figure 5.16.

Siamese networks, an SMN architecture proposed by Zagoruyko and Komodakis [16] that is able to calculate an *explicit* similarity measure between the authentic and questioned images, are detailed in the following section.

5.5.2 Siamese networks

Siamese networks are different from 2CH networks in the sense that the reference and questioned image are each fed into a separate *branch* of the network. A branch within the context of Siamese networks constitutes a typical CNN with its own parameters that is independent of the other branch. The motivation behind the design of a Siamese network is that each CNN branch produces a feature embedding of the corresponding class, that is the authentic or imposter class, after which the *explicit* difference between the feature embeddings for a given pair of authentic and questioned images is calculated by means of a suitable dissimilarity measure function, such as the squared Euclidean distance. The basic architecture of a Siamese network is depicted in Figure 5.17.

It is important to note that the high-dimensional feature embeddings obtained from each CNN branch in the Siamese network is *first* reduced to a one-dimensional vector by employing a fully connected layer. The explicit difference vector between these two one-dimensional vectors is subsequently calculated and fed into the final decision layer which outputs the probabilities of the questioned image belonging to the authentic and imposter classes.

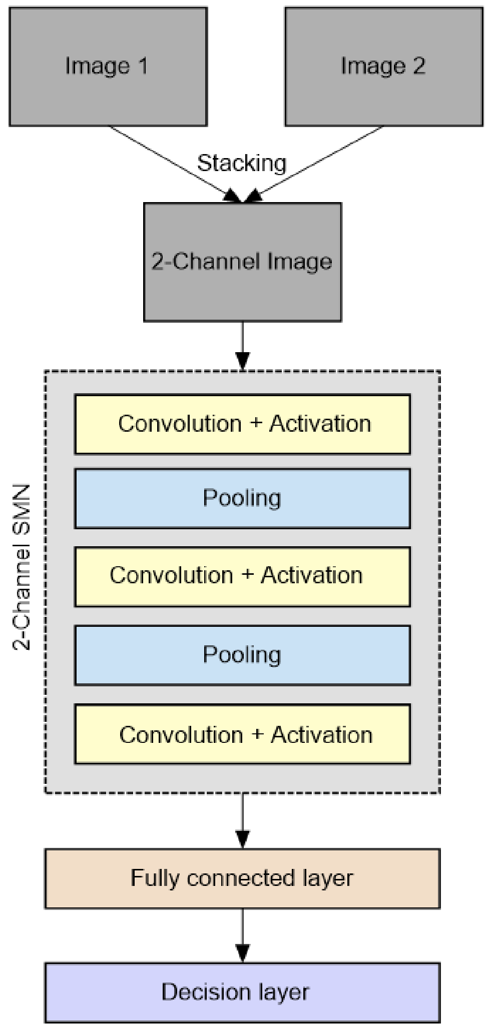


Figure 5.16: The basic design of a 2CH SMN.

5.6 Concluding remarks

A brief history of artificial neural networks was provided in this chapter, along with the fundamental theory associated with the network components and the training process. The reader was also introduced to the specific neural network architectures employed in this study, that is two-channel (2CH) networks and Siamese networks, and how these networks are employed within the context of this study for the purpose of hand vein-based biometric authentication. The hand vein-based biometric authentication systems developed in this study are detailed in the next chapter.

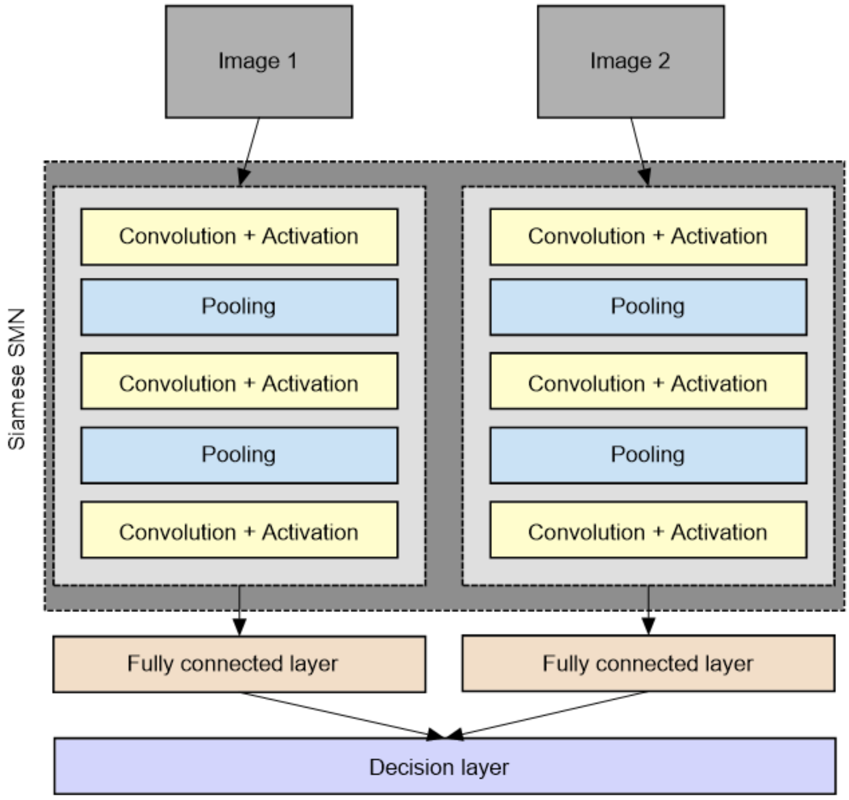


Figure 5.17: The basic design of a Siamese SMN.

Chapter 6

System design

6.1 Introduction

The feature extraction, feature matching and classification components of the hand vein-based biometric authentication systems developed in this study are detailed in this chapter. Each of these systems comprises of

1. an image preprocessing protocol (see Section 3.3),
2. a neural network architecture (see Section 5.4),
3. a convolutional neural network-based feature extractor (CNN-based FE) (see Section 6.2),
4. a classifier (see Section 6.3) and
5. a threshold selection criterion (see Section 7.6.2).

The proposed system design involves gauging the proficiency of *all* 96 possible combinations of the aforementioned system, after which the obtained proficiencies are *ranked* in a descending fashion. The *top performing* system is subsequently selected, which comprises of

1. a *specific* image preprocessing protocol,
2. a *specific* neural network architecture,
3. a *specific* CNN-based FE,
4. a *specific* classifier and
5. a *specific* threshold selection criterion.

To the best of the author's knowledge, the process of experimentally selecting the top performing *system* constitutes a *novel* protocol proposed in this study within the context of dorsal hand vein-based authentication. A similar protocol has been proposed by [35], which comprises of *only* selecting the top performing CNN-based FE and classifier combination, while employing *only* a *single* neural network architecture and a *single* preprocessing protocol. The protocol proposed by [35] also does *not* experiment with different configurations of neural network *layers*, but rather evaluates system proficiency by changing the *depth* of the *existing* neural network design. This is accomplished by considering a number of so-called

"CNN *blocks*", which comprises of a *specific* CNN design. These CNN blocks are iteratively *stacked* in a depth-wise fashion in order to iteratively increase the depth of the network. A novel Bayesian optimisation approach is proposed for the purpose of determining the optimal number of CNN *blocks* for the CNN-based FE. The protocol proposed in this study therefore constitutes a significant contribution to the current literature in the sense that it addresses the problem of *system* design in an *empirical* fashion within the context of hand vein-based authentication systems. The devastating effects on system proficiency caused by employing a sub-optimal *system* design are shown in Chapter 8, together with the mitigation thereof by employing the proposed protocol.

This chapter details the employed CNN-based FEs (see Section 6.2) and classifiers (see Section 6.3), since the employed image preprocessing protocols are presented in Section 3.3, the employed neural network architectures are presented in Section 5.5 and the employed threshold selection criteria are presented in Section 7.6.2. It is important to note that the feature matching protocols are inherent to the neural network architectures (2CH and Siamese SMNs) which are detailed in Section 5.5.

6.2 Convolutional neural network-based feature extractors

A convolutional neural network-based feature extractor (CNN-based FE) refers to the CNN component of the systems proposed in this study. For example, the CNN-based FE component within the context of a 2CH network is outlined in red on the left of Figure 6.1, while the CNN-based FE component within the context of a Siamese network is outlined in red on the right of Figure 6.1 for the purpose of illustration.

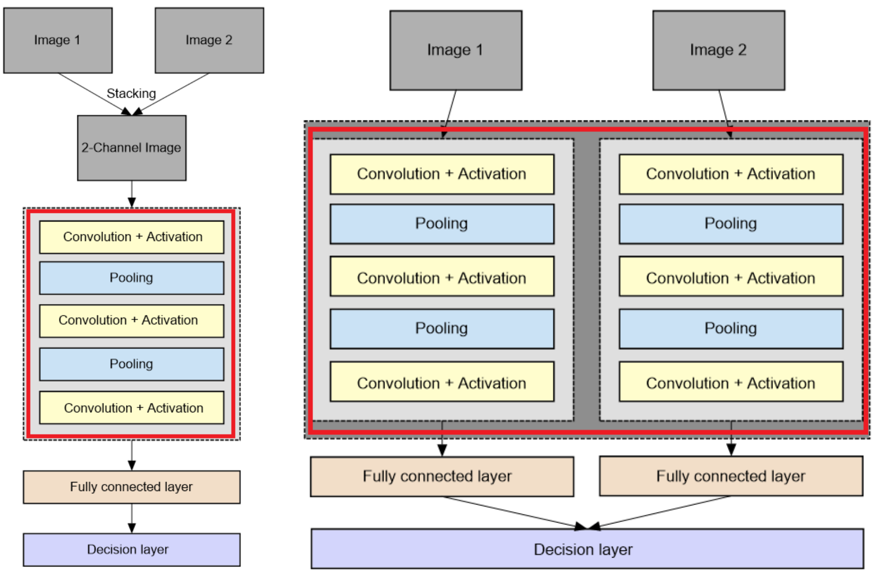


Figure 6.1: (Left) The CNN-based FE component within the context of a 2CH network outlined in red. **(Right)** The CNN-based FE component within the context of a Siamese network outlined in red.

A total of four different CNN-based FEs are developed in this study, and are henceforth referred to as (1) the standard variation (see Section 6.2.1), (2) the batch normalisation variation (see Section 6.2.2), (3) the dropout variation (see Section 6.2.3) and (4) the batch normalisation *and* dropout variation (see Section 6.2.4).

It is important to note that the experimental protocols developed in this study within the context of dorsal hand vein-based authentication (see Chapter 7) are *novel* and that no benchmark results exist yet for any dorsal hand vein-based authentication system within the context of these specific experimental scenarios. It was therefore decided to gauge the proficiency of each of the four aforementioned variations *separately* in order to establish whether or not certain variations consistently outperform the rest within the context of the experimental protocols proposed in this study, and draw possible conclusions as to why this may be so.

6.2.1 The standard variation

The so-called "standard" variation comprises of *only* convolutional and pooling layers, which are the fundamental layers employed within the context of the majority of CNNs proposed in recent literature. This variation therefore constitutes the *benchmark* variation against which the other variations proposed in this study are evaluated. The standard variation is depicted in Figure 6.2, while the transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

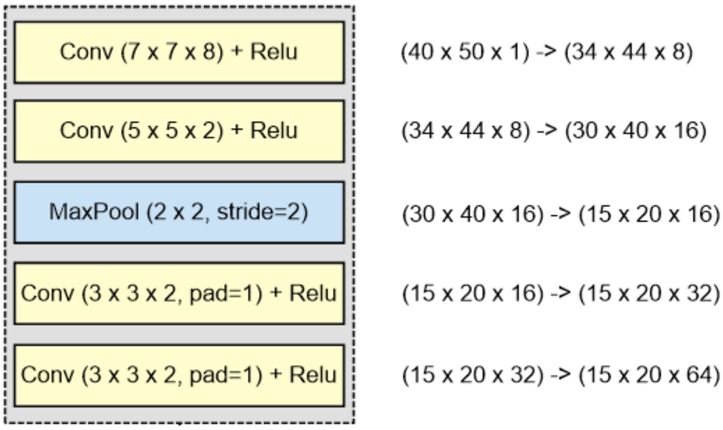


Figure 6.2: A conceptualisation of the proposed standard variation. The transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

It is important to note that the dimensions of the input images are $40 \times 50 \times 1$ within the context of the Siamese networks and $40 \times 50 \times 2$ within the context of the two-channel networks. The dimensions of each layer in *all* four variations are identical, and are a result of experimentally calibrating the parameters of each layer such as the kernel size, padding and stride, while also considering the complexity of detail contained within the input images. Said experimental calibration forms part of the hyperparameter tuning protocol detailed in Section 7.6.1.

6.2.2 The batch normalisation variation

The so-called "batch normalisation" variation comprises of *all* the layers contained in the standard variation, together with two additional batch normalisation layers (see Section 5.3.3.5). The batch normalisation variation is depicted in Figure 6.3, while the transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

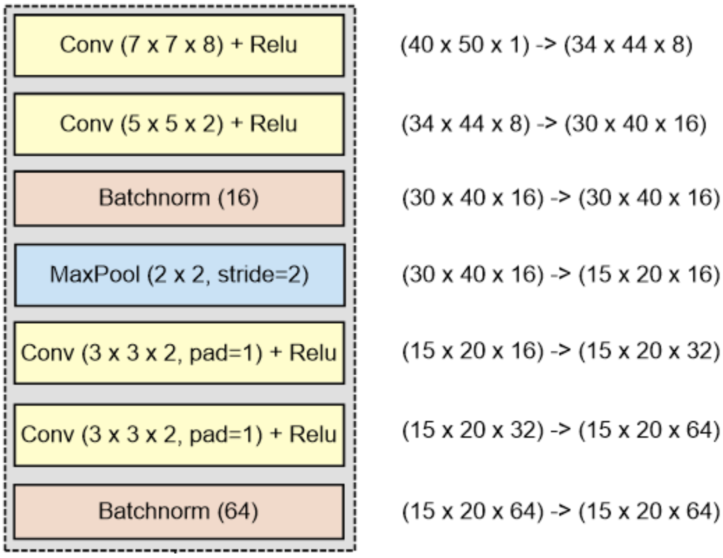


Figure 6.3: A conceptualisation of the proposed batch normalisation variation. The transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

It is important to note that the batch normalisation layers are situated *after* the convolutional layers for the purpose of regularising the convolutional feature embeddings *before* max pooling is applied. Additionally, it is important to note that batch normalisation layers have *no* effect on the dimensions of the feature maps.

6.2.3 The dropout variation

The so-called "dropout" variation comprises of *all* the layers contained in the standard variation, together with two additional dropout layers (see Section 5.3.3.4). The dropout variation is depicted in Figure 6.4, while the transformations of the dimensions of the input image as it passes through said variation are depicted on the right. The probability of dropout is fixed at 0.2 in order to determine whether the addition of minor dropout may improve system proficiency, since experimental results indicate no evidence of overfitting without the addition of dropout. It is however considered standard practice to consider the dropout rate as a hyperparameter of the network that needs to be optimised. This constitutes a valid avenue for future work within the context of this study.

It is important to note that the dropout layers are situated *after* the convolutional layers, as is the case within the context of the batch normalisation variation. Dropout layers ultimately aim to achieve the same effect as batch normalisation layers, that is regularisation, but is fundamentally different to batch normalisation layers in the sense that these layers act on the network *structure*, as opposed to batch normalisation layers which act *directly* on the output of the preceding layer. Additionally, it is important to note that batch normalisation layers also have *no* effect on the dimensions of the feature maps, as is the case within the context of the batch normalisation variation.

6.2.4 The batch normalisation and dropout variation

The so-called "batch normalisation and dropout" variation comprises of *all* the layers contained in the standard variation, together with two additional batch normalisation layers

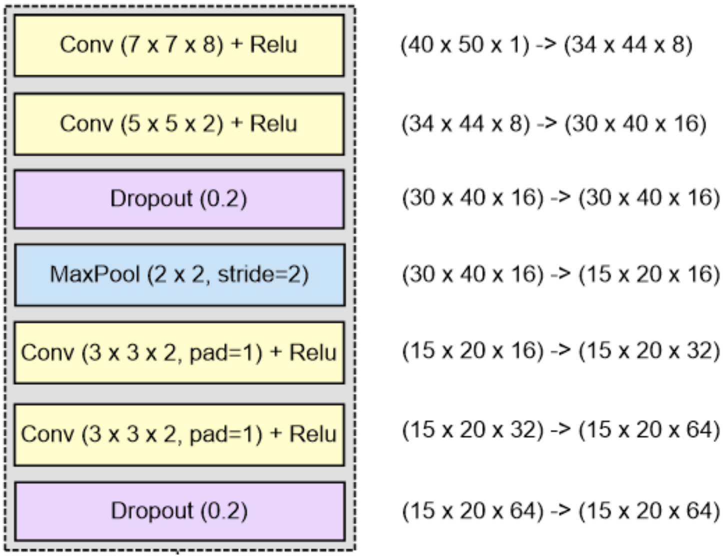


Figure 6.4: A conceptualisation of the proposed dropout variation. The transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

(see Section 5.3.3.5) *and* two additional dropout layers (see Section 5.3.3.4). The batch normalisation and dropout variation is depicted in Figure 6.5, while the transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

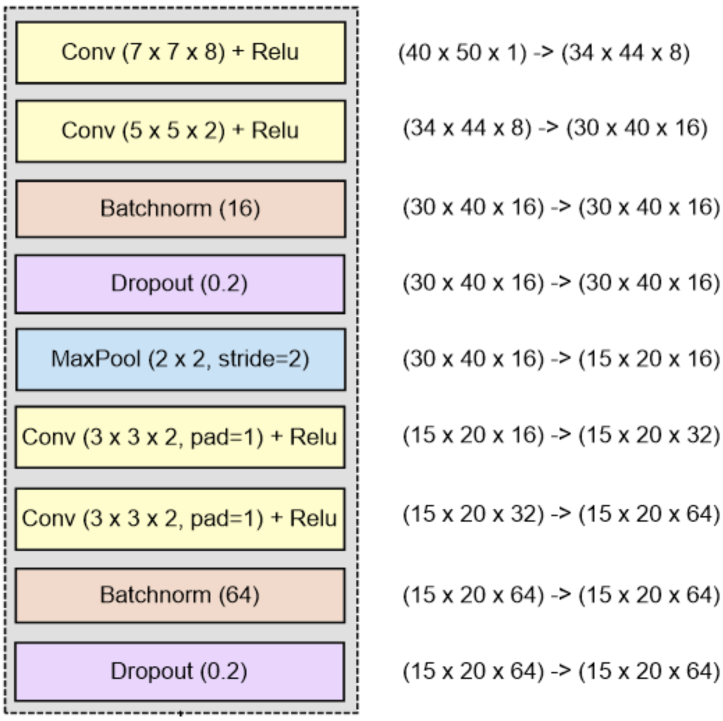


Figure 6.5: A conceptualisation of the proposed batch normalisation and dropout CNN-based FE. The transformations of the dimensions of the input image as it passes through said variation are depicted on the right.

The use of batch normalisation *and* dropout layers within a single CNN is typically considered redundant, since the two types of layers ultimately achieve the same effect. Garbin *et al.* [19] however reported that batch normalisation layers may be used in conjunction with dropout layers in order to improve system proficiency, which motivates the utilisation thereof within the context of this study.

The classifier employed within the context of this study are detailed in the next section.

6.3 Classifiers

Recall that the fundamental problem addressed in this study is that of image *verification*, rather than *classification*. The classification component in the proposed systems is therefore referred to as a *verifier*, that is a binary classifier, in which the two classes respectively represent the authentic and questioned hand vein images. The final layer in all the networks developed in this study is therefore designed in such a way that it produces a 2×1 probability vector which contains the predicted probabilities that a questioned sample belongs to either the client or an imposter. This is accomplished by applying the softmax activation function to the final layer of the networks.

The verifier employed within the context of this study, which is a probabilistic threshold-based verifier referred to as the "softmax" verifier, is detailed in the the following section.

6.3.1 Softmax verifier

The so-called softmax verifier comprises of the application of a probabilistic threshold value to the output of the networks proposed in this study. Said value determines whether a questioned sample is accepted as authentic or rejected. The optimal probabilistic threshold is experimentally determined during the *validation* stage of the training process (see Section 7.6.2).

A 2CH network which comprises of the standard variation is depicted in Figure 6.6, in which the softmax verifier is shown in green.

6.4 Concluding remarks

The CNN-based FEs and verifiers employed within the context of the proposed hand vein-based authentication systems were presented in this chapter. The experimental protocols developed for the purpose of gauging the proficiency of the proposed systems are discussed in the following chapter.

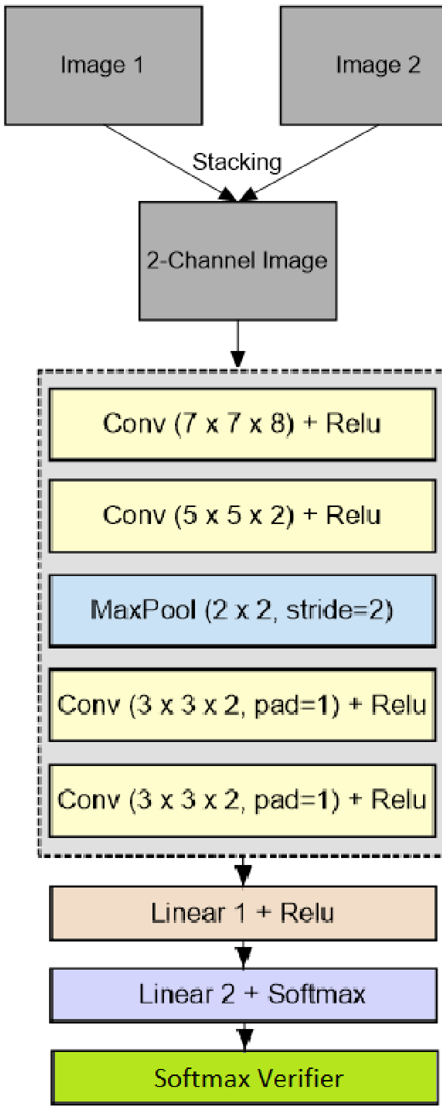


Figure 6.6: A 2CH network which comprises of the standard variation, in which the softmax verifier is shown in green.

Chapter 7

Experimental protocols

7.1 Introduction

The experimental protocols developed for the purpose of gauging the proficiency of the systems proposed in this study are detailed in this chapter. Two experimental scenarios are developed for the purpose of simulating real world scenarios. The first scenario is henceforth referred to as the individual *dependent* scenario (IDS), while the second scenario is henceforth referred to as the individual *independent* scenario (IIS).

The individual *dependent* scenario (IDS) constitutes a simulation of the real-world scenario in which a *tailor-made* network is trained for *each* client enrolled into the system. This approach is motivated by the fact that the enrolment of each client (1) is independent of other clients in the system and (2) requires a minimal amount of data samples from the client. In addition to the aforementioned benefits, it is shown in Section B.2 that a tailor-made network may be operational in mere minutes after the data samples have been acquired. The independence between the networks associated with different clients is also advantageous within the context of system outages, since it is highly unlikely that potentially millions of networks will have problems at the same time. Possible drawbacks of this approach include (1) the fact that the storage cost of potentially millions of tailor-made networks could be very high and (2) the administrative overhead associated with managing and hosting these networks could be excessive. The experiment associated with the IDS is henceforth referred to as the "individual dependent experiment" (IDE).

The individual *independent* scenario (IIS) constitutes a simulation of the real-world scenario in which a *single* network is trained on a set of so-called *holdout* individuals *prior* to the enrolment of *any* clients into the system. This network is trained in a *once-off* fashion, and may be utilised for the purpose of authenticating a questioned sample against a single so-called *reference* sample belonging to the client in question. The benefits of this approach include (1) the fact that *only one* reference sample is required from the client during enrolment and (2) the storage costs and administrative overhead associated with managing only one network is insignificant when compared to the storage cost and overhead associated with potentially millions of networks within the context of the IDS. A drawback of this approach constitutes the fact that a number of hand vein samples is required from a *sufficient* number of *different* individuals in order to train the network. This issue is however addressed in this study by means of utilising the proposed GenVeins database that contains hand vein samples from a *sufficient* number of *fictitious* individuals (see Chapter 4) for the purpose of training the network in question. It is shown in Chapter 8 that outstanding system proficiency may be achieved by training a single network on the hand veins associated with a sufficient number of *fictitious* individuals, while testing on hand vein samples associated with *actual* individuals. A total of three experiments are conducted within the context of the IIS, and are outlined below.

1. The proficiency of the proposed systems are first gauged *only* within the context of the available *actual* individuals from the Bosphorus and Wilches databases. The purpose of this experiment is to obtain benchmark results which quantify the extent of the problem of attempting to train a general hand vein-based similarity measure network on a non-representative set of individuals. This experiment is henceforth referred to as the "benchmark experiment" (BME).

2. The proficiency of the proposed systems are subsequently gauged by training and validating with images from the GenVeins database, while testing on the available *actual* individuals from the Bosphorus and Wilches databases. This experiment is henceforth referred to as the "first GenVeins experiment" (FGE), of which the purpose is to determine the proficiency gained over the benchmark experiment.

3. The final experiment, which is henceforth referred to as the "second GenVeins experiment" (SGE), comprises of the same experimental protocol as the first GenVeins experiment, but each *actual* individual is instead granted a total of *three* attempts for authentication during testing. The purpose of this experiment is to determine the proficiency gained by discarding low-quality test samples.

It is important to note that a single experiment within the aforementioned context refers to the training, validation and testing of *all* 96 systems proposed in this study across all nine cross-validation folds.

The specific cross-validation protocol developed in this study for the purpose of obtaining a robust set of statistical performance measures for each experiment is detailed in Section 7.2. The proposed data augmentation strategies are detailed in Section 7.3, while the proposed data partitioning protocols are discussed in Sections 7.4 and 7.5 respectively. The proposed training, validation and testing protocols developed in order to gauge the proficiency of the proposed systems within the context of the IDS and IIS are finally discussed in Section 7.6.

It is important to note that the data augmentation strategies detailed in Section 7.3 are *only* applicable within the context of the IDS, and that *no* augmentation is applied within the context of the IIS.

7.2 Cross-validation

Cross-validation plays a fundamental role in gauging the proficiency of a machine learning system in an unbiased fashion. The main purpose of a cross-validation protocol is to obtain a *set* of statistical performance measures of the system in question on a number of *different* data partitions, as opposed to only obtaining a *single* statistical performance measure on a *single* data partition. The main problem with employing only a *single* data partition constitutes the fact that the samples in said partition may not be representative of the entire population of samples, which invariably results in a *biased* statistical performance measure. Another drawback, which is a result of the aforementioned drawback, constitutes the fact that two different machine learning systems *cannot* be statistically compared when *only* a single statistical performance measure is obtained for each of the two machine learning systems by employing two *different* data partitions.

Beleites *et al.* [67] showed that it is crucial to employ a cross-validation protocol for the purpose of variance reduction when gauging the proficiency of machine learning systems in the case where only a limited number of data samples are available. Beleites *et al.* [67] suggested that either a so-called bootstrap resampling [68] or a k-fold cross-validation protocol [69] be employed in order to obtain a set of statistical performance measures of the system in question when trained and tested with a number of different data partitions. This reduces the probability of bias when compared to the case in which *no* cross-validation protocol is employed.

Dietterich [69] conducted an analysis of a number of different cross-validation protocols by which two different machine learning systems may be compared, of which the most popular one employed within the context of machine learning, as reported by Dietterich [69], is referred to as the "resampled paired t-test". For brevity, the resampled paired t-test is henceforth referred to as "repeated random sampling" (RRS). RRS comprises of *randomly* partitioning the available data into training, validation and test sets over n iterations. The machine learning system in question is then trained, validated and tested for all iterations, which produces a distribution of statistical performance measures. This distribution constitutes a robust and unbiased measurement of the proficiency of the system in question by minimising the probability of bias associated with a *single* data partition. The standard deviation can subsequently be computed for said distribution, which adds insight into the stability of the system in question over a number of different, randomly selected training, validation and test sets. RRS additionally enables the statistical comparison between two systems, since the distribution of statistical performance measures is *representative* of the proficiency of the proposed systems on the *entire* population of samples.

Dietterich [69] however emphasised an important issue when comparing two distributions of statistical performance measures obtained by employing RRS, which constitutes the fact that the *test* sets between the two distributions are *neither* paired *nor* independent. This is a result of repeatedly sampling from the *same* population during each iteration for each of the systems. This may influence the validity of a statistical comparison drawn by either employing a paired samples t-test [69] or an independent samples t-test [70], since the former assumes that the test sets are *identical* during each iteration, while the latter assumes that the test sets are *mutually exclusive* during each iteration. The extent of the aforementioned influence however depends on the percentage of overlap between test sets. In other words, if the overlap is expected to be very small, as is typically expected within the context of sparse datasets, the influence will be insignificant when conducting an *independent* samples t-test. In order to mitigate this problem, Dietterich [69] suggested that a k-fold cross-validation protocol be employed by creating k *mutually exclusive* partitions of the data, which results in k *mutually exclusive* test sets which are *fixed* for *all* experiments. In this way, a paired t-test may be conducted between the two distributions of k statistical performance measures for two different experiments.

The decision to employ RRS with $n = 9$ within the context of this study for the purpose of partitioning the *small* number of available *individuals* is however based on the notion that the probability of selecting 9 non-representative test sets is even further reduced when compared to a 9-fold cross-validation protocol, which *further* minimises the probability of bias of the distribution of statistical performance measures. The probability of overlap between test sets associated with *different* iterations of *different* experiments is also *sufficiently small* within the context of this study, since the m samples in a given, randomly selected test set are *paired* by the Cartesian product, which results in m^2 test samples, from which a relatively small number of samples is *also* randomly selected during each iteration.

It is important to note that, while the utilisation of RRS for the purpose of statistically comparing the proficiency of two machine learning algorithms is advised against by Dietterich [69], the paired t-tests conducted in this study are *not* conducted on a *per-system* basis, but rather on a *per-experimental scenario* basis. In other words, 96 different paired t-tests are *not* conducted in this study on the two sets of 9 statistical performance measures for *each* of the 96 systems within the context of two different experimental scenarios. Instead, a *single* paired t-test is conducted by considering the 96 *averages* of the 9 statistical performance measures for *all* of the 96 systems within the context of the two different experimental scenarios in question. By doing so, the 96 average scores may be assumed to be normally distributed and unbiased towards any single data partition, provided that the standard deviation across the 9 individual scores for the systems in question is sufficiently small. This significantly reduces the risk of a high type I error rate. The two sets of 96 unbiased average statistical performance measures may therefore be compared by employing the *paired* t-test in which the "subjects" of the test constitutes the 96 systems proposed in this study.

The protocol for comparing the proficiency of the proposed systems between two different experimental scenarios is detailed in Section 8.7.

7.3 Data augmentation: individual dependent scenario

The process of data augmentation involves the application of a set of transformations to each sample in a dataset to increase the dataset's size in situations where acquiring additional *actual* samples is time-consuming. This technique is frequently used within the context of deep learning, since larger training datasets are generally more beneficial than smaller ones. It has additionally been demonstrated by Shorten and Khoshgoftaar [71] that data augmentation may enhance the generalisation ability of neural networks, which leads to increased proficiency when prompted with unseen samples.

It is important to note that the augmentation process must replicate the expected types of variations that might arise during the acquisition of actual samples. The data augmentation strategy proposed in this study therefore comprises of image rotation and translation, as well as Gaussian smoothing in order to incorporate some random noise for the purpose of simulating possible variations in orientation between multiple actual samples acquired from the same individual.

The proposed augmentation strategy is based on the number of samples obtained per client during enrolment, and is therefore *calibrated* accordingly within the context of the employed databases. The proposed augmentation strategies are detailed in Sections 7.3.1 and 7.3.2 within the context of the Bosphorus and Wilches databases respectively.

7.3.1 Bosphorus database

During the enrolment process, it is assumed that each client provides a total of 8 authentic hand-vein images. These images are randomly divided into 4 authentic training samples and 4 authentic validation samples. The remaining 4 (out of the available 12) images are used as authentic test images for the purpose of simulation. It is important to note that, within the context of the Bosphorus database, the augmentation strategy applied to the authentic training, validation, and test sets is identical, since each set contains 4 images.

Each input image from the Bosphorus dataset, that is either (1) the original ROI (see Figure 3.19), or (2) the CLAHE contrast-enhanced ROI (see Figure 3.20), or (3) the binarised ROI (see Figure 3.26), undergoes the following data augmentation protocol:

- 20 unique rotations through an angle θ, where $\theta \in \{-5° + \frac{k}{2} \mid k \in \{0°, 1°, 2°, ..., 20°\}\}$, followed by
- 20 random translations of the pixel values in the x and y directions, where $x, y \in \{-2, -1, 0, 1, 2\}$, and
- Gaussian smoothing with $\sigma = 0.4$.

In this way, a total of 20 augmented samples are produced for each image within the context of the Bosphorus dataset.

7.3.2 Wilches database

Certain adjustments to the data augmentation protocol outlined in Section 7.3.1 is required within the context of the Wilches database due to the fact that only 4 authentic images are available per individual, as opposed to 12 within the context of the Bosphorus database. These adjustments are necessary in order to obtain an *equal* number of *positive* augmented training, validation, and test samples for each individual within the context of the Wilches database. The data augmentation protocol applied to the training, validation, and test *imposters* within the context of the Wilches database is the same as those outlined in Section 7.3.1.

During the enrolment process, it is assumed that each client provides a total of 3 authentic hand vein images. These images are randomly partitioned into 2 authentic training samples and 1 authentic validation sample. The remaining image is used as an authentic test image for the purpose of simulation.

Each positive authentic *training* image from the Wilches dataset is subjected to the following augmentation protocol, which produces a total of 80 (2×40) positive authentic training images:

- 40 unique rotations through an angle θ, where $\theta \in \{-5° + \frac{k}{4} \mid k \in \{0°, 1°, 2°, ..., 40°\}\}$, followed by
- 40 random translations of the pixel values in the x and y directions, where $x, y \in \{-2, -1, 0, 1, 2\}$, and
- Gaussian smoothing with $\sigma = 0.4$.

Each positive authentic *validation* and *test* image is subjected to the following augmentation protocol, which produces a total of 80 (1×80) positive authentic validation and test images:

- 80 unique rotations through an angle θ, where $\theta \in \{-5° + \frac{k}{8} \mid k \in \{0°, 1°, 2°, ..., 80°\}\}$, followed by
- 80 random translations of the pixel values in the x and y directions, where $x, y \in \{-2, -1, 0, 1, 2\}$, and
- Gaussian smoothing with $\sigma = 0.4$.

The data partitioning and cross-validation protocols are discussed in Sections 7.4 and 7.5 within the context of the IDS and IIS respectively.

7.4 Data partitioning and cross-validation: individual dependent scenario

The employed data partitioning and cross-validation protocol is identical for *both* the Bosphorus and Wilches databases within the context of the individual dependent scenario (IDS). It is important to note that the GenVeins database (see Chapter 4) is *not* employed within the context of the IDS.

Both the Bosphorus and Wilches databases employed in this study comprises of a total of 100 individuals. These individuals are randomly separated into 70 clients and 30 *test* imposters, where the purpose of the test imposters is to represent criminals during simulation. Said partitioning protocol is repeated a total of three times for the purpose of cross-validation.

It is important to note that, since a tailor-made network is created for *each* individual client within the context of the IDS, a total of 69 *other* clients are available for the purpose of representing imposters during the creation of a tailor-made network for the client in question. A total of 35 training and 34 validation *imposters* are therefore randomly selected out of the available 69 other clients. This process is also repeated three times for each client for the purpose of cross-validation.

7.4.1 Training

The data augmentation protocol discussed in Section 7.3 is first employed for the purpose of generating a sufficient number of training samples, which results in

- 80 (4×20) and 80 (2×40) authentic training samples within the context of the Bosphorus and Wilches databases respectively and
- 8 400 ($12 \times 20 \times 35$) and 2 800 ($4 \times 20 \times 35$) imposter training samples within the context of the Bosphorus and Wilches databases respectively.

In order to obtain positive and negative training pairs within the context of the IDS, the augmented training samples are paired with each other, as well as with the augmented imposter training samples by utilising the Cartesian product, which produces

- 6 400 (80×80) and 6 400 (80×80) positive training pairs within the context of the Bosphorus and Wilches databases respectively and
- 672 000 (80×8400) and 224 000 (80×2800) negative training pairs within the context of the Bosphorus and Wilches databases respectively.

It has been experimentally determined that 2 000 positive and 2 000 negative training pairs that are randomly selected and ordered in an alternating fashion constitutes a sufficient number of training pairs in order to minimise the loss function within the context of the IDS and the systems proposed in this study. Said experimental calibration comprises of iteratively training a subset of the proposed systems with a different number of training samples in each iteration. The main goals of said experimental calibration comprises of (1) achieving maximum training performance while (2) optimising training time, which is an important factor to consider due to the large number of experiments conducted within the context of the IDS. Another factor which also influenced the decision constitutes the fact that the complexity of the optimisation function is fairly simple within the context of the IDS in the sense that a network is trained to authenticate a *single* individual, which motivates the use of fewer training samples in order to avoid potential overfitting.

7.4.2 Validation

The data augmentation protocol discussed in Section 7.3 is subsequently employed for the purpose of generating a sufficient number of validation samples, which results in

- 80 (4×20) and 80 (1×80) authentic validation samples within the context of the Bosphorus and Wilches databases respectively and
- 8 160 ($12 \times 20 \times 34$) and 2 720 ($4 \times 20 \times 34$) imposter validation samples within the context of the Bosphorus and Wilches databases respectively.

In order to obtain positive and negative validation pairs within the context of the IDS, the augmented authentic validation samples are subsequently paired with each other, as well as with the augmented imposter validation samples by utilising the Cartesian product, which produces

- 6 400 (80×80) and 6 400 (80×80) positive validation pairs within the context of the Bosphorus and Wilches databases respectively and
- 652 800 ($80 \times 8\,160$) and 217 600 ($80 \times 2\,720$) negative validation pairs within the context of the Bosphorus and Wilches databases respectively.

It has been experimentally determined that 2 000 positive and 2 000 negative validation pairs that are randomly selected and ordered in an alternating fashion constitutes a sufficient number of validation pairs for the systems proposed in this study within the context of the IDS for the purpose of obtaining unbiased statistical performance measures from which to determine the optimal probabilistic threshold. Said experimental calibration comprises of a similar protocol to the one employed for the purpose of selecting the final training set, of which the goal is to determine the minimum number of validation samples for which system performance changes insignificantly over all discrete probabilistic thresholds. This enables the selection of a trustworthy optimal probabilistic threshold while minimising execution time.

7.4.3 Testing

The data augmentation protocol discussed in Section 7.3 is finally employed for the purpose of generating a sufficient number of test samples, which results in

- 80 (4×20) and 80 (1×80) authentic test samples within the context of the Bosphorus and Wilches databases respectively and
- 7200 ($12 \times 20 \times 30$) and 2 400 ($4 \times 20 \times 30$) imposter test samples within the context of the Bosphorus and Wilches databases respectively.

In order to obtain positive and negative test pairs within the context of the IDS, the augmented authentic test samples are subsequently paired with each other, as well as with the augmented imposter test samples by utilising the Cartesian product, which produces

- 6 400 (80×80) and 6 400 (80×80) positive test pairs within the context of the Bosphorus and Wilches databases respectively and
- 576 000 ($80 \times 7\,200$) and 192 000 ($80 \times 2\,400$) negative test pairs within the context of the Bosphorus and Wilches databases respectively,

from which 2 000 positive and 2 000 negative test pairs are randomly selected and ordered in an alternating fashion. This produces a balanced test set which comprises of a sufficient number of test pairs within the context of the IDS in order to obtain unbiased test results.

7.5 Data partitioning and cross-validation: individual independent scenario

The data partitioning and cross-validation protocols for the benchmark experiment (BME), the first GenVeins experiment (FGE) and the second GenVeins experiment (SGE) are detailed in Sections 7.5.1 and 7.5.2 respectively. It is important to note that the data partitioning and cross-validation protocol is identical for the FGE and the SGE.

7.5.1 Data partitioning and cross-validation: the benchmark experiment

Recall from Chapter 3 that there are a total of 100 individuals, for each of whom a total of 12 and 4 images are available within the context of the Bosphorus and Wilches database respectively. These 100 individuals are partitioned as follows within the context of the BME in order to obtain sets of *actual* training, validation and test individuals.

- 30 actual training clients and 30 actual training imposters are first randomly selected from the available 100 actual individuals.
- 10 actual validation clients and 10 actual validation imposters are subsequently randomly selected from the remaining 40 actual individuals.
- 10 actual test clients and 10 actual test imposters are finally randomly selected from the remaining 20 actual individuals.

This partitioning protocol is repeated a total of nine times for the purpose of cross-validation within the context of the BME.

7.5.1.1 Training set: the benchmark experiment

In order to obtain balanced training sets within the context of the BME, the actual authentic training samples associated with the *same* client are paired with each other, as well as with the actual imposter training samples by utilising the Cartesian product, which produces

- 4 320 ($(12 \times 12) \times 30$) actual positive training pairs and
- 129 600 ($12 \times (12 \times 30) \times 30$) actual negative training pairs

within the context of the Bosphorus database, from which 4 320 actual positive training pairs and 4 320 actual negative training pairs are randomly selected and ordered in an alternating fashion, and

- 480 (4 × 4 × 30) actual positive training pairs and
- 14 400 (4 × (4 × 30) × 30) actual negative training pairs

within the context of the Wilches database, from which 480 actual positive training pairs and 480 actual negative training pairs are randomly selected and ordered in an alternating fashion.

7.5.1.2 Validation set: the benchmark experiment

In order to obtain balanced validation sets within the context of the BME, the actual authentic validation samples associated with the *same* client are paired with each other, as well as with the actual imposter validation samples by utilising the Cartesian product, which produces

- 1 440 (12 × 12 × 10) actual positive validation pairs and
- 14 400 (12 × (12 × 10) × 10) actual negative validation pairs

within the context of the Bosphorus database, from which 1 440 actual positive and negative validation pairs are randomly selected and ordered in an alternating fashion, and

- 160 (4 × 4 × 10) actual positive validation pairs and
- 1 600 (4 × (4 × 10) × 10) actual negative validation pairs

within the context of the Wilches database, from which 160 actual positive validation pairs and 160 actual negative validation pairs are randomly selected and ordered in an alternating fashion.

7.5.1.3 Test set: the benchmark experiment

In order to obtain balanced test sets within the context of the BME, the actual authentic test samples associated with the *same* client are paired with each other, as well as with the actual imposter test samples by utilising the Cartesian product, which produces

- 1 440 (12 × 12 × 10) actual positive test pairs and
- 14 400 (12 × (12 × 10) × 10) actual negative test pairs

within the context of the Bosphorus database, from which 1 440 actual positive test pairs and 1 440 actual negative test pairs are randomly selected and ordered in an alternating fashion, and

- 160 (4 × 4 × 10) actual positive test pairs and
- 1 600 (4 × (4 × 10) × 10) actual negative test pairs

within the context of the Wilches database, from which 160 actual positive test pairs and 160 actual negative test pairs are randomly selected and ordered in an alternating fashion.

It is important to note that the size of the training, validation and test sets obtained within the context of the BME is relatively small within the context of ANNs which typically require thousands of samples in order to achieve desired proficiency. Another experiment called the “augmented BME” is also conducted in this study, in which the samples from the Bosphorus and Wilches databases are *first* augmented by the same protocols outlined in Sections 7.3.1 and 7.3.2 respectively. The results obtained within the context of the augmented BME are however *significantly* worse than the results associated with the BME, and are presented and discussed in the appendix of this dissertation (see Section A.4) for reference.

7.5.2 Data partitioning: the first and second GenVeins experiments

Recall from Chapter 4 that there are a total of 20 000 fictitious individuals in the GenVeins database. It has been experimentally determined that 8 000 fictitious individuals are sufficient for the purpose of training, while a total of 6 000 fictitious individuals are sufficient for the purpose of validation. These 14 000 fictitious individuals are partitioned as follows within the context of the FGE and the SGE in order to obtain sets of *fictitious* training and validation individuals.

- 4 000 fictitious training clients and 4 000 fictitious training imposters are first randomly selected from the available 14 000 fictitious individuals.
- 3 000 fictitious validation clients and 3 000 fictitious validation imposters are subsequently randomly selected from the remaining 6 000 fictitious individuals.

In order to obtain sets of *actual* authentic and imposter test individuals, the 100 actual individuals from the Bosphorus and Wilches databases are randomly partitioned into 70 actual authentic test individuals and 30 actual imposter test individuals.

This partitioning protocol is repeated a total of nine times for the purpose of cross-validation.

7.5.2.1 Training set: FGE and SGE

In order to obtain balanced training sets within the context of the FGE and the SGE, the fictitious authentic training samples associated with the *same* client are paired with each other, as well as with the fictitious imposter training samples by utilising the Cartesian product, which produces

- 64 000 ($4 \times 4 \times 4000$) fictitious positive training pairs and
- 256 000 000 ($4 \times (4 \times 4000) \times 4000$) fictitious negative training pairs,

from which 5000 fictitious positive training pairs and 5000 fictitious negative training pairs are randomly selected and ordered in an alternating fashion. The protocol by which the number of training samples is determined is identical to the one employed within the context of the IDS (see Section 7.4.1).

7.5.2.2 Validation set: FGE and SGE

In order to obtain balanced validation sets within the context of the FGE and the SGE, the fictitious authentic validation samples associated with the *same* client are paired with each other, as well as with the fictitious imposter validation samples by utilising the Cartesian product, which produces

- 48 000 (4 × 4 × 3000) fictitious positive validation pairs and
- 144 000 000 (4 × (4 × 3000) × 3000) fictitious negative validation pairs,

from which 5000 fictitious positive validation pairs and 5000 fictitious negative validation pairs are randomly selected and ordered in an alternating fashion. The protocol by which the number of validation samples is determined is identical to the one employed within the context of the IDS (see Section 7.4.2).

7.5.2.3 Test set: FGE and SGE

In order to obtain balanced test sets within the context of the FGE and the SGE, the *actual* authentic test samples associated with the *same* client are paired with each other, as well as with the *actual* imposter test samples by utilising the Cartesian product, which produces

- 10 080 (12 × 12 × 70) *actual* positive test pairs and
- 302 400 (12 × (12 × 30) × 70) *actual* negative test pairs

within the context of the Bosphorus database, from which 10 080 *actual* positive test pairs and 10 080 *actual* negative test pairs are randomly selected and ordered in an alternating fashion, and

- 1 120 (4 × 4 × 70) *actual* positive test pairs and
- 33 600 (4 × (4 × 30) × 70) *actual* negative test pairs

within the context of the Wilches database, from which 1 120 *actual* positive test pairs and 1 120 *actual* negative test pairs are randomly selected and ordered in an alternating fashion.

7.6 Protocols

The training, validation and test protocols developed for the purpose of gauging the proficiency of the proposed systems within the context of the individual dependent scenario (IDS) and individual independent scenario (IIS) are discussed in this section. It is important to note that these protocols are the same within the context of *both* the IDS and IIS.

7.6.1 Training protocol

The parameters associated with the training of *all* the different systems proposed in Chapter 6 comprise of

- the Averaged Stochastic Gradient Descent (ASGD) optimisation algorithm [72] with a momentum of 0.9 (see Section 5.3.4.5),

- a *constant* learning rate (see Section 5.3.4.5) of $1e^{-4}$ and
- a mini-batch size (see Section 5.3.4.5) of 32.

The aforementioned parameters, together with the hyperparameters of the CNN-based feature extractors (see Section 6.2), were *manually* tuned *prior* to the start of the final experiments. The manual hyperparameter tuning protocol employed in this study is outlined below.

- A feasible search space of hyperparameters is first defined by considering the complexity of (1) the detail contained in the input images and (2) the classification task in question. Given the relatively simple nature of the NIR images employed in this study when compared to color images containing dense information, together with the fact that only two classes are being trained (see Section 6.3), it is feasible to assume that (1) the CNN-based feature extractors may contain a relatively small number of convolutional and pooling layers when compared to deep CNNs such as ResNet [65], while a constant, smaller learning rate may be sufficient to ensure consistent convergence of the networks.
- The systems in question are subsequently trained in an iterative fashion with a unique combination of manually defined hyperparameters during each iteration, after which system proficiency is observed.
- The optimal set of hyperparameters may finally be obtained by ranking the proficiency of the aforementioned systems in a descending fashion.

It is important to note that, while a manual hyperparameter tuning protocol may be sufficient within the context of relatively simple classification tasks, there are many drawbacks and shortcomings when compared to a properly defined *automatic* hyperparameter tuning protocol. These drawbacks are discussed in Section 9.1.4. A design choice was however made during the course of this study to opt for a manual hyperparameter tuning protocol due to the fact that one of the main objectives of this study is to obtain *benchmark* results of a number of well-known, feasible preprocessing protocols and neural network architectures within the context of the novel experimental protocols proposed in this study. In other words, the objectives of this study do *not* include the development of SOTA hand vein-based authentication systems, which renders the utilisation of a manual hyperparameter tuning protocol sufficient.

It is also important to note that careful initialization of neural networks is crucial in order to prevent training failures [62]. The initialisation problem is addressed in this study by utilising the same protocol suggested by Beukes and Coetzer [24], which is summarised as follows: First, the weights of each network are randomly initialised, after which the training stage begins. In the event that the training loss decreases insignificantly during the first 10 epochs, the weights of the network is simply reinitialized, after which the training stage is restarted. It is experimentally determined that a maximum of 10 reinitializations is sufficient in order to ensure convergence.

This concludes the training protocol proposed in this study.

7.6.2 Validation protocol

The purpose of the validation stage within the context of this study is *only* to determine an optimal probabilistic threshold for accepting a questioned sample. This forms part of the hyperparameter tuning of the systems proposed in this study, and is fundamental for the purpose of maximising the proficiency of the proposed systems as opposed to simply employing a threshold of 0.5 [73]. An *early stopping criterion* is *not* applicable within the context of this study, since there is no evidence of overfitting during training within the context of the proposed systems and the problem addressed in this study.

The optimal probabilistic threshold is determined by employing a total of four different threshold selection criteria. The proficiency of each system is gauged after predicting the labels of the validation data by calculating the false acceptance rate (FAR) and false rejection rate (FRR) at each of 50 discrete possible probabilistic thresholds in the interval $[0,1]$.

The FAR and FRR are defined as $\frac{\text{FP}}{\text{N}}$ and $\frac{\text{FN}}{\text{P}}$ respectively, where FP and FN denotes the number of false positives and false negatives, while N and P denotes the total number of negative and positive samples in the employed dataset. The four threshold selection criteria are defined as follows by employing the definitions of the FAR, the FRR and the average error rate (AER), where AER = $\frac{\text{FAR+FRR}}{2}$:

- The equal error rate (EER), that is the probabilistic threshold where FAR = FRR,
- the *zero* FAR (FAR_{zero}), that is the smallest probabilistic threshold where FAR = 0,
- the *zero* FRR (FRR_{zero}), that is the largest probabilistic threshold where FRR = 0 and
- the *minimum* AER (AER_{min}), that is the probabilistic threshold corresponding to the smallest AER.

This concludes the validation protocol proposed in this study.

7.6.3 Test protocol

During the test stage, the optimal probabilistic threshold determined during the validation stage is employed for the purpose of predicting the labels of the test data. Once the proficiency of *all* the systems have been gauged, the AERs are ranked in a descending fashion in order to determine the top performing system. This concludes the test protocol proposed in this study.

7.7 Concluding remarks

The experimental protocols developed within the context of this study for the purpose of gauging the proficiency of the proposed systems were discussed in this chapter. The data augmentation strategies employed for the purpose of generating a sufficient number of data samples were also detailed in this chapter, along with the data partitioning and cross-validation protocols which are utilised for the purpose of gauging the stability of the proposed systems across different training, validation and test sets. The training, validation and test protocols were finally also outlined in this chapter. The experimental results obtained within the context of the proposed experimental scenarios, that is the IDS and IIS, are presented and discussed in the following chapter.

Chapter 8

Experiments

8.1 Introduction

The results obtained by gauging the proficiency of the proposed systems within the context of the individual dependent scenario (IDS) (see Section 7.1) and the individual independent scenario (IIS) (see Section 7.1) are presented and discussed in this chapter. The following issues form the basis of the discussions and are repeatedly addressed throughout this chapter:

1. the risks associated with employing a sub-optimal combination of system components,
2. the challenges faced when an *insufficient number* of *different* individuals are available for training within the context of the IIS and
3. the impact of so-called *vein deformity* across different images belonging to the *same* individual on system proficiency.

The first issue is addressed by employing the novel protocol proposed in Chapter 6, which comprises of gauging the proficiency of a large number of different combinations of system components within the context of a specific database and experimental scenario, after which the top performing system may be determined. This protocol is a simple solution to the problem of establishing the optimal system design *prior* to the start of the experiment. The experimental results presented in this chapter clearly quantify the significant differences between the proficiency of sub-optimal and top performing systems.

The main challenge faced when only a small number of *different* individuals are available for training constitutes the fact that the employed system will inevitably become *biased* towards the training individuals. It is shown in Section 8.6.1 that systems trained in this way do *not* perform well on images from *unseen* individuals. Another challenge constitutes the fact that the proficiency of the system cannot be trusted when trained and tested on small mutually exclusive sets of individuals, since a small number of individuals *cannot* be considered representative of the population. Two different solutions to the aforementioned issue are proposed in this study, namely (1) training a tailor-made network for each client (IDS) or (2) utilising the GenVeins database (see Chapter 4) for the purpose of training and validation within the context of the IIS. Both of these solutions are shown to achieve outstanding proficiency, while providing additional benefits (see Section 1.5) which are *not*, to the best of the author's knowledge, provided by *any* SOTA system in current literature.

The phrase "vein deformity" refers to the deformation of the hand vein structure across multiple samples of an individual's hand when acquired by means of near infra-red (NIR) imaging under improperly-controlled conditions. More specifically, the variation in orientation of the hand of a specific individual during the acquisition of multiple near infra-red images leads *not only* to perceived variation in *orientation*, but *also* to perceived *deformity* of the hand veins. The aforementioned deformity is severe for some individuals, and occurs more frequently within the context of the Bosphorus database, as a result of the less-controlled conditions during acquisition. It is shown that, by granting individuals a total of three chances for authentication during testing within the context of the second GenVeins experiment (SGE), the negative impact of the aforementioned problem on system proficiency can be drastically reduced (see Section 8.6.3).

The statistical performance measures employed for the purpose of gauging the proficiency of the proposed systems are presented in Section 8.2, while the technique employed for the purpose of gaining insight into the predictions of the proposed systems is detailed in Section 8.3. A legend of abbreviations and descriptions of the different system components is defined in Section 8.4. The results obtained by gauging the proficiency of the proposed systems within the context of the IDS and IIS are presented and discussed in Sections 8.5 and 8.6 respectively. Concluding remarks are finally provided in Section 8.8.

8.2 Statistical performance measures

The statistical performance measures employed for the purpose of gauging the proficiency of the proposed systems are detailed in this section. Recall that the problem addressed in this study is that of identity *verification*, which involves a questioned sample and a claim about which client the sample belongs to. A questioned sample may therefore either be a *positive* sample (p), that is a sample belonging to the client in question, or a *negative* sample (n), that is a sample belonging to an imposter. The outcome of the verification of a questioned sample may therefore either constitute

1. a true positive (tp), that is the outcome in which a positive sample is accepted,
2. a true negative (tn), that is the outcome in which a negative sample is rejected,
3. a false positive (fp), that is the outcome in which a negative sample is accepted, or
4. a false negative (fn), that is the outcome in which a positive sample is rejected.

A *trial* comprises of the classification of the entire test set within the context of a *specific* iteration of the cross-validation protocol employed in this study, namely repeated random sampling (RRS), of a given experiment. The total number of positive samples within a single trial is denoted by P, while the total number of negative samples is denoted by N. The total number of verification outcomes of a single trial is denoted by

1. TP, that is the total number of true positives,
2. TN, that is the total number of true negatives,
3. FP, that is the total number of false positives and
4. FN, that is the total number of false negatives.

The following five statistical performance measures are employed in this study for the purpose of gauging the proficiency of the proposed systems within the context of a single trial:

$$\text{False rejection rate (FRR)} = \frac{\text{FN}}{\text{P}} \tag{8.1}$$

$$\text{False acceptance rate (FAR)} = \frac{\text{FP}}{\text{N}} \tag{8.2}$$

$$\text{Average error rate (AER)} = \frac{\text{FRR} + \text{FAR}}{2} \tag{8.3}$$

$$\text{Specificity (SPE)} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{8.4}$$

$$\text{Sensitivity (SEN)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{8.5}$$

The FAR and FRR denote errors made by the system, while the SPE and SEN are measures that provide insight into the discriminative ability of the system. SPE is a measure of the ability of the system to correctly *identify* true negatives amongst all test samples, while SEN is a measure of the ability of the system to correctly *identify* true positives amongst all test samples.

In addition to the five statistical performance measures listed above, the standard deviation (STD) of the AERs associated with the nine repeated random sampling (RRS) iterations is also reported for each experiment for the purpose of gaining insight into the stability of the proposed systems.

The technique employed for gaining insight into the predictions of the proposed systems is detailed in the next section.

8.3 Model explainability

ANNs are often referred to as so-called "black-box" models, since internal computations are not visible to the user. Algorithms for gaining insight and transparency into the decision-making process of ANNs have therefore become increasingly necessary due to the significant increase in neural network utilisation during recent years. One of the more popular algorithms commonly employed for the purpose of model explainability is called Shapley Additive Explanations [74].

An overview of Shapley Additive Explanations (SHAP) is provided in Section 8.3.1, while the specific implementation of SHAP that is employed within the context of this study is explained in Section 8.3.2.

8.3.1 An overview of SHAP

The Shapley Additive Explanations (SHAP) algorithm comprises of a unified approach that is able to quantify the importance of each individual feature in a sample to the prediction made by the model in question. Simple models such as linear regression models are relatively easy to interpret without additional assistance, while a so-called *explanation model* is required within the context of more complex machine learning algorithms such as ANNs. An explanation model is defined as any interpretable approximation of the machine learning model in question.

The SHAP explanation model identifies the class of additive feature importance *methods* [74], which comprises of six existing additive feature importance methods, after which three desired so-called feature attribution properties are unified from these methods, namely (1) local accuracy, (2) absence and (3) consistency.

It is proved by Lundberg and Lee [74] that, by utilising cooperative game theory, the SHAP explanation model has a unique solution for *any* model in question. This, together with the fact that the output of SHAP, called SHAP *values*, encapsulate all three desirable properties of feature attribution, is sufficient motivation that the SHAP explanation model is a feasible solution to the problem of model interpretability within the context of complex machine learning models such as ANNs. The SHAP value of a given feature x constitutes a quantification of the attribution of x to the prediction made by the machine learning model.

It is important to note that, due to the complexity of the mathematical definition of SHAP, the true SHAP values are computationally very expensive to calculate. Different approximation algorithms of SHAP values are therefore proposed by Lundberg and Lee [74] for many popular machine learning models. The so-called "Deep SHAP" algorithm, which is the algorithm proposed by Lundberg and Lee [74] for the purpose of approximating SHAP values within the context of ANNs, is employed in this study and further discussed in the following section.

8.3.2 Deep SHAP

The Deep SHAP algorithm approximates SHAP values by combining Shapley values [75] and the DeepLIFT explanability method [76]. DeepLIFT satisfies the local accuracy and absence properties of the explanation model, while Shapley values satisfy the consistency property. The Deep SHAP algorithm approximates the SHAP value of a given feature in an input sample, assuming the machine learning model in question has already been trained, in the following manner:

1. **Generate reference sample**: A reference sample is generated by randomly altering the value of the feature in question.

2. **Execute forward pass**: A forward pass through the model is executed for the given sample *and* the corresponding reference sample.

3. **Estimate isolated feature importance**: The *isolated* importance of the feature in question to the output of the model is quantified by calculating the difference between the model output of the given sample and the reference sample.

4. **Estimate interactive feature importance**: A number of feature subsets are first selected from the reference sample, all of which contain the feature in question. Cooperative game theory is then utilized in order to calculate the Shapley value of the feature in question. The *interactive* importance of the feature in question is subsequently obtained by averaging the Shapley values of all feature subsets for the feature in question.

5. **Calculate the SHAP value**: The importance of the isolated and interactive features are finally added together in order to obtain the approximate SHAP value for the feature in question

Steps one to five are repeated for all input features in order to obtain the approximate SHAP values for the input sample.

Three examples of the output of the Deep SHAP algorithm within the context of a CNN that has been trained to classify binary images of handwritten digits is visualised in Figure 8.1 for reference. Input samples containing a handwritten sample of three digits are depicted on the left, while the SHAP values associated with the input samples on the left are depicted on the right for each of the ten digit classes. Red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question.

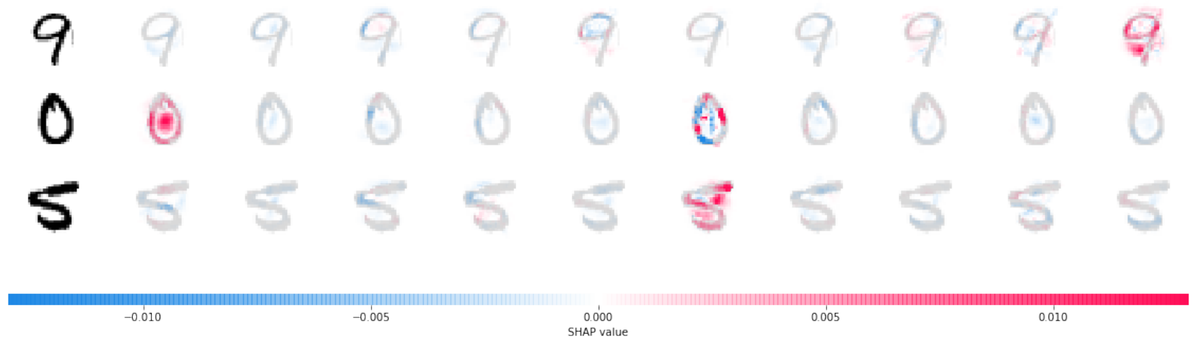


Figure 8.1: Three examples of the output of the Deep SHAP algorithm within the context of a CNN that has been trained to classify binary images of handwritten digits (courtesy of [3]). Red SHAP values indicate *positive* evidence that the questioned image belongs to the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question.

8.4 Abbreviations of system components

A legend of abbreviations and descriptions of all the different system components is defined in Table 8.1 in order to simplify the visualisation of the results presented in the following few sections. References to this table are made where necessary for convenience.

Table 8.1: Abbreviations of different system components and their descriptions.

Preprocessing protocols
NO: No preprocessing (see Section 3.3.1)
CL: CLAHE-based contrast enhancement (see Section 3.3.2)
FB: Full binarisation (see Section 3.3.3)
Neural network architectures
2CH: Two-channel networks (see Section 5.5.1)
SM: Siamese networks (see Section 5.5.2)
CNN-based feature extraction
ST: Standard (see Section 6.2.1)
BN: Standard with added batch normalisation layers (see Section 6.2.2)
DO: Standard with added dropout layers (see Section 6.2.3)
BD: Standard with added batch normalisation and dropout layers (see Section 6.2.4)
Probabilistic threshold criteria

EER: Equal error rate (see Section 7.6.2)
AER_{min}: Minimum average error rate (see Section 7.6.2)
FAR_{zero}: Zero false acceptance rate (see Section 7.6.2)
FRR_{zero}: Zero false rejection rate (see Section 7.6.2)

The results obtained when gauging the proficiency of the proposed systems within the context of the IDS is presented in the next section.

8.5 Results: individual dependent scenario

The purpose of the individual dependent scenario (IDS) is to simulate a simple real-world scenario in which a tailor-made network is trained for *each* client during enrolment. This solution is a feasible alternative to training an individual *independent* network, in which case near infra-red (NIR) images of the hand veins of a large number of *different* individuals must be acquired *prior* to training. The results presented in this section clearly show that the majority of the proposed systems are highly proficient when trained to authenticate a single client. The results of the individual dependent experiment (IDE) (see Section 7.1) within the context of the Bosphorus and Wilches databases are presented in the following fashion:

1. The results of the top 20 systems are presented and discussed in Section 8.5.1.
2. The SHAP values for a selected number of clients within the context of the top performing systems are presented and discussed in Section 8.5.2.
3. The extent of the deformity problem stated in Section 8.1 is emphasised and discussed in Section 8.5.3.

8.5.1 System ranking: individual dependent experiment

Recall from Chapter 6 that a total number of 96 systems are proposed in this study. For brevity, the performance measures for only the top 20 systems are presented and discussed within the context of the Bosphorus and Wilches databases respectively. The full results are presented in the appendix of this dissertation (see Section A.2).

The results for the top 20 systems within the context of the Bosphorus and Wilches databases and the IDE are depicted in Tables 8.2 and 8.3 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Table 8.2: The results of the top 20 systems ranked according to the AER within the context of the IDE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	SM BD CL AER_{min}	1.01	2.405	1.708	98.99	97.595	1.351

2	SM ST FB AER_{min}	1.035	2.504	1.769	98.965	97.496	1.095
3	SM ST FB EER	1.563	2.16	1.862	98.437	97.84	1.151
4	SM BD CL EER	1.523	2.249	1.886	98.477	97.751	1.552
5	SM BN FB AER_{min}	0.462	3.455	1.958	99.538	96.545	0.689
6	SM DO FB AER_{min}	0.654	3.516	2.085	99.346	96.484	1.427
7	SM BD FB AER_{min}	0.338	3.84	2.089	99.662	96.16	0.701
8	SM DO FB EER	1.148	3.139	2.144	98.852	96.861	1.436
9	SM BD NO AER_{min}	1.815	2.581	2.198	98.185	97.419	2.438
10	2CH ST FB AER_{min}	0.961	3.45	2.206	99.039	96.55	1.299
11	SM BN CL AER_{min}	1.672	2.77	2.221	98.328	97.23	1.86
12	SM BN FB EER	2.026	2.591	2.309	97.974	97.409	0.855
13	2CH ST FB EER	2.054	2.69	2.372	97.946	97.31	1.321
14	2CH DO FB AER_{min}	1.142	3.616	2.379	98.858	96.384	1.612
15	2CH BN FB AER_{min}	0.798	3.971	2.385	99.202	96.029	1.155
16	SM BD FB FAR_{zero}	0.029	4.777	2.403	99.971	95.223	1.396
17	SM BN CL EER	1.964	2.907	2.435	98.036	97.093	2.054
18	SM BD FB EER	1.82	3.057	2.438	98.18	96.943	0.908
19	SM BD NO EER	1.951	3.093	2.522	98.049	96.907	3.044
20	2CH BD FB AER_{min}	0.91	4.142	2.526	99.09	95.858	1.117

Table 8.3: The results of the top 20 systems ranked according to the AER within the context of the IDE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	SM BD FB AER_{min}	0.507	1.198	0.853	99.493	98.802	0.928
2	SM BD FB EER	0.763	1.113	0.938	99.237	98.887	1.017
3	SM BN FB AER_{min}	0.533	1.435	0.984	99.467	98.565	1.286
4	SM BN FB EER	0.767	1.326	1.047	99.233	98.674	1.21
5	SM BD CL AER_{min}	0.943	1.542	1.242	99.057	98.458	1.704
6	SM BD NO AER_{min}	1.396	1.355	1.375	98.604	98.645	1.96
7	SM BD CL EER	1.227	1.845	1.536	98.773	98.155	2.603
8	SM BN CL AER_{min}	1.446	1.727	1.587	98.554	98.273	2.066
9	SM ST CL AER_{min}	1.802	1.58	1.691	98.198	98.42	2.134
10	SM ST FB AER_{min}	1.462	1.959	1.71	98.538	98.041	2.391
11	2CH BN NO AER_{min}	0.972	2.583	1.778	99.028	97.417	3.88
12	SM DO CL AER_{min}	0.954	2.702	1.828	99.046	97.298	3.379

13	SM BD NO EER	1.223	2.505	1.864	98.777	97.495	2.772
14	2CH BN NO EER	0.904	2.848	1.876	99.096	97.152	4.067
15	2CH BN CL AER_{min}	1.144	2.619	1.882	98.856	97.381	3.3
16	2CH BN FB AER_{min}	1.114	2.688	1.901	98.886	97.312	2.79
17	SM BN CL EER	1.427	2.376	1.901	98.573	97.624	2.398
18	2CH BN CL EER	1.237	2.669	1.953	98.763	97.331	3.395
19	2CH BD CL AER_{min}	1.048	2.864	1.956	98.952	97.136	3.282
20	2CH BD NO AER_{min}	1.132	2.795	1.963	98.868	97.205	3.632

The following noteworthy conclusions may be drawn from the results depicted in Tables 8.2 and 8.3 within the context of the IDE and the Bosphorus and Wilches databases respectively:

1. The systems in which more extensive preprocessing protocols such as contrast enhancement and full binarisation are employed tend to outperform the systems in which the "no preprocessing" protocol is employed, which indicates that the proposed systems benefit from enhanced textural detail within the context of the IDE.

2. The majority of systems in which the AER_{min} or EER threshold selection criterion is employed significantly outperform the systems in which either the FAR_{zero} or FRR_{zero} threshold selection criteria (see full results in Tables A.2 and A.3) is employed. This indicates that the systems benefit from employing a more balanced threshold selection criterion as opposed to employing either a very strict or very lenient criterion within the context of the IDE.

3. The differences between the top performing systems and the systems with the 20th rank are relatively small, which indicates that the so-called "sub-optimal system" problem is not a significant concern within the context of the IDS, provided that either the AER_{min} or EER threshold selection criterion is employed.

4. The standard deviations of the AERs are relatively small within the context of *both* databases, which indicates that the proposed systems are not overly sensitive to different random data partitions. This subsequently strengthens the notion that the employed cross-validation protocol, namely RRS (see Section 7.2), is successful in minimising the probability of bias for any one random data partition.

When the results depicted in Tables 8.2 and 8.3 are compared, it is clear that the majority of the proposed systems perform better within the context of the Wilches database. This may be due to the superior image acquisition protocol employed for the purpose of obtaining NIR images of the hand veins of the individuals within the context of the Wilches database.

The SHAP values for the top performing systems within the context of the Bosphorus and Wilches databases and the IDE are presented and discussed in the next section.

8.5.2 SHAP values: individual dependent experiment

The SHAP values (see Section 8.3) for a selected number of clients and test pairs within the context the IDE are depicted in Figure 8.2. The SHAP values are calculated for the CNN-based FEs (see Section 6.2) of the top performing systems depicted in Tables 8.2 and 8.3

within the context of the Bosphorus and Wilches databases respectively. Recall from Section 8.3.2 that red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question. It is also important to note the following in order to interpret Figure 8.2.

1. **The four principal images framed in black:**
   (a) Each of the four principal images are associated with a specific client.
   (b) The top two principal images are associated with two different clients from the Bosphorus database, while the bottom two principal images are associated with two different clients from the Wilches database.
2. **The four columns of subimages depicted in each of the four principal images framed in black:**
   (a) The first column of subimages depict the *reference* samples, while the second column of subimages depict the *questioned* samples. The first and second subimages in each of the four rows of images therefore constitute the test pairs in question. The test pairs are ordered in an alternating fashion, starting with a positive pair.
   (b) The third column of subimages depict the SHAP values of the corresponding test pair within the context of the *positive* class, while the fourth column of subimages depict the SHAP values of the corresponding test pair within the context of the *negative* class. It is therefore desired to have overwhelmingly red values in the *third* column of subimages for the *positive* test pairs (first and third rows of test pairs) and overwhelmingly red values in the *fourth* column of subimages for the *negative* test pairs (second and fourth rows of test pairs). It is important to note that, since a two-channel image cannot be displayed as an image, *only* the *questioned* sample (second column of subimages) is depicted in the third and fourth column of subimages in order to gain insight into the evidence that the *questioned* sample belongs to the positive or negative class respectively.

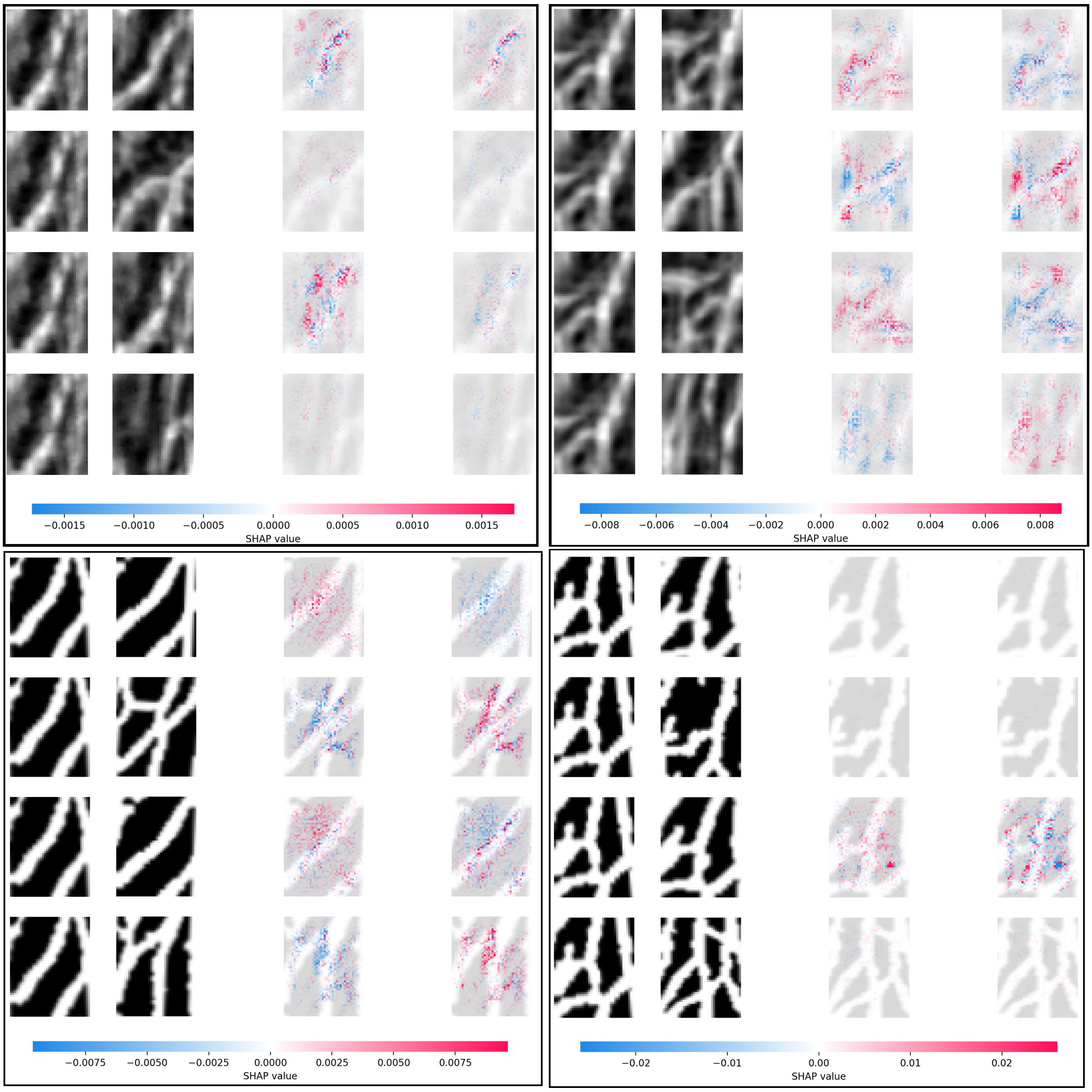


Figure 8.2: The SHAP values (see Section 8.3) for a selected number of clients and test pairs within the context of the Bosphorus and Wilches databases and the IDE. Red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question.

It is clear from Figure 8.2 that the CNN-based FEs of the top performing systems are proficient and robust when prompted to determine whether a questioned sample belongs to the positive or negative class respectively. More specifically, the CNN-based FEs in question are able to correctly associate higher feature importances with the true class labels for the majority of test pairs, which indicates that the CNN-based FEs of the top performing systems are highly proficient and robust across multiple databases within the context of the IDS.

The deformity problem is discussed in the next section within the context of the IDE.

8.5.3 The deformity problem: individual dependent experiment

In order to illustrate the extent of the deformity problem, bar plots of the results for two manually selected clients from the Bosphorus and Wilches databases are depicted in Figure 8.3. The bar plots depict the results for each of the selected clients over the three inner cross-validation folds (see Section 7.4), where all three inner folds are associated with (1) the *same* trained top performing systems and (2) a different random selection of positive and negative *test* pairs. Four randomly selected images for each of the four manually selected clients are depicted in Figure 8.4 for reference. The bar plots on the left of Figure 8.3 are associated with clients from the Bosphorus and Wilches databases with *low* deformity (first and third row of images in Figure 8.4), while the bar plots on the right are associated with clients from the Bosphorus and Wilches databases with *high* deformity (second and fourth row of images in Figure 8.4).

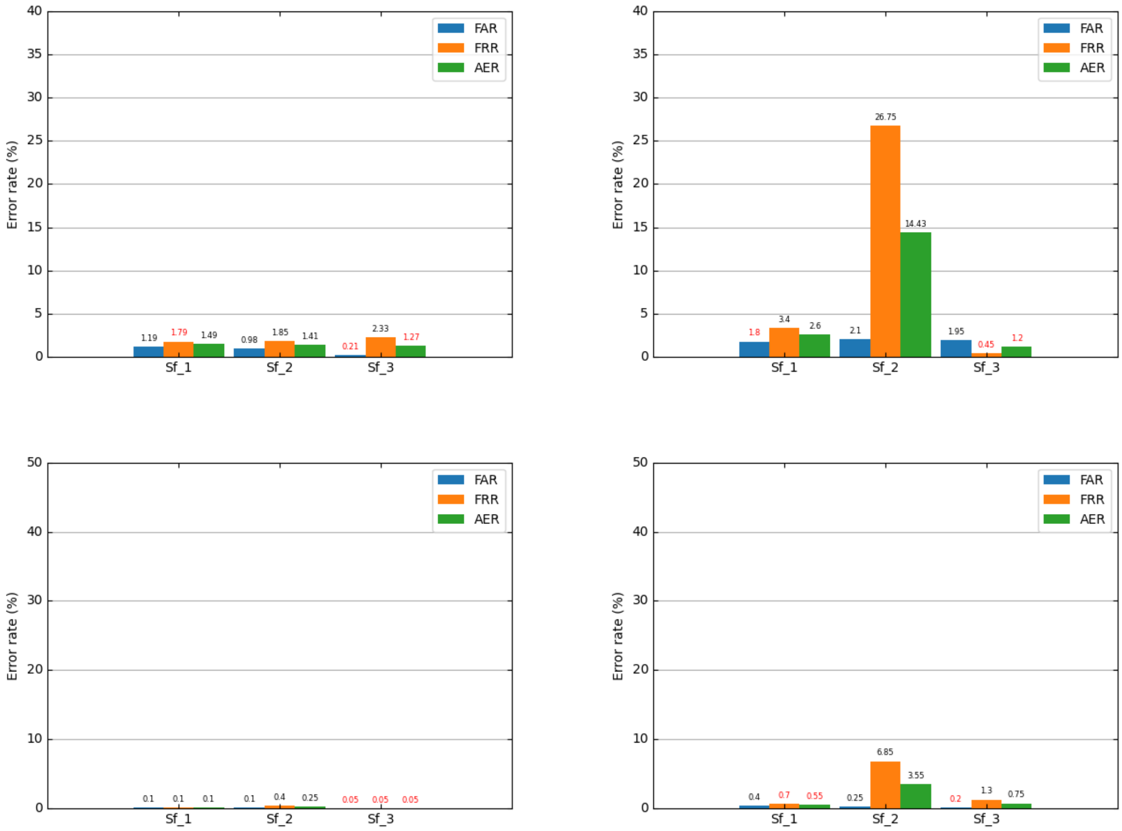


Figure 8.3: The results of the top performing systems for each of the three inner cross-validation folds associated with two manually selected clients from the Bosphorus and Wilches databases respectively. **(Left)** Clients from the Bosphorus (top) and Wilches (bottom) databases with *low* deformity (first and third row of images in Figure 8.4). **(Right)** Clients from the Bosphorus (top) and Wilches (bottom) databases with *high* deformity (second and fourth row of images in Figure 8.4).

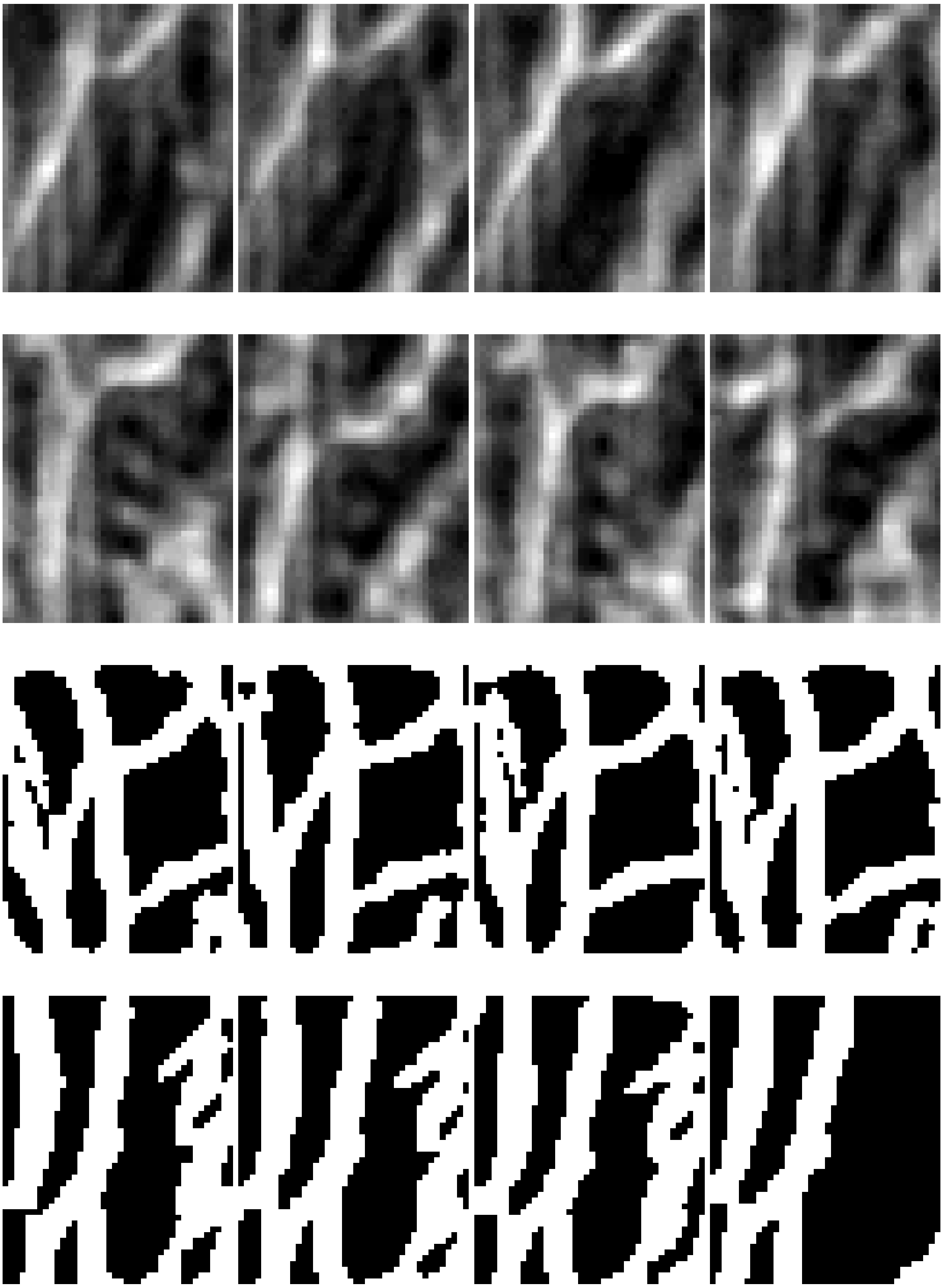


Figure 8.4: Four randomly selected images for four manually selected clients in order to illustrate the extent of the deformity problem within the context of the IDE. **(First row)** Images of the client corresponding to the the bar plot on the top left of Figure 8.3. **(Second row)** Images of the client corresponding to the bar plot on the top right of Figure 8.3. **(Third row)** Images of the client corresponding to the the bar plot on the bottom left of Figure 8.3. **(Final row)** Images of the client corresponding to the the bar plot on the bottom right of Figure 8.3.

It is clear from Figures 8.3 and 8.4 that the systems associated with clients with low deformity across images perform significantly more consistently across the three inner cross-validation folds than those systems associated with clients with high deformity across images. It is therefore conceivable that the proficiency of the proposed systems within the context of the IDS may be further enhanced by granting an individual a total of three attempts for authentication during testing.

The results obtained when gauging the proficiency of the proposed systems within the context of the IIS are presented and discussed in the next section.

8.6 Results: individual independent scenario

The purpose of the individual independent scenario (IIS) is to simulate a real-world scenario in which a *single* network is trained in a *once-off* fashion *prior* to the enrolment of *any* clients. This is a specific scenario of training a so-called *general* similarity measure network (SMN) [23], where the SMN is trained and tested on *mutually exclusive* sets of objects.

In order to accomplish the aforementioned goal within the context of hand vein-based biometric authentication, the hand vein samples of a large group of individuals that is representative of the population need to be acquired *prior* to the start of training, which is a time-consuming task. One way to overcome this problem is to utilise existing hand vein databases. The majority of existing, publicly accessible hand vein databases however have been acquired *only* for the purpose of experimentation, and do *not* contain a representative subset of the population.

The hand vein databases employed in this study, namely the Bosphorus and Wilches databases, are each associated with *only* 100 individuals. In order to train a general SMN, these individuals need to be partitioned into 6 sets of *mutually exclusive* individuals, namely

1. training clients,
2. training imposters,
3. validation clients,
4. validation imposters,
5. test clients and
6. test imposters.

The partitioning of these 100 individuals invariably results in a set of training individuals that does *not* constitute a representative subset of the population (see Section 7.5.1). The results presented in Section 8.6.1 clearly indicate that training a *general* SMN on an *insufficient* number of *different* individuals that is able to achieve SOTA-level proficiency is infeasible.

The first solution to the aforementioned problem proposed in this study is that of the individual dependent scenario (IDS), in which a tailor-made network is trained for each client enrolled into the system. The results presented in Section 8.5 clearly show that the IDS is a viable alternative to the IIS when an insufficient number of individuals are available for training a general SMN. The IDS however has two main disadvantages when compared to a general SMN, namely the fact that (1) the management and (2) the storage of potentially millions of networks can be challenging in a real world scenario.

The GenVeins database (see Chapter 4) is therefore proposed in this study as another viable solution to the aforementioned problem. The utilisation of the GenVeins database for the purpose of obtaining four *mutually exclusive* sets of individuals, that is (1) training clients, (2) training imposters, (3) validation clients and (4) validation imposters, each of which contains a *sufficient* number of *different* individuals so as to be representative samples of the population, provides the following two very important benefits:

1. It eliminates the need for acquiring hand vein samples of *any* actual individuals prior to training *and* validation.

2. Networks trained and validated on the GenVeins database are *robust* and *consistent* across *multiple* actual hand vein acquisition protocols, provided that these acquisition protocols are sufficiently controlled.

The results of the three experiments conducted within the context of the IIS, that is the benchmark experiment (BME), the first GenVeins experiment (FGE) and the second GenVeins experiment (SGE) (see Section 7.1) are presented in Sections 8.6.1, 8.6.2 and 8.6.3 respectively in the following fashion:

1. The results of the top 20 systems are first presented and discussed.

2. The SHAP values for the top performing systems are subsequently presented and discussed.

3. The extent of the deformity problem is finally discussed.

It is important to note that the number of parameters of the CNN-based FEs (see Figures 6.2, 6.3, 6.4 and 6.5) are multiplied by 4 within the context of the IIS, in order to account for the increased complexity of the optimisation function. The results obtained within the context of the IDS and IIS should therefore not be explicitly compared.

8.6.1 The benchmark experiment

The results for only the top 20 systems are presented and discussed in this section within the context of the Bosphorus and Wilches databases and the BME. The full results are presented in the appendix of this dissertation (see Section A.3). The results of the top 20 systems within the context of the Bosphorus and Wilches databases and the BME are depicted in Tables 8.4 and 8.5 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Table 8.4: The results of the top 20 systems ranked according to the AER within the context of the BME and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH BN NO EER	20.471	18.086	19.278	79.529	81.914	5.221
2	2CH BD NO EER	18.912	20.139	19.525	81.088	79.861	3.48
3	2CH BN NO AER_{min}	18.017	21.281	19.649	81.983	78.719	5.068

4	2CH BD NO AER_{min}	16.775	22.562	19.668	83.225	77.438	3.952
5	2CH BN CL EER	20.424	21.713	21.069	79.576	78.287	4.863
6	2CH BD CL EER	23.434	18.943	21.188	76.566	81.057	3.288
7	2CH BN CL AER_{min}	16.89	25.779	21.335	83.11	74.221	4.678
8	2CH BD CL AER_{min}	22.978	20.478	21.729	77.022	79.522	3.468
9	2CH ST NO EER	26.906	19.498	23.202	73.094	80.502	3.434
10	2CH ST NO AER_{min}	24.228	22.84	23.534	75.772	77.16	3.696
11	2CH ST CL EER	19.406	28.719	24.063	80.594	71.281	3.678
12	2CH ST CL AER_{min}	19.221	28.974	24.097	80.779	71.026	3.668
13	SM BD NO AER_{min}	25.131	24.645	24.888	74.869	75.355	3.208
14	2CH BD FB EER	26.844	24.56	25.702	73.156	75.44	4.682
15	SM BD NO EER	22.315	29.275	25.795	77.685	70.725	3.652
16	2CH BD FB AER_{min}	24.066	28.04	26.053	75.934	71.96	5.23
17	SM DO NO EER	10.069	44.151	27.11	89.931	55.849	2.25
18	SM DO NO AER_{min}	10.069	44.151	27.11	89.931	55.849	2.25
19	2CH BN FB EER	26.582	29.051	27.816	73.418	70.949	5.095
20	2CH BN FB AER_{min}	25.324	30.872	28.098	74.676	69.128	4.717

Table 8.5: The results of the top 20 systems ranked according to the AER within the context of the BME and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, pre-processing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH BD CL EER	6.181	4.792	5.486	93.819	95.208	2.036
2	2CH BD CL AER_{min}	5.556	5.903	5.729	94.444	94.097	1.718
3	2CH BD NO AER_{min}	5.764	5.903	5.833	94.236	94.097	1.593
4	2CH BD NO EER	8.472	5.208	6.84	91.528	94.792	3.345
5	2CH BN CL AER_{min}	4.236	9.583	6.91	95.764	90.417	2.236
6	2CH ST NO AER_{min}	6.875	7.014	6.945	93.125	92.986	3.313
7	2CH BN CL EER	6.319	7.639	6.979	93.681	92.361	2.165
8	2CH ST NO EER	8.056	6.042	7.049	91.944	93.958	3.405
9	2CH BN NO EER	5.556	9.444	7.5	94.444	90.556	3.115
10	2CH BD FB EER	11.181	3.958	7.569	88.819	96.042	3.556
11	2CH BD FB AER_{min}	11.042	5.625	8.333	88.958	94.375	4.538
12	2CH BN NO AER_{min}	4.861	12.222	8.542	95.139	87.778	3.697
13	2CH BN FB EER	9.167	9.792	9.479	90.833	90.208	4.893
14	2CH BN FB AER_{min}	7.847	11.181	9.514	92.153	88.819	4.375

15	2CH DO NO EER	12.292	9.375	10.833	87.708	90.625	4.658
16	2CH ST CL EER	10.139	12.014	11.076	89.861	87.986	1.539
17	SM DO CL EER	5.694	16.944	11.319	94.306	83.056	4.218
18	SM DO CL AER_{min}	5.694	16.944	11.319	94.306	83.056	4.218
19	2CH DO NO AER_{min}	11.319	12.986	12.153	88.681	87.014	4.738
20	2CH ST CL AER_{min}	11.181	13.264	12.222	88.819	86.736	2.331

The following noteworthy conclusions may be drawn from the results depicted in Tables 8.4 and 8.5 within the context of the BME and the Bosphorus and Wilches databases respectively:

1. When the ranking of the preprocessing protocols is considered, it is clear that the proposed systems tend to prefer images with additional background information (no preprocessing and CLAHE preprocessing) as opposed to images with no background information (full binarisation) within the context of the BME. This may be due to the increased complexity of the optimisation function at hand, which involves learning a *general* similarity function that is able to perform well on *unseen* individuals. In other words, the systems tend to become significantly more biased on the training individuals when trained on images containing *only* information about the hand veins.

2. The majority of networks employing the AER_{min} and EER threshold selection criteria significantly outperform the systems in which either the FAR_{zero} and FRR_{zero} threshold selection criteria (see Tables A.4 and A.5) are employed. This indicates that the systems benefit from employing a more balanced threshold selection criterion as opposed to employing either a very strict or very lenient criterion within the context of the BME.

3. The differences between the top performing systems and the 20th ranked system depicted in Tables 8.4 and 8.5 are relatively large, which indicates that the proposed systems are highly sensitive to sub-optimal design within the context of the BME.

4. The standard deviations of the AERs are relatively small within the context of *both* databases, which indicates that the proposed systems are not overly sensitive to different random data partitions. This subsequently strengthens the notion that the employed cross-validation protocol, namely RRS (see Section 7.2), is successful in minimising the probability of bias for any one random data partition.

When the results depicted in Tables 8.4 and 8.5 are compared, it is clear that the majority of the proposed systems perform better on the Wilches database, as is the case within the context of the IDE.

The SHAP values (see Section 8.3) for a selected number of clients and test pairs within the context of the BME are depicted in Figure 8.5. The SHAP values are calculated for the CNN-based FEs (see Section 6.2) of the top performing systems depicted in Tables 8.4 and 8.5 within the context of the Bosphorus and Wilches databases respectively. Recall from Section 8.3.2 that red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question. It is also important to note the following in order to interpret Figure 8.5.

1. **The two principal images framed in black:**
   (a) The left principal image is associated with four clients from the Bosphorus database, while the right principal image is associated with four clients from the Wilches database.
2. **The four columns of subimages depicted in each of the two principal images framed in black:**
   (a) The first column of subimages depict the *reference* samples, while the second column of subimages depict the *questioned* samples. The first and second subimages in each of the four rows of images therefore constitute the test pairs in question. The test pairs are ordered in an alternating fashion, starting with a positive pair.
   (b) The third column of subimages depict the SHAP values of the corresponding test pair within the context of the *positive* class, while the fourth column of subimages depict the SHAP values of the corresponding test pair within the context of the *negative* class. It is therefore desired to have overwhelmingly red values in the *third* column of subimages for the *positive* test pairs (first and third rows of test pairs) and overwhelmingly red values in the *fourth* column of subimages for the *negative* test pairs (second and fourth rows of test pairs). It is important to note that, since a two-channel image cannot be displayed as an image, *only* the *questioned* sample (second column of subimages) is depicted in the third and fourth column of subimages in order to gain insight into the evidence that the *questioned* sample belongs to the positive or negative class respectively.

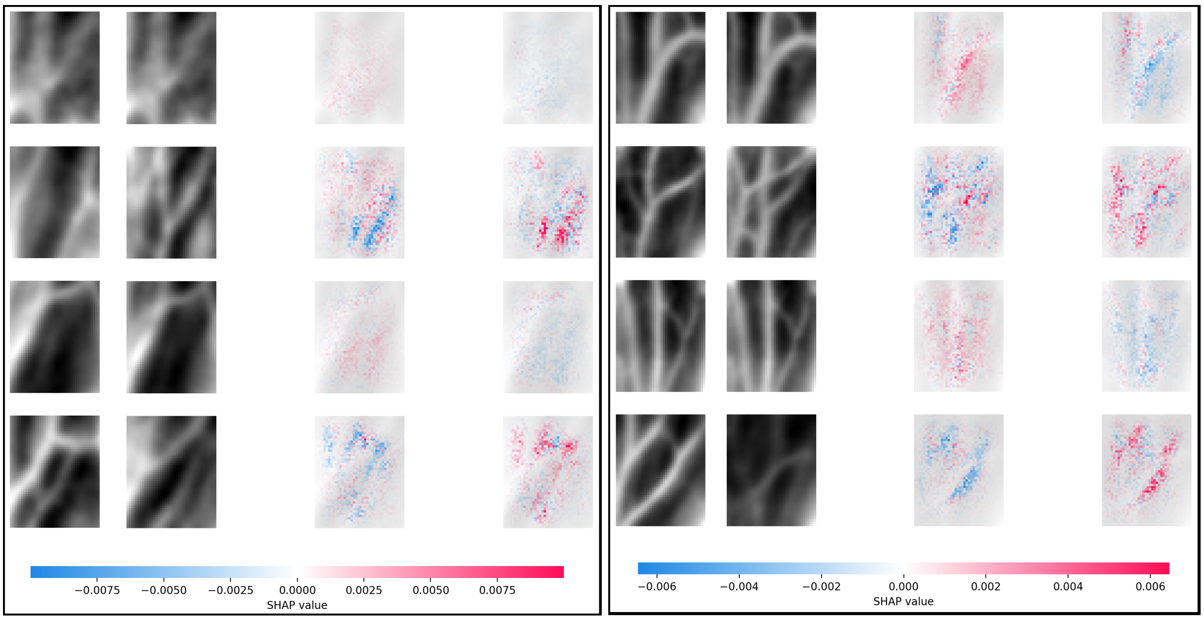


Figure 8.5: The SHAP values (see Section 8.3) for a selected number of clients and test pairs within the context of the Bosphorus (left) and Wilches (right) databases and the BME. Red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question.

When the images depicted in Figure 8.5 are considered, it is clear that the SHAP values are neither overwhelmingly positive nor overwhelmingly negative for the majority of test pairs within the context of the BME. This also strengthens the notion that the proficiency of the proposed CNN-based FEs is negatively impacted by training on a non-representative set of individuals, since the networks are not overwhelmingly confident about the true class labels of the majority of test pairs that belong to *unseen* individuals. This poses a security risk, and is therefore not a feasible application in any real world scenario.

It is more challenging to determine the extent of the deformity problem within the context of the IIS, since the systems are trained and tested on *sets* of *multiple* individuals. The proposed solution is to gauge the consistency of the top performing systems within the context of the BME and the Bosphorus and Wilches databases (see Tables 8.4 and 8.5) over the nine cross-validation folds (see Section 7.5). Higher consistency between the results of the nine cross-validation folds corresponds to a smaller impact of the deformity problem. The aforementioned results are depicted on the left of Figure 8.6 for the Bosphorus database, and on the right of Figure 8.6 for the Wilches database.

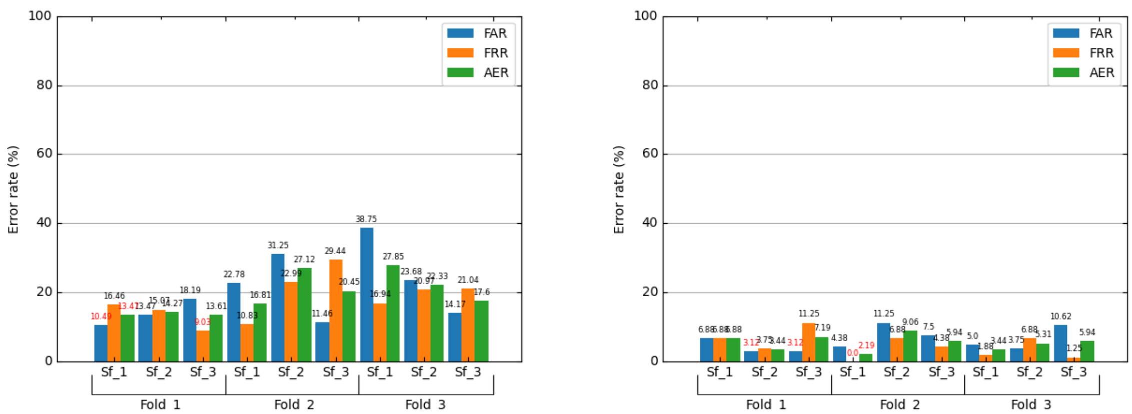


Figure 8.6: The results of the nine cross validation folds associated with the top performing systems within the context of the BME and the Bosphorus (left) and Wilches (right) databases respectively.

It is clear from Figure 8.6 that the consistency of the top performing systems over the nine cross validation folds is poor for *both* databases within the context of the BME. This is due to the fact that the proposed systems are trained on a non-representative set of individuals during the BME, which invariably leads to poor generalisation.

8.6.2 The first GenVeins experiment

The results for only the top 20 systems are presented and discussed in this section within the context of the Bosphorus and Wilches databases and the FGE. The full results are presented in the appendix of this dissertation (see Section A.5). The results of the top 20 ranked systems within the context of the Bosphorus and Wilches databases and the FGE are depicted in Tables 8.6 and 8.7 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Table 8.6: The results of the top 20 systems ranked according to the AER within the context of the FGE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, pre-processing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH DO NO AER_{min}	4.542	25.669	15.105	95.458	74.331	0.679
2	2CH ST NO AER_{min}	7.862	22.64	15.251	92.138	77.36	0.757
3	2CH BD CL AER_{min}	9.809	21.336	15.573	90.191	78.664	0.698
4	2CH DO NO FRR_{zero}	15.219	16.656	15.937	84.781	83.344	2.754
5	2CH BD CL EER	5.216	26.818	16.017	94.784	73.182	0.916
6	2CH ST NO EER	4.151	28.012	16.082	95.849	71.988	0.79
7	2CH BN CL EER	13.345	18.955	16.15	86.655	81.045	0.631
8	2CH BN CL AER_{min}	20.858	14.831	17.844	79.142	85.169	1.236
9	2CH BD FB FRR_{zero}	5.697	30.553	18.125	94.303	69.447	0.79
10	2CH DO CL FRR_{zero}	2.099	34.194	18.146	97.901	65.806	1.823
11	2CH BD CL FRR_{zero}	21.861	14.618	18.239	78.139	85.382	3.199
12	2CH DO NO EER	1.431	35.836	18.633	98.569	64.164	1.074
13	2CH ST CL FRR_{zero}	14.377	25.401	19.889	85.623	74.599	10.672
14	2CH DO CL AER_{min}	1.075	38.917	19.996	98.925	61.083	1.27
15	2CH ST NO FRR_{zero}	28.222	12.843	20.532	71.778	87.157	10.72
16	2CH ST CL AER_{min}	0.908	41.48	21.194	99.092	58.52	1.752
17	2CH BN FB AER_{min}	1.849	40.963	21.406	98.151	59.037	1.129
18	2CH BD FB AER_{min}	1.294	43.789	22.541	98.706	56.211	1.661
19	2CH BN FB EER	1.218	44.789	23.004	98.782	55.211	0.759
20	SM DO NO EER	26.527	20.088	23.307	73.473	79.912	2.492

Table 8.7: The results of the top 20 systems ranked according to the AER within the context of the FGE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, pre-processing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH ST NO EER	1.885	6.657	4.271	98.115	93.343	0.447
2	2CH DO NO AER_{min}	2.907	5.893	4.4	97.093	94.107	0.751
3	2CH ST NO AER_{min}	4.177	4.782	4.479	95.823	95.218	0.645
4	2CH DO NO EER	0.823	9.018	4.921	99.177	90.982	0.798
5	2CH DO FB AER_{min}	2.202	8.105	5.154	97.798	91.895	0.51
6	2CH BN FB EER	2.996	7.55	5.273	97.004	92.45	0.596

7	2CH BD FB AER_{min}	2.708	7.956	5.332	97.292	92.044	0.436
8	2CH BN FB AER_{min}	4.434	6.359	5.397	95.566	93.641	0.631
9	2CH DO FB EER	1.24	9.633	5.437	98.76	90.367	0.697
10	2CH BD FB EER	1.567	9.712	5.64	98.433	90.288	0.564
11	2CH ST FB AER_{min}	4.206	7.302	5.754	95.794	92.698	0.583
12	2CH ST FB EER	3.403	8.234	5.818	96.597	91.766	0.438
13	2CH DO CL EER	8.661	3.115	5.888	91.339	96.885	0.774
14	2CH DO NO FRR_{zero}	11.032	3.294	7.163	88.968	96.706	4.036
15	2CH BD CL EER	12.55	2.976	7.763	87.45	97.024	0.549
16	2CH ST CL EER	12.946	2.827	7.887	87.054	97.173	1.757
17	2CH BD FB FRR_{zero}	12.688	3.482	8.085	87.312	96.518	1.853
18	2CH DO CL AER_{min}	15.873	2.113	8.993	84.127	97.887	2.094
19	SM DO CL EER	8.829	9.385	9.107	91.171	90.615	1.156
20	SM DO CL AER_{min}	8.829	9.385	9.107	91.171	90.615	1.156

The following noteworthy conclusions may be drawn from the results depicted in Tables 8.6 and 8.7 within the context of the FGE and the Bosphorus and Wilches databases respectively:

1. The majority of systems in which either the AER_{min} or EER threshold selection criterion are employed significantly outperform the systems in which either the FAR_{zero} or FRR_{zero} threshold selection criterion (see Tables A.8 and A.9) are employed, as is the case within the context of the BME.

2. The differences between the top performing systems and the 20th ranked system depicted in Tables 8.6 and 8.7 are relatively large, as is the case within the context of the BME.

3. The FRRs are significantly larger within the context of the Bosphorus database, which indicates that the so-called "vein-deformity" problem is much more severe for said database.

4. The standard deviations of the AERs are relatively small within the context of *both* databases, which indicates that the proposed systems are not overly sensitive to different random data partitions. This subsequently strengthens the notion that the employed cross-validation protocol, namely RRS (see Section 7.2), is successful in minimising the probability of bias for any one random data partition. Is is also important to note that the STDs within the context of the FGE are *significantly* smaller when compared to the STDs within the context of the BME, which further strengthens the hypothesis that the utilisation of the GenVeins database for the purpose of training and validating the proposed systems within the context of the IIS *significantly* increases their generalisation potential when presented with images from unseen individuals.

When the results depicted in Tables 8.6 and 8.7 are compared, it is clear that the majority of the proposed systems perform better within the context of the Wilches database, as is the case within the context of the BME.

The SHAP values for a selected number of clients and test pairs within the context of the FGE are depicted in Figure 8.7. The interpretation of these images is similar to the interpretation of the images depicted in Figure 8.5.

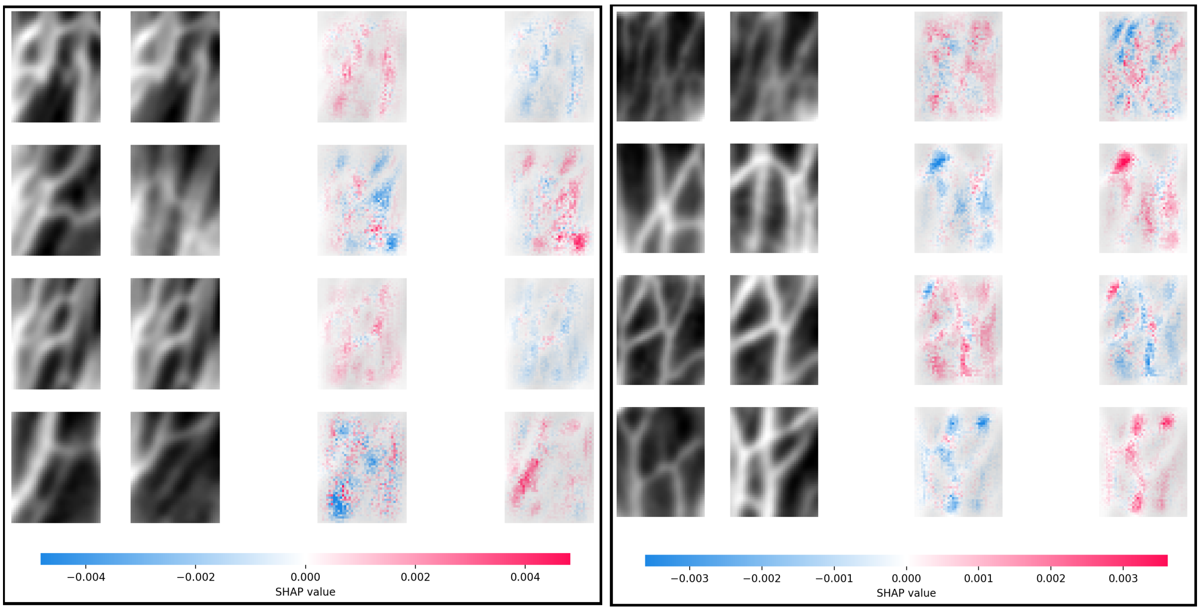


Figure 8.7: The SHAP values (see Section 8.3) for a selected number of clients and test pairs within the context of the Bosphorus (left) and Wilches (right) databases and the FGE. Red SHAP values indicate *positive* evidence that the questioned image belongs the class in question, while blue SHAP values indicate *negative* evidence that the questioned image belongs to the class in question.

When the images depicted in Figure 8.7 are considered, it is clear that the SHAP values are stronger when compared to the SHAP values depicted in Figure 8.5. This indicates firstly that the proposed systems benefit directly from the utilisation of the GenVeins database for training and validation, which in turn shows that the GenVeins database is a *feasible* solution to the problem of the unavailability of a set of training individuals that is large enough so as to be representative of the population. The stronger SHAP values furthermore indicate that the utilisation of the GenVeins database for the purpose of training and validating the proposed systems mitigates the deformity problem to a significant extent, since the hand vein samples of the fictitious individuals in the GenVeins database are generated under sufficiently controlled conditions, which in turn enables the proposed systems to deliver predictions with increased confidence.

In order to furthermore strengthen the notion that the GenVeins database is able to significantly reduce the effect of the deformity problem, the results of the nine cross-validation folds within the context of the FGE are depicted on the left of Figure 8.8 for the Bosphorus database, and on the right of Figure 8.8 for the Wilches database.

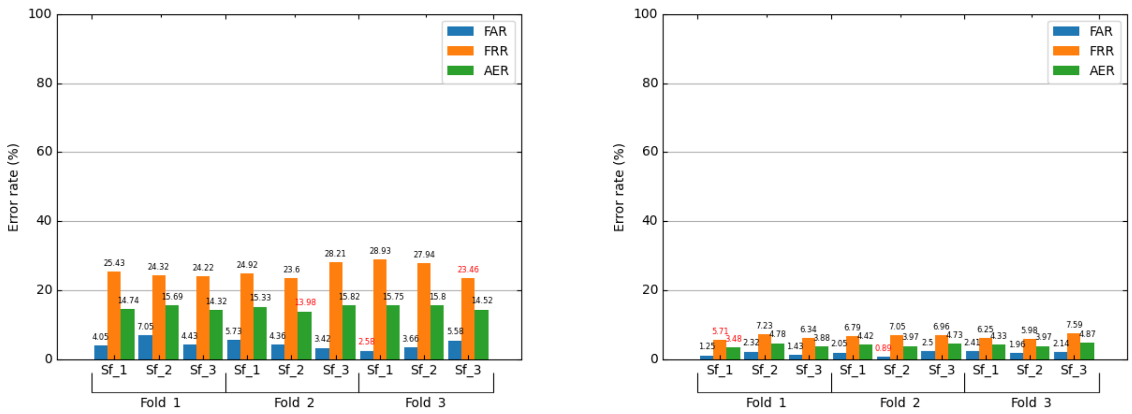


Figure 8.8: The results of the nine cross-validation folds associated with the top performing system within the context of the FGE and the Bosphorus (left) and Wilches (right) databases respectively.

When the results depicted in Figure 8.8 are compared to the results depicted in Figure 8.6, it is clear that the proficiency *and* consistency of the proposed systems are significantly increased by training on the GenVeins database, as opposed to training on a non-representative set of actual individuals.

8.6.3 The second GenVeins experiment

The results for only the top 20 systems are presented and discussed in this section within the context of the Bosphorus and Wilches databases and the SGE. The full results are presented in the appendix of this dissertation (see Section A.6). It is important to note that an individual is granted a total of three attempts at authentication within the context of the SGE. The results of the top 20 ranked systems within the context of the Bosphorus and Wilches databases and the SGE are depicted in Tables 8.8 and 8.9 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Table 8.8: The results of the top 20 systems ranked according to the AER within the context of the SGE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	2CH DO CL AER_{min}	4.462	9.593	7.028	95.538	90.407	0.398
2	2CH ST CL AER_{min}	4.439	10.291	7.365	95.561	89.709	0.546
3	2CH DO CL EER	2.12	13.453	7.786	97.88	86.547	0.77
4	2CH DO NO EER	9.978	5.655	7.816	90.022	94.345	1.438
5	2CH ST CL EER	2.403	13.442	7.923	97.597	86.558	0.771
6	2CH BN FB AER_{min}	5.074	12.52	8.797	94.926	87.48	0.689
7	2CH BD FB AER_{min}	4.125	13.542	8.833	95.875	86.458	0.684

8	2CH BN FB EER	3.266	15.139	9.203	96.734	84.861	0.841
9	2CH BD FB EER	2.448	16.194	9.321	97.552	83.806	0.68
10	2CH DO FB AER_{min}	2.609	17.03	9.82	97.391	82.97	0.899
11	2CH ST FB AER_{min}	3.299	16.66	9.979	96.701	83.34	0.817
12	2CH ST FB EER	2.128	19.372	10.75	97.872	80.628	1.138
13	2CH DO FB EER	1.565	20.268	10.916	98.435	79.732	1.02
14	2CH ST NO EER	18.628	3.548	11.088	81.372	96.452	2.425
15	2CH BD FB FRR_{zero}	18.611	6.22	12.416	81.389	93.78	2.421
16	2CH BD CL EER	21.473	4.087	12.78	78.527	95.913	1.756
17	2CH DO NO AER_{min}	22.247	3.317	12.782	77.753	96.683	3.661
18	2CH ST NO AER_{min}	26.528	2.563	14.545	73.472	97.437	2.854
19	2CH DO CL FRR_{zero}	25.256	4.57	14.913	74.744	95.43	12.599
20	2CH ST CL FRR_{zero}	25.238	4.891	15.065	74.762	95.109	12.683

Table 8.9: The results of the top 20 systems ranked according to the AER within the context of the SGE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, pre-processing protocol and probabilistic threshold criterion. Refer to Table 8.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	2CH DO NO EER	3.228	0.0	1.614	96.772	100.0	0.751
2	2CH ST FB EER	4.074	0.0	2.037	95.926	100.0	1.215
3	2CH DO FB EER	5.026	0.0	2.513	94.974	100.0	1.425
4	2CH BD FB EER	6.138	0.0	3.069	93.862	100.0	1.461
5	2CH BN FB EER	6.667	0.0	3.333	93.333	100.0	1.728
6	2CH ST NO EER	6.984	0.0	3.492	93.016	100.0	0.761
7	2CH ST FB AER_{min}	7.09	0.0	3.545	92.91	100.0	1.613
8	2CH DO FB AER_{min}	7.936	0.0	3.968	92.064	100.0	1.796
9	SM BN CL EER	8.73	0.0	4.365	91.27	100.0	3.389
10	2CH DO NO AER_{min}	9.206	0.0	4.603	90.794	100.0	2.772
11	2CH BD FB AER_{min}	9.788	0.0	4.894	90.212	100.0	2.459
12	2CH BN FB AER_{min}	10.159	0.0	5.079	89.841	100.0	1.71
13	2CH ST NO AER_{min}	11.587	0.0	5.794	88.413	100.0	1.285
14	SM DO CL EER	12.699	0.0	6.349	87.301	100.0	2.564
15	SM BD NO EER	12.857	0.0	6.429	87.143	100.0	1.928
16	SM DO CL AER_{min}	13.386	0.0	6.693	86.614	100.0	2.84
17	SM BD FB EER	15.132	0.0	7.566	84.868	100.0	2.939
18	SM BD CL EER	15.979	0.0	7.989	84.021	100.0	2.15

19	SM DO FB EER	17.354	0.0	8.677	82.646	100.0	3.358
20	SM BN CL AER_{min}	17.831	0.0	8.915	82.169	100.0	4.129

The following noteworthy conclusions may be drawn from the results depicted in Tables 8.8 and 8.9 within the context of the SGE and the Bosphorus and Wilches databases respectively:

1. When the ranking of the preprocessing protocols is considered, it is clear that the systems in which either the full binarisation preprocessing protocol or the CLAHE preprocessing protocol is employed rank among the top performing systems of both databases. This indicates that the utilisation of the GenVeins database, in which the acquisition protocol is sufficiently controlled, enables the proposed systems to sufficiently distinguish between positive and negative test pairs in which the hand vein information is enhanced, as opposed to applying no enhancement.

2. The majority of systems in which either the AER_{min} or EER threshold selection criterion are employed significantly outperform the systems in which either the FAR_{zero} or FRR_{zero} threshold selection criterion (see Tables A.10 and A.11) are employed. This is also the case within the context of the BME and the FGE.

3. The differences between the top performing systems and the 20th ranked system depicted in Tables 8.8 and 8.9 are relatively large as is the case within the context of the BME and FGE.

4. An interesting observation from the results depicted in Table 8.9 is that the FRR is zero for *all* of the top 20 ranked systems, which is not the case within the context of the Bosphorus database (see Table 8.8). This strengthens the notion that the extent of the deformity problem is much more severe within the context of the Bosphorus database, due to the fact that these images were acquired under insufficiently controlled conditions.

When the results depicted in Tables 8.8 and 8.9 are compared, it is clear that the majority of the proposed systems perform better within the context of the Wilches database, as is the case within the context of the BME and the FGE.

The SHAP values within the context of the SGE provide no additional measure of the proficiency and consistency of the proposed CNN-based FEs, since the training and validation protocols are identical to the FGE, and is therefore not presented in this section.

The results of the nine cross-validation folds associated with the top performing system within the context of the SGE are depicted on the left of Figure 8.9 for the Bosphorus database, and on the right of Figure 8.9 for the Wilches database.

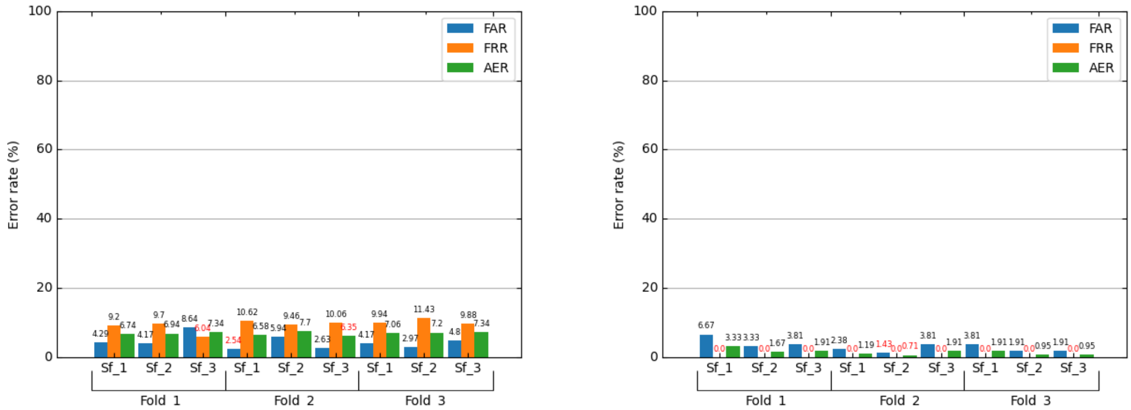


Figure 8.9: The results of the nine cross-validation folds associated with the top performing system within the context of the SGE and the Bosphorus (left) and Wilches (right) databases respectively.

When the results depicted in Figure 8.9 are compared to the results depicted in Figure 8.8, it is clear that the AERs of the proposed systems are more than halved by simply granting an individual a total of three chances to authenticate during testing, as opposed to only one chance. The results of the SGE therefore motivates the feasibility of this approach for the purpose of effectively mitigating the negative impact of poor-quality test samples.

The statistical comparisons between the results of (1) the BME and the augmented BME (see Appendix A.4) and (2) the BME and the FGE are presented and discussed in the following section.

8.7 Statistical comparison of results

The statistical comparisons between the results of (1) the BME and the augmented BME (see Appendix A.4) and (2) the BME and the FGE are presented and discussed in this section. The paired t-test [69] is employed for the purpose of determining whether or not the proficiency of the proposed systems is (1) better within the context of the BME than within the context of the augmented BME and (2) better within the context of the FGE than within the context of the BME. The paired t-test accomplishes the aforementioned goal by statistically evaluating the underlying distributions from which the two sets of scores are produced. The details regarding the paired t-tests conducted in this study are provided below.

- The “subjects” of the paired t-tests comprise of the 96 systems proposed in this study.

- The “scores” of the subjects constitute the average of the AERs over the 9 iterations of cross-validation (see Section 7.2) for *all* 96 systems within the context of a specific experimental scenario and a specific *test* database. In other words, the paired t-tests are conducted by employing the AER column of the tables depicting the full results presented in Appendix A for the experimental scenarios and databases in question. It is important to note that, within the context of a *paired* t-test, the scores are *not* ordered by the AER in an ascending fashion, since the paired t-test requires two different scores of the *same* system design to be compared. A useful analogy to illustrate the aforementioned explanation constitutes the scores of a number of students in two different exams. The students in the aforementioned analogy represents the specific system designs, while the two different exams represent the two different experimental scenarios in question.

- The paired t-tests are conducted in a "one-sided" fashion, which comprises of testing the hypotheses that the *first* set of AERs is statistically less than the *second* set of AERs.
- A total of four paired t-tests are therefore conducted in this study, which are summarised below:
  1. Test the null hypothesis that the distribution which produced the scores of the proposed systems within the context of the BME is *identical* to the distribution which produced the scores of the proposed systems within the context of the augmented BME when considering test samples from the Bosphorus database.
  2. Test the null hypothesis that the distribution which produced the scores of the proposed systems within the context of the BME is *identical* to the distribution which produced the scores of the proposed systems within the context of the augmented BME when considering test samples from the Wilches database.
  3. Test the null hypothesis that the distribution which produced the scores of the proposed systems within the context of the FGE is *identical* to the distribution which produced the scores of the proposed systems within the context of the BME when considering test samples from the Bosphorus database.
  4. Test the null hypothesis that the distribution which produced the scores of the proposed systems within the context of the FGE is *identical* to the distribution which produced the scores of the proposed systems within the context of the BME when considering test samples from the Bosphorus database.

The paired t-tests conducted within the context of the BME and the augmented BME are presented in Section 8.7.1, while the paired t-tests conducted within the context of the BME and the FGE are presented in Section 8.7.2.

8.7.1 The BME vs the augmented BME

The paired t-tests conducted in order to establish whether or not the proficiency of the proposed systems is statistically *better* within the context of the BME than within the context of the augmented BME are presented in this section. For reference, the scores of the first 10 systems within the context of the BME and the augmented BME when employing *test* samples from the Bosphorus database are depicted in Table 8.10, while the scores of the first 10 systems within the context of the BME and the augmented BME when employing *test* samples from the Wilches database are depicted in Table 8.11. It is important to note that, while only the scores of the first 10 systems are depicted in Tables 8.10 and 8.11, the scores of *all* 96 systems are employed for the purpose of conducting the paired t-test.

Table 8.10: The average of the AERs over the 9 iterations of cross-validation of the first 10 systems within the context of the BME and the augmented BME when employing test samples from the Bosphorus database.

System Design	AER: BME	AER: Augmented BME
2CH NO BN EER	19.27844	24.14111

2CH NO BD EER	19.52544	25.33333
2CH NO BN AER_{min}	19.64889	24.36333
2CH NO BD AER_{min}	19.66811	25.35333
2CH CL BN EER	21.06856	26.22889
2CH CL BD EER	21.18833	26.90889
2CH CL BN AER_{min}	21.33478	26.53111
2CH CL BD AER_{min}	21.72856	27.57111
2CH NO ST EER	23.20222	29.72556
2CH NO ST AER_{min}	23.53389	29.86667

Table 8.11: The average of the AERs over the 9 iterations of cross-validation of the first 10 systems within the context of the BME and the augmented BME when employing test samples from the Wilches database.

System Design	**AER: BME**	**AER: Augmented BME**
2CH CL BD EER	5.48633	9.73889
2CH CL BD AER_{min}	5.729	9.80889
2CH NO BD AER_{min}	5.83344	10.21111
2CH NO BD EER	6.84044	10.25222
2CH CL BN AER_{min}	6.90978	12.09333
2CH NO ST AER_{min}	6.94456	15.20444
2CH CL BN EER	6.97922	12.12
2CH NO ST EER	7.04856	15.17889
2CH NO BN EER	7.5	10.69778
2CH FB BD EER	7.56933	14.89667

The results of the one-sided paired t-tests between the scores of the proposed systems within the context of the BME and the augmented BME are presented in Table 8.12.

Table 8.12: The results of the one-sided paired t-tests between the scores of the proposed systems within the context of the BME and the augmented BME.

Test database	**t-statistic**	**p-value**
Bosphorus	-0.13322	0.44715
Wilches	1.02855	0.84685

If the null hypothesis may only be *rejected* if the p-value is less than 0.05, it is clear from the results depicted in Table 8.12 that *no* conclusion can be reached about whether or not the proficiency of the proposed systems is *statistically* less within the context of the BME than within the context of the augmented BME. In other words, it is unclear from the paired t-test between the scores of the proposed systems within the context of the BME and the augmented BME whether or not augmentation is beneficial. It is however important to note that, by considering the robust cross-validation protocol employed in this study, namely RRS (see Section 7.2, it is still possible to *visually* compare the scores of the proposed systems between the two aforementioned experiments, and by doing so, it is clear that the proficiency of the proposed systems does appear to be negatively impacted by employing the proposed data augmentation protocol (see Section 7.3). Further investigations are however warranted based on the outcome of the results of the paired t-tests depicted in Table 8.12 in order to statistically determine whether or not augmentation is beneficial within the context of the BME.

8.7.2 The FGE vs the BME

The paired t-tests conducted in order to establish whether or not the proficiency of the proposed systems is statistically *better* within the context of the FGE than within the context of the BME are presented in this section. For reference, the scores of the first 10 systems within the context of the FGE and the BME when employing *test* samples from the Bosphorus database are depicted in Table 8.13, while the scores of the first 10 systems within the context of the FGE and the BME when employing *test* samples from the Wilches database are depicted in Table 8.14. It is important to note that, while only the scores of the first 10 systems are depicted in Tables 8.13 and 8.14, the scores of *all* 96 systems are employed for the purpose of conducting the paired t-test.

Table 8.13: The average of the AERs over the 9 iterations of cross-validation of the first 10 systems within the context of the FGE and the BME when employing test samples from the Bosphorus database.

System Design	AER: FGE	AER: BME
2CH NO DO AER_{min}	15.10522	31.61656
2CH NO ST AER_{min}	15.25078	23.53389
2CH CL BD AER_{min}	15.57256	21.72856
2CH NO DO FRR_{zero}	15.93744	50.0
2CH CL BD EER	16.01689	21.18833
2CH NO ST EER	16.082	23.20222
2CH CL BN EER	16.15033	21.06856
2CH CL BN AER_{min}	17.84422	21.33478
2CH FB BD FRR_{zero}	18.12511	50.0
2CH CL DO FRR_{zero}	18.14644	50.0

Table 8.14: The average of the AERs over the 9 iterations of cross-validation of the first 10 systems within the context of the FGE and the BME when employing test samples from the Wilches database.

System Design	AER: FGE	AER: BME
2CH NO ST EER	4.27078	7.04856
2CH NO DO AER_{min}	4.39978	12.15278

2CH NO ST AER_{min}	4.47922	6.94456
2CH NO DO EER	4.92056	10.83322
2CH FB DO AER_{min}	5.15378	28.99289
2CH FB BN EER	5.27278	9.479
2CH FB BD AER_{min}	5.33233	8.33322
2CH FB BN AER_{min}	5.397	9.51378
2CH FB DO EER	5.43656	26.38878
2CH FB BD EER	5.63989	7.56933

The results of the one-sided paired t-tests between the scores of the proposed systems within the context of the FGE and the BME are presented in Table 8.15.

Table 8.15: The results of the one-sided paired t-tests between the scores of the proposed systems within the context of the FGE and the BME.

Test database	**t-statistic**	**p-value**
Bosphorus	-4.07371	4.79474e-05
Wilches	-5.01578	1.22608e-06

If the null hypothesis may be *rejected* if the p-value is less than 0.05, it is clear from the results depicted in Table 8.15 that the proficiency of the proposed systems is *statistically* better within the context of the FGE than within the context of the BME. This further strengthens the hypothesis that the utilisation of the GenVeins database for the purpose of training and validating the proposed systems on *representative* sets of individuals *significantly* increases the proficiency of the proposed systems, which in turn validates the significant contribution the GenVeins database to the current literature within the context of dorsal hand vein-based authentication using deep learning.

It is important to note that a statistical comparison between the proficiency of the systems proposed in this study and that of SOTA systems proposed in recent literature (see Chapter 2) is *deliberately* omitted. The reason for this constitutes the fact that the experimental protocols proposed in this study is designed in such a way that the proficiency of the proposed systems are gauged within the context of two *actual* real-world scenarios, that is the individual dependent scenario (IDS) and the individual independent scenario (IIS) (see Section 7.1). This is in contrast to the majority of experimental protocols reviewed in Chapter 2, in which case the proposed systems are trained *and* tested on the *same* set of individuals. In other words, the majority of the results reported in recent literature *only* serve as proofs-of-concept (POCs), since it is infeasible to implement a system in a real world scenario that is trained and tested on the *same* set of individuals. There are however studies [22] [16] [23] which employed similar experimental protocols to the IIS (see Section 7.1). The experimental protocols, databases and statistical performance measures are however significantly different when compared to those employed in this study, which renders a comparison between the results reported by Thapar *et al.* [22], Zagoruyko and Komodakis [16] and Zagoruyko and Komodakis [23] and the results reported within the context of the IIS infeasible.

8.8 Concluding remarks

The results obtained by gauging the proficiency of the proposed systems within the context of the IDS and the IIS were presented and discussed in this chapter. The results of the BME and the augmented BME were statistically compared (see Section 8.7.1) in order to determine whether or not data augmentation is beneficial within the context of the BME. The results of the BME and the FGE were also statistically compared (see Section 8.7.2) in order to determine whether or not the utilisation of the GenVeins database for the purpose of training the proposed systems on a sufficient number of different individuals enhances system proficiency during testing. A number of feasible solutions were provided to the deformity problem (see Section 8.1) and the problem of the unavailability of a sufficient number of *different* training individuals, which includes either the solution associated with the IDS, or the solution associated with the SGE. This dissertation is concluded in the next chapter, in which some avenues for future research are also discussed.

Chapter 9

Future work and conclusion

The proficiency of the systems proposed in this study may be further improved in a number of ways. Possible avenues for future work are envisioned in Section 9.1, while a conclusion of the research conducted in this study is provided in Section 9.2.

9.1 Future work

A number of ways in which the proficiency of the systems proposed in this study may be further increased is discussed in this section. These avenues include the development and/or utilisation of (1) alternative preprocessing protocols (see Section 9.1.1), (2) improved neural network-based feature extractors (see Section 9.1.2) and (3) alternative verification protocols (see Section 9.1.3). Possible shortcomings and improvement of the employed experimental and hyperparameter tuning protocols are discussed in Section 9.1.4, while the necessary comparison with prior work is emphasised in Section 9.1.5.

9.1.1 Alternative preprocessing protocols

Feasible alternatives to the preprocessing protocols employed in this study, which may further mitigate the deformity problem discussed in Chapter 8, are presented in this section. Alternative contrast enhancement techniques are discussed in Section 9.1.1.1, while alternative binarisation techniques are discussed in Section 9.1.1.2.

9.1.1.1 Alternative contrast enhancement protocols

The contrast enhancement algorithm employed in this study, namely CLAHE (see Section 3.3.2), is a popular method that is commonly used to enhance the contrast in images based on *local* patches. Certain drawbacks of the CLAHE algorithm, for example the over-amplification of noise and the introduction of so-called "halo" artefacts, may however increase the negative effects of the deformity problem, which warrants an investigation into alternative contrast enhancement algorithms within the context of the systems proposed in this study.

The utilisation of Retinex-based algorithms is a feasible alternative to CLAHE in the sense that these algorithms are specifically suited to enhance details *and* mitigate the effect of illumination variations in an image. These algorithms represent the image in question as a combination of reflectance and illumination components, after which adaptive adjustments are made in order to enhance local details *while* maintaining global contrast. Wang *et al.*

[77] reported that a single-scale retinex is robust to *uneven* illumination and shadows when utilised for the purpose of enhancing the contrast of a grey-scale palm vein image.

Local Laplacian filters constitute another feasible alternative to CLAHE, since these filters are able to enhance local details *while* minimising the generation of unwanted artefacts. This is accomplished by adjusting the Laplacian response at different scales and spatial locations in order to *locally* control contrast enhancement, which may further mitigate the negative effects of the deformity problem.

Diffusion models are also fast gaining popularity within the context of image enhancement. These models enhance contrast in low-light digital images by iteratively updating pixel values based on local image characteristics. Diffusion models are furthermore able to reduce noise in images by "diffusing" information from neighbouring pixels, while enhancing contrast by strengthening the image gradient. Wang *et al.* [78] proposed a diffusion equation which stretches or extends the distribution of luminance data of an image, which is shown to be superior to the Markov random field model *and* a fully convolutional network within the context of contrast enhancement of low-contrast remote sensing images.

9.1.1.2 Alternative binarisation protocols

The novel hand vein segmentation protocol employed in this study (see Section 3.3.3) was originally proposed by Beukes [2], and is specifically well suited for the removal of high contrast regions that do *not* match the morphology of hand veins. Arguably more proficient and robust methods have however been proposed in recent literature which may lead to even more accurate segmentation results when compared to the aforementioned segmentation protocol.

The utilisation of deep learning-based segmentation approaches has recently shown promising results in a number of image segmentation tasks. This approach involves training a neural network on a large database of NIR hand vein images with corresponding ground truth segmentations in order to obtain a deep learning-based segmentation model that is able to accurately segment hand veins in unseen images. A hand vein-based segmentation ensemble that employs a CNN is proposed by Lefkovits *et al.* [79], and is shown to outperform many traditional segmentation methods.

Level set methods are known for their efficacy within the context of segmenting *complex* shapes, and are commonly employed in the field of medicine for the purpose of blood vessel segmentation. Li *et al.* [47] developed a novel level set-based method called Adaptive Prior Shape Level Set Evolution (APSLSE) which is shown to effectively mitigate the problems of initial contour sensitivity and unidirectional variation of area terms. The proficiency of APSLSE is gauged by comparing the segmentation results of a number of blood vessel images to those that were manually segmented by professionals. A FRR of 0.1930% and a FAR of 0.04633% is reported.

9.1.2 Improved neural network-based feature extractors

The CNN-based FEs employed in this study (see Section 6.2) are based on traditional CNNs, which are commonly utilised for the purpose of automatically extracting suitable features from images. These methods have however since been significantly improved and are outperformed by more proficient methods and architectures, which warrants an investigation into the utilisation of these updated methods and architectures as FEs within the context of the systems proposed in this study.

Normalised cross-correlation (NCC) networks are shown by Zagoruyko and Komodakis [23] to significantly outperform traditional CNNs within the context of image matching, specifically when presented with images containing *unseen* objects. The main advantage of NCC networks over traditional CNNs constitutes the fact that these networks are inherently *invariant* to rotation, scale *and* illumination, which renders them a viable alternative to the CNN-based feature extractors proposed in this study.

Capsule networks [80] constitute another neural network architecture which has recently shown promising results within the context of image matching. These networks are superior to traditional CNNs in the sense that they are designed to encode and represent various *parts* of an object, as opposed to traditional CNNs in which the image is represented on a *pixel* level. The parts of the object in question, including their orientation, are represented by groups of neurons called *capsules*, which trains the network to associate different parts/capsules with objects *regardless* of their orientation and even possible occlusion. Capsule network-based feature extractors are therefore a viable alternative to the CNN-based feature extractors proposed in this study.

The utilisation of transformer-based systems within the context of image classification has significantly increased in recent years, mainly due to their ability to model long-range pixel dependencies. Transformers are therefore superior to traditional CNNs in the sense that (1) traditional CNNs are unable to model long-range pixel dependencies due to their *local* receptive fields and (2) positional information is explicitly encoded into transformers. These advantages allow transformers to model the *relative* position of pixels in an image. The utilisation of transformer-based feature extractors should therefore be investigated within the context of the systems proposed in this study.

9.1.3 Alternative verification protocols

The softmax verifier employed in this study (see Section 6.3.1) constitutes the benchmark verifier within the context of CNNs. Several alternative classification/verification techniques have however been recently proposed within the context of CNNs that demonstrate increased proficiency when compared to the softmax verifier.

Several recently published papers have demonstrated that system proficiency may be enhanced by replacing the softmax classifier with a more advanced machine learning architecture such as Support Vector Machines (SVMs). Niu and Suen [81] concluded that a system which comprises of a CNN-based feature extractor and a SVM classifier significantly outperforms several other systems that were evaluated on the same database.

Model ensembles are also a popular practice within the context of machine learning, and involve obtaining *multiple* predictions from a number of models, after which these predictions are collectively considered in order to obtain the final prediction. Model ensembles generally outperform each of the individual models, which warrants further investigation into this practice within the context of the systems proposed in this study.

A conclusion of the research conducted in this study is presented in the following section.

9.1.4 Improved experimental and hyperparameter tuning protocols

The experimental protocols developed in this study may be improved in a number of ways. First, the utilisation of more of the available training, validation and test samples could provide valuable insight into the generalisation potential and the proficiency of the underlying systems within the context of both the IDS and IIS. Furthermore, insight may also be gained by training, validating and testing the systems with a ratio of positive and negative samples that is expected in a typical real world scenario involving hand vein-based authentication. The majority of FARs reported in this study may therefore be larger than expected in a typical real world scenario, in the sense that the proposed systems were tested with an unusually large number of imposter samples.

The manual hyperparameter tuning protocol employed in this study (see Section 7.6.1) has many drawbacks such as the risk of manually defining a suboptimal search space and the fact that it is much more time consuming than a properly implemented automatic hyperparameter tuning protocol. Some of the aspects of the manual hyperparameter tuning protocol employed in this study that may be improved by a suitable automatic hyperparameter tuning protocol is briefly discussed below.

1. A design choice was made in this study to consider a single CNN-based feature extractor with the inclusion or omission of batch normalisation and/or dropout layers as different models referred to as the four "variations" (see Section 6.2). This leads to an unnecessarily large number of highly similar experiments with potentially limited insight into the proficiency of the different systems. It is arguably a much better approach to include the decision of whether or not to add and/or remove batch normalisation and dropout layers into an automatic hyperparameter tuning protocol, which would produce a properly tuned system and significantly reduce (1) the number of results to analyse and (2) the execution time.

2. The dropout rate has been fixed at 0.2 for all the systems proposed in this study in which dropout layers are included (see Section 6.2.3). It is arguably much better to instead consider the dropout rate as a hyperparameter, which could then be tuned by employing a suitable automatic hyperparameter tuning protocol. The main advantage of doing so constitutes a possible increase in the proficiency of the systems in which dropout layers are included.

3. The learning rate has been fixed at $1e^{-4}$ for all the systems proposed in this study (see Section 7.6.1) due to the fact that the vast majority of systems achieve near-zero loss at the end of training. A feasible avenue for future work however is to consider the learning rate as another hyperparameter which could be optimised by employing a suitable automatic hyperparameter tuning protocol. Possible advantages of doing so include a significant decrease in training time due to faster convergence, whereby training may be terminated early by employing a suitable early stopping criterion. It may also be useful to employ the so-called ADAM optimiser [82] as opposed to ASGD which is employed in this study (see Section 7.6.1), which automatically adjusts the learning rate after each epoch in a suitable fashion.

4. The hyperparameters associated with the CNN-based feature extractors such as the shape of the kernels and the depth of the CNNs were *manually* optimised within the context of this study (see Section 7.6.1). A feasible avenue for future work therefore constitutes the utilisation of a suitable automatic hyperparameter tuning protocol for the purpose of determining the optimal number of convolutional and pooling layers, as well as the optimal configuration of the sizes and shapes of the kernels in question. One of the main advantages of doing so includes the ability to consider a much larger search space of different hyperparameter configurations which would invariably lead to at least a slight increase in system proficiency. Such an automatic hyperparameter

tuning protocol will additionally yield much more trustworthy results when compared to a manually defined hyperparameter configuration of the networks.

The proposed manual hyperparameter tuning protocol is also arguably insufficient for the purpose of comparing the proficiency of the systems proposed in this study within the context of a *single* experimental scenario, which also motivates the development or utilisation of an improved hyperparameter tuning protocol.

9.1.5 Comparison with prior work

It is important to note that the systems proposed in this study is not compared to *any* state-of-the-art (SOTA) systems as explained in Section 1.4.2.2. Such a comparison is highly warranted in order to gauge the potential gain in system proficiency by employing more advanced deep learning approaches within the context of the experimental scenarios proposed in this study.

The reason for not employing nor designing SOTA machine learning systems within the context of this study constitutes the fact that one of the main objectives of this study is to obtain benchmark results of various basic CNN architectures within the context of the novel experimental protocols developed during the course of this study. These results also provide valuable insight into the key areas of concern which may subsequently guide future researchers to improved system designs that specifically target said areas of concern.

A system in which improved CNN architectures (see Section 9.1.2) and improved verifiers (see Section 9.1.3) are employed may therefore yield significantly better results than the proposed systems within the context of the novel experimental protocols developed in this study, and is therefore considered by the author to be a feasible avenue for future research.

9.2 Conclusion

A number of hand vein-based biometric authentication systems were proposed in this dissertation. These systems comprise of (1) a preprocessing protocol (see Chapter 3), (2) a neural network architecture (see Section 5.4), (3) a convolutional neural network-based feature extractor (CNN-based FE) (see Section 6.2) and (4) a probabilistic threshold selection criterion (see Section 7.6.2). The proficiency of the proposed systems were gauged within the context of a total of *four* unique experiments, of which the individual *dependent* experiment (IDE) is associated with the individual *dependent* scenario (IDS), while the other *three*, namely the benchmark experiment (BME), the first GenVeins experiment (FGE) and the second GenVeins experiment (SGE), are associated with the individual *independent* scenario (IIS). These experiments are specifically designed for the purpose of simulating the proficiency of the proposed systems within the context of *actual* real world scenarios, as opposed to *only* serving as so-called proofs-of-concept (POCs), as is the case with the majority of systems proposed in recent literature.

The IIS constitutes the ideal scenario within the context of hand vein-based authentication and comprises of a system that is *pre-trained* on a *sufficient* number of *different* individuals, which allows the system to generalise sufficiently well when presented with images from *unseen* individuals during implementation. The challenge associated with this scenario however constitutes the fact that it is time-consuming and difficult to acquire hand vein images of a set of individuals that is large enough so as to be *representative* of the population. The majority of publicly available databases do therefore *not* contain a representative set of individuals. The four experiments conducted in this study were specifically designed in order to address this issue.

The IDE involves training a *tailor-made* network for *each* individual enrolled into the system. This solution constitutes a simple alternative to the problem of acquiring hand vein images of a sufficiently large set of *different* individuals, since a model for the individual in question may be trained *during* enrolment in mere minutes. Experimental results clearly show that this solution is extremely proficient and a viable solution to the aforementioned problem which requires *no alteration* before being implemented within the context of an *actual* real world scenario.

The BME was conducted in order to gauge the negative impact of training a *single* network on an *insufficient* number of different individuals that is unable to generalise sufficiently well when presented with images from *unseen* individuals during implementation. Experimental results within the context of the BME (see Section 8.6.1) clearly show that the proficiency of the systems proposed in this study during testing is severely impaired by the unavailability of a sufficiently large number of *different* training individuals.

The GenVeins database is proposed in this study as *another* viable solution to the problem of acquiring hand vein samples of a sufficiently large number of different individuals for the purpose of training (see Chapter 4). The GenVeins database is an artificially generated hand vein database which contains hand vein images of a total of 20 000 *fictitious* individuals. The results of the FGE (see Section 8.6.2) clearly indicate the significant increase in proficiency of the proposed systems when trained on a *sufficiently* large number of *different* individuals, as opposed to being trained on an *insufficient* number of different individuals.

Another significant problem within the context of hand vein-based biometric authentication constitutes so-called “vein deformity”, which is a result of excessive variation in the orientation of an individual’s hand during the acquisition of multiple NIR images. This issue was sufficiently mitigated by the SGE in which an individual is simply granted a total of three chances for authentication. The proficiency of the proposed systems within the context of the SGE (see Section 8.6.3) is outstanding, especially considering the fact that these results are a *direct* indication of the proficiency of the proposed systems within the context of an *actual* real world scenario.

The systems and experimental scenarios proposed in this study open up a vast network of avenues for future research, especially considering the major contribution of the GenVeins database. The author is therefore of the opinion that this study should be expanded and integrated into the field of hand vein-based biometric authentication. The possibilities are virtually endless.

Appendix A

Full experimental results

A.1 Introduction

The results for the four experiments conducted in this study, that is (1) the individual dependent experiment (IDE), (2) the benchmark experiment (BME), (3) the first GenVeins experiment (FGE) and (4) the second GenVeins experiment (SGE), for the purpose of gauging the proficiency of *all* 96 systems proposed in this study are presented in this appendix. The discussions of the aforementioned results are provided in Chapter 8. The results of the so-called "augmented" BME are also presented and discussed in this chapter (see Section A.4).

Table 8.1, which contains abbreviations and descriptions of the different system components, is reproduced in Table A.1 for convenience.

Table A.1: Abbreviations of different system components and their descriptions.

Preprocessing protocols
NO: No preprocessing (see Section 3.3.1)
CL: CLAHE-based contrast enhancement (see Section 3.3.2)
FB: Full binarisation (see Section 3.3.3)
Neural network architectures
2CH: Two-channel networks (see Section 5.5.1)
SM: Siamese networks (see Section 5.5.2)
CNN-based feature extraction
ST: Standard (see Section 6.2.1)
BN: Standard with added batch normalisation layers (see Section 6.2.2)
DO: Standard with added dropout layers (see Section 6.2.3)
BD: Standard with added batch normalisation and dropout layers (see Section 6.2.4)
Probabilistic threshold criteria
EER: Equal error rate (see Section 7.6.2)
AER_{min}: Minimum average error rate (see Section 7.6.2)
FAR_{zero}: Zero false acceptance rate (see Section 7.6.2)
FRR_{zero}: Zero false rejection rate (see Section 7.6.2)

A.2 The individual dependent experiment

The results for all 96 systems within the context of the Bosphorus and Wilches databases and the IDE are depicted in Tables A.2 and A.3 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Table A.2: The results of all 96 systems ranked according to the AER within the context of the IDE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	SM BD CL AER_{min}	1.01	2.405	1.708	98.99	97.595	1.351
2	SM ST FB AER_{min}	1.035	2.504	1.769	98.965	97.496	1.095
3	SM ST FB EER	1.563	2.16	1.862	98.437	97.84	1.151
4	SM BD CL EER	1.523	2.249	1.886	98.477	97.751	1.552
5	SM BN FB AER_{min}	0.462	3.455	1.958	99.538	96.545	0.689
6	SM DO FB AER_{min}	0.654	3.516	2.085	99.346	96.484	1.427
7	SM BD FB AER_{min}	0.338	3.84	2.089	99.662	96.16	0.701
8	SM DO FB EER	1.148	3.139	2.144	98.852	96.861	1.436
9	SM BD NO AER_{min}	1.815	2.581	2.198	98.185	97.419	2.438
10	2CH ST FB AER_{min}	0.961	3.45	2.206	99.039	96.55	1.299
11	SM BN CL AER_{min}	1.672	2.77	2.221	98.328	97.23	1.86
12	SM BN FB EER	2.026	2.591	2.309	97.974	97.409	0.855
13	2CH ST FB EER	2.054	2.69	2.372	97.946	97.31	1.321
14	2CH DO FB AER_{min}	1.142	3.616	2.379	98.858	96.384	1.612
15	2CH BN FB AER_{min}	0.798	3.971	2.385	99.202	96.029	1.155
16	SM BD FB FAR_{zero}	0.029	4.777	2.403	99.971	95.223	1.396
17	SM BN CL EER	1.964	2.907	2.435	98.036	97.093	2.054
18	SM BD FB EER	1.82	3.057	2.438	98.18	96.943	0.908
19	SM BD NO EER	1.951	3.093	2.522	98.049	96.907	3.044
20	2CH BD FB AER_{min}	0.91	4.142	2.526	99.09	95.858	1.117
21	2CH DO FB EER	2.263	2.835	2.549	97.737	97.165	1.699
22	2CH BN FB EER	2.3	2.996	2.648	97.7	97.004	1.232
23	2CH BD FB EER	2.388	3.163	2.775	97.612	96.837	1.16
24	SM BN NO AER_{min}	2.918	3.057	2.987	97.082	96.943	3.77
25	SM BN FB FAR_{zero}	0.039	6.058	3.049	99.961	93.942	2.924
26	SM ST CL AER_{min}	2.247	4.227	3.237	97.753	95.773	3.625
27	SM BN NO EER	2.938	3.871	3.405	97.062	96.129	4.115

28	SM ST CL EER	2.634	4.558	3.596	97.366	95.442	3.702
29	2CH BN FB FAR_{zero}	0.044	7.339	3.691	99.956	92.661	2.522
30	2CH BD CL AER_{min}	2.324	5.528	3.926	97.676	94.472	4.565
31	2CH BD FB FAR_{zero}	0.041	7.833	3.937	99.959	92.167	2.642
32	2CH BN CL AER_{min}	2.39	5.543	3.966	97.61	94.457	4.889
33	2CH DO FB FAR_{zero}	0.055	8.152	4.104	99.945	91.848	3.901
34	2CH ST FB FAR_{zero}	0.039	8.218	4.128	99.961	91.782	4.141
35	2CH BN CL EER	3.13	5.221	4.176	96.87	94.779	4.891
36	2CH BD CL EER	3.034	5.329	4.181	96.966	94.671	4.673
37	SM DO CL AER_{min}	1.199	7.21	4.204	98.801	92.79	5.781
38	SM DO CL EER	1.28	7.373	4.327	98.72	92.627	5.682
39	SM ST NO AER_{min}	3.555	5.214	4.385	96.445	94.786	4.473
40	2CH BD NO AER_{min}	2.633	6.439	4.536	97.367	93.561	5.497
41	2CH BN NO AER_{min}	2.497	6.612	4.555	97.503	93.388	5.695
42	2CH BD NO EER	2.836	6.452	4.644	97.164	93.548	5.475
43	2CH BN NO EER	2.748	6.751	4.749	97.252	93.249	5.727
44	SM DO NO AER_{min}	2.046	7.479	4.762	97.954	92.521	5.943
45	2CH DO CL AER_{min}	3.315	6.711	5.013	96.685	93.289	6.06
46	SM ST NO EER	3.18	6.892	5.036	96.82	93.108	4.665
47	SM DO NO EER	2.005	8.074	5.04	97.995	91.926	6.101
48	2CH DO CL EER	4.044	6.682	5.363	95.956	93.318	5.999
49	2CH ST CL AER_{min}	2.994	7.932	5.463	97.006	92.068	6.738
50	2CH ST CL EER	3.834	7.659	5.747	96.166	92.341	6.815
51	SM DO FB FAR_{zero}	0.044	11.453	5.748	99.956	88.547	8.277
52	2CH ST NO AER_{min}	3.125	8.523	5.824	96.875	91.477	7.201
53	2CH DO NO AER_{min}	3.45	8.365	5.907	96.55	91.635	6.928
54	2CH ST NO EER	3.343	8.895	6.119	96.657	91.105	7.156
55	2CH DO NO EER	3.589	9.133	6.361	96.411	90.867	7.023
56	2CH BD NO FRR_{zero}	15.777	3.644	9.711	84.223	96.356	11.864
57	2CH ST NO FRR_{zero}	13.563	5.892	9.727	86.437	94.108	12.882
58	2CH BD CL FAR_{zero}	0.101	21.051	10.576	99.899	78.949	12.354
59	2CH DO NO FRR_{zero}	16.514	5.236	10.875	83.486	94.764	14.454
60	2CH BN NO FRR_{zero}	18.414	3.696	11.055	81.586	96.304	13.552
61	2CH DO CL FAR_{zero}	0.135	22.294	11.214	99.865	77.706	11.463
62	2CH BN CL FAR_{zero}	0.098	22.67	11.384	99.902	77.33	13.152
63	2CH ST CL FAR_{zero}	0.142	22.696	11.419	99.858	77.304	11.638
64	SM ST FB FAR_{zero}	0.034	22.946	11.49	99.966	77.054	10.972
65	SM DO CL FAR_{zero}	0.128	24.698	12.413	99.872	75.302	12.493

66	SM BD NO FRR$_{zero}$	24.803	0.848	12.826	75.197	99.152	16.185
67	2CH BD NO FAR$_{zero}$	0.112	25.548	12.83	99.888	74.452	13.52
68	2CH BN NO FAR$_{zero}$	0.102	25.936	13.019	99.898	74.064	13.589
69	SM ST NO FRR$_{zero}$	23.957	2.806	13.381	76.043	97.194	18.647
70	2CH BD CL FRR$_{zero}$	25.801	2.095	13.948	74.199	97.905	15.374
71	2CH ST CL FRR$_{zero}$	23.692	4.386	14.039	76.308	95.614	16.491
72	2CH BN CL FRR$_{zero}$	28.426	1.721	15.074	71.574	98.279	15.648
73	2CH DO CL FRR$_{zero}$	27.479	3.187	15.333	72.521	96.813	18.088
74	2CH ST NO FAR$_{zero}$	0.134	32.016	16.075	99.866	67.984	14.706
75	SM BN NO FRR$_{zero}$	33.692	0.771	17.232	66.308	99.229	17.5
76	SM ST CL FRR$_{zero}$	34.763	1.669	18.216	65.237	98.331	21.938
77	SM ST CL FAR$_{zero}$	0.118	38.397	19.258	99.882	61.603	15.724
78	2CH DO NO FAR$_{zero}$	0.142	38.522	19.332	99.858	61.478	16.781
79	SM BD CL FRR$_{zero}$	42.287	0.531	21.409	57.713	99.469	18.842
80	SM BN CL FRR$_{zero}$	43.76	0.434	22.097	56.24	99.566	18.555
81	SM DO NO FAR$_{zero}$	0.14	46.361	23.25	99.86	53.639	16.997
82	SM DO NO FRR$_{zero}$	45.369	3.173	24.271	54.631	96.827	23.036
83	2CH ST FB FRR$_{zero}$	52.102	0.156	26.129	47.898	99.844	15.07
84	SM BD CL FAR$_{zero}$	0.05	56.077	28.063	99.95	43.923	22.429
85	2CH BN FB FRR$_{zero}$	56.47	0.147	28.309	43.53	99.853	14.119
86	2CH DO FB FRR$_{zero}$	58.913	0.181	29.547	41.087	99.819	16.049
87	2CH BD FB FRR$_{zero}$	59.709	0.148	29.929	40.291	99.852	13.725
88	SM ST FB FRR$_{zero}$	61.189	0.184	30.687	38.811	99.816	20.881
89	SM DO CL FRR$_{zero}$	62.772	1.831	32.302	37.228	98.169	22.862
90	SM BN CL FAR$_{zero}$	0.045	65.574	32.809	99.955	34.426	21.163
91	SM ST NO FAR$_{zero}$	0.129	67.241	33.685	99.871	32.759	16.337
92	SM BD NO FAR$_{zero}$	0.057	70.024	35.041	99.943	29.976	19.69
93	SM BN FB FRR$_{zero}$	74.206	0.056	37.131	25.794	99.944	13.638
94	SM BN NO FAR$_{zero}$	0.057	77.473	38.765	99.943	22.527	17.702
95	SM BD FB FRR$_{zero}$	88.374	0.028	44.201	11.626	99.972	11.921
96	SM DO FB FRR$_{zero}$	92.386	0.053	46.219	7.614	99.947	12.651

Table A.3: The results of all 96 systems ranked according to the AER within the context of the IDE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**

1	SM BD FB AER_{min}	0.507	1.198	0.853	99.493	98.802	0.928
2	SM BD FB EER	0.763	1.113	0.938	99.237	98.887	1.017
3	SM BN FB AER_{min}	0.533	1.435	0.984	99.467	98.565	1.286
4	SM BN FB EER	0.767	1.326	1.047	99.233	98.674	1.21
5	SM BD CL AER_{min}	0.943	1.542	1.242	99.057	98.458	1.704
6	SM BD NO AER_{min}	1.396	1.355	1.375	98.604	98.645	1.96
7	SM BD CL EER	1.227	1.845	1.536	98.773	98.155	2.603
8	SM BN CL AER_{min}	1.446	1.727	1.587	98.554	98.273	2.066
9	SM ST CL AER_{min}	1.802	1.58	1.691	98.198	98.42	2.134
10	SM ST FB AER_{min}	1.462	1.959	1.71	98.538	98.041	2.391
11	2CH BN NO AER_{min}	0.972	2.583	1.778	99.028	97.417	3.88
12	SM DO CL AER_{min}	0.954	2.702	1.828	99.046	97.298	3.379
13	SM BD NO EER	1.223	2.505	1.864	98.777	97.495	2.772
14	2CH BN NO EER	0.904	2.848	1.876	99.096	97.152	4.067
15	2CH BN CL AER_{min}	1.144	2.619	1.882	98.856	97.381	3.3
16	2CH BN FB AER_{min}	1.114	2.688	1.901	98.886	97.312	2.79
17	SM BN CL EER	1.427	2.376	1.901	98.573	97.624	2.398
18	2CH BN CL EER	1.237	2.669	1.953	98.763	97.331	3.395
19	2CH BD CL AER_{min}	1.048	2.864	1.956	98.952	97.136	3.282
20	2CH BD NO AER_{min}	1.132	2.795	1.963	98.868	97.205	3.632
21	2CH BN FB EER	1.332	2.624	1.978	98.668	97.376	2.901
22	SM DO CL EER	0.904	3.1	2.002	99.096	96.9	3.601
23	2CH BD FB AER_{min}	1.212	2.868	2.04	98.788	97.132	3.247
24	2CH BD CL EER	1.214	2.887	2.05	98.786	97.113	3.433
25	2CH BD NO EER	1.068	3.058	2.063	98.932	96.942	3.808
26	2CH BD FB EER	1.45	2.753	2.102	98.55	97.247	3.222
27	SM ST FB EER	1.304	2.924	2.114	98.696	97.076	2.668
28	SM BN NO AER_{min}	2.599	1.764	2.181	97.401	98.236	3.195
29	SM ST CL EER	1.501	2.979	2.24	98.499	97.021	2.459
30	2CH DO CL AER_{min}	1.669	3.065	2.367	98.331	96.935	3.58
31	SM DO NO AER_{min}	1.376	3.465	2.421	98.624	96.535	4.835
32	SM ST NO AER_{min}	3.351	1.549	2.45	96.649	98.451	2.638
33	2CH DO CL EER	1.891	3.174	2.532	98.109	96.826	3.812
34	2CH ST CL AER_{min}	1.385	3.701	2.543	98.615	96.299	4.332
35	2CH ST FB AER_{min}	1.53	3.579	2.554	98.47	96.421	4.599
36	2CH DO NO AER_{min}	1.627	3.551	2.589	98.373	96.449	4.842
37	2CH ST CL EER	1.468	3.845	2.657	98.532	96.155	4.455
38	2CH ST FB EER	1.633	3.773	2.703	98.367	96.227	4.765

39	2CH DO FB AER_{min}	1.327	4.181	2.754	98.673	95.819	4.527
40	SM DO FB AER_{min}	1.621	3.925	2.773	98.379	96.075	6.288
41	2CH ST NO AER_{min}	1.602	3.966	2.784	98.398	96.034	5.496
42	SM DO NO EER	1.184	4.411	2.797	98.816	95.589	5.126
43	2CH DO FB EER	1.502	4.253	2.878	98.498	95.747	4.727
44	2CH DO NO EER	1.399	4.496	2.948	98.601	95.504	5.385
45	SM BN NO EER	2.31	3.663	2.987	97.69	96.337	4.453
46	2CH ST NO EER	1.343	4.694	3.018	98.657	95.306	5.919
47	SM DO FB EER	1.403	4.638	3.021	98.597	95.362	6.541
48	SM BD NO FRR_{zero}	5.81	0.873	3.342	94.19	99.127	7.311
49	2CH BD NO FRR_{zero}	5.7	1.84	3.77	94.3	98.16	7.586
50	2CH BN NO FRR_{zero}	5.524	2.057	3.79	94.476	97.943	7.524
51	SM ST NO EER	2.72	4.945	3.832	97.28	95.055	3.45
52	2CH ST NO FRR_{zero}	5.362	3.164	4.263	94.638	96.836	8.165
53	2CH DO NO FRR_{zero}	6.586	2.845	4.715	93.414	97.155	9.511
54	SM BN NO FRR_{zero}	8.478	1.019	4.749	91.522	98.981	8.903
55	2CH BN FB FAR_{zero}	0.104	9.436	4.77	99.896	90.564	7.254
56	SM BD FB FAR_{zero}	0.088	9.677	4.882	99.912	90.323	9.965
57	2CH BD FB FAR_{zero}	0.124	10.296	5.21	99.876	89.704	8.625
58	SM ST NO FRR_{zero}	9.914	0.989	5.451	90.086	99.011	11.628
59	SM ST CL FRR_{zero}	10.828	1.026	5.927	89.172	98.974	13.427
60	2CH ST CL FAR_{zero}	0.15	11.914	6.032	99.85	88.086	8.758
61	2CH BD CL FRR_{zero}	10.789	1.36	6.074	89.211	98.64	9.959
62	2CH BN NO FAR_{zero}	0.145	12.066	6.105	99.855	87.934	11.26
63	2CH DO CL FAR_{zero}	0.165	12.166	6.166	99.835	87.834	9.024
64	2CH ST FB FAR_{zero}	0.113	12.394	6.253	99.887	87.606	10.047
65	2CH BD CL FAR_{zero}	0.116	12.946	6.531	99.884	87.054	11.104
66	2CH ST CL FRR_{zero}	11.222	2.015	6.619	88.778	97.985	10.602
67	SM BN FB FAR_{zero}	0.073	13.31	6.691	99.927	86.69	11.787
68	2CH BN CL FRR_{zero}	12.473	1.315	6.894	87.527	98.685	11.336
69	2CH BN CL FAR_{zero}	0.102	13.722	6.912	99.898	86.278	11.942
70	2CH BD NO FAR_{zero}	0.192	13.769	6.981	99.808	86.231	12.013
71	2CH DO FB FAR_{zero}	0.15	13.831	6.991	99.85	86.169	11.083
72	2CH BN FB FRR_{zero}	13.111	1.107	7.109	86.889	98.893	10.734
73	2CH BD FB FRR_{zero}	14.055	1.126	7.59	85.945	98.874	10.948
74	2CH ST FB FRR_{zero}	13.37	1.908	7.639	86.63	98.092	13.014
75	SM BD CL FRR_{zero}	14.487	0.821	7.654	85.513	99.179	12.837
76	2CH DO CL FRR_{zero}	13.778	1.864	7.821	86.222	98.136	13.179

77	SM BN FB FRR_{zero}	15.529	0.427	7.978	84.471	99.573	12.868
78	SM BD FB FRR_{zero}	15.979	0.451	8.215	84.021	99.549	13.681
79	2CH DO FB FRR_{zero}	16.098	2.328	9.213	83.902	97.672	14.97
80	SM BN CL FRR_{zero}	17.889	0.609	9.249	82.111	99.391	13.704
81	SM ST FB FRR_{zero}	18.707	1.012	9.859	81.293	98.988	17.987
82	2CH ST NO FAR_{zero}	0.201	20.725	10.463	99.799	79.275	14.006
83	SM DO NO FRR_{zero}	19.106	2.238	10.672	80.894	97.762	18.648
84	SM DO CL FAR_{zero}	0.131	22.439	11.285	99.869	77.561	14.262
85	2CH DO NO FAR_{zero}	0.205	23.441	11.823	99.795	76.559	16.028
86	SM DO FB FAR_{zero}	0.094	29.529	14.811	99.906	70.471	16.209
87	SM DO CL FRR_{zero}	33.785	1.262	17.524	66.215	98.738	22.83
88	SM DO NO FAR_{zero}	0.181	37.043	18.612	99.819	62.957	18.151
89	SM DO FB FRR_{zero}	40.824	1.873	21.348	59.176	98.127	23.508
90	SM ST FB FAR_{zero}	0.08	45.272	22.676	99.92	54.728	19.841
91	SM BD CL FAR_{zero}	0.066	51.016	25.541	99.934	48.984	22.323
92	SM ST CL FAR_{zero}	0.094	53.352	26.723	99.906	46.648	19.082
93	SM BD NO FAR_{zero}	0.115	54.552	27.333	99.885	45.448	21.092
94	SM BN CL FAR_{zero}	0.052	62.865	31.459	99.948	37.135	21.304
95	SM BN NO FAR_{zero}	0.094	67.708	33.901	99.906	32.292	19.93
96	SM ST NO FAR_{zero}	0.058	83.649	41.854	99.942	16.351	15.027

A.3 The benchmark experiment

The results for all 96 systems within the context of the Bosphorus and Wilches databases and the BME are depicted in Tables A.4 and A.5 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Table A.4: The results of all 96 systems ranked according to the AER within the context of the BME and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH BN NO EER	20.471	18.086	19.278	79.529	81.914	5.221
2	2CH BD NO EER	18.912	20.139	19.525	81.088	79.861	3.48
3	2CH BN NO AER_{min}	18.017	21.281	19.649	81.983	78.719	5.068
4	2CH BD NO AER_{min}	16.775	22.562	19.668	83.225	77.438	3.952
5	2CH BN CL EER	20.424	21.713	21.069	79.576	78.287	4.863

6	2CH BD CL EER	23.434	18.943	21.188	76.566	81.057	3.288
7	2CH BN CL AER_{min}	16.89	25.779	21.335	83.11	74.221	4.678
8	2CH BD CL AER_{min}	22.978	20.478	21.729	77.022	79.522	3.468
9	2CH ST NO EER	26.906	19.498	23.202	73.094	80.502	3.434
10	2CH ST NO AER_{min}	24.228	22.84	23.534	75.772	77.16	3.696
11	2CH ST CL EER	19.406	28.719	24.063	80.594	71.281	3.678
12	2CH ST CL AER_{min}	19.221	28.974	24.097	80.779	71.026	3.668
13	SM BD NO AER_{min}	25.131	24.645	24.888	74.869	75.355	3.208
14	2CH BD FB EER	26.844	24.56	25.702	73.156	75.44	4.682
15	SM BD NO EER	22.315	29.275	25.795	77.685	70.725	3.652
16	2CH BD FB AER_{min}	24.066	28.04	26.053	75.934	71.96	5.23
17	SM DO NO EER	10.069	44.151	27.11	89.931	55.849	2.25
18	SM DO NO AER_{min}	10.069	44.151	27.11	89.931	55.849	2.25
19	2CH BN FB EER	26.582	29.051	27.816	73.418	70.949	5.095
20	2CH BN FB AER_{min}	25.324	30.872	28.098	74.676	69.128	4.717
21	SM ST CL AER_{min}	32.994	24.259	28.626	67.006	75.741	3.817
22	2CH ST FB EER	29.437	27.847	28.642	70.563	72.153	4.911
23	2CH ST FB AER_{min}	24.244	33.295	28.769	75.756	66.705	4.345
24	SM DO CL EER	12.307	46.312	29.31	87.693	53.688	1.977
25	SM DO CL AER_{min}	12.307	46.312	29.31	87.693	53.688	1.977
26	SM ST NO AER_{min}	37.708	22.5	30.104	62.292	77.5	4.079
27	SM BD CL AER_{min}	37.971	23.457	30.714	62.029	76.543	4.706
28	SM BD FB AER_{min}	34.066	27.778	30.922	65.934	72.222	3.504
29	SM ST CL EER	28.171	33.796	30.984	71.829	66.204	4.019
30	SM ST FB AER_{min}	20.741	41.512	31.126	79.259	58.488	3.463
31	SM BD CL EER	33.457	29.074	31.265	66.543	70.926	4.832
32	SM BD FB EER	31.752	30.802	31.277	68.248	69.198	3.662
33	SM ST FB EER	20.309	42.315	31.312	79.691	57.685	3.668
34	2CH DO NO AER_{min}	32.816	30.417	31.617	67.184	69.583	5.11
35	2CH DO NO EER	35.27	28.634	31.952	64.73	71.366	4.942
36	2CH DO CL EER	30.895	35.965	33.43	69.105	64.035	7.766
37	SM BN NO AER_{min}	55.671	12.176	33.924	44.329	87.824	5.103
38	SM BN FB AER_{min}	49.576	20.525	35.05	50.424	79.475	5.492
39	2CH DO CL AER_{min}	44.275	26.728	35.501	55.725	73.272	9.148
40	SM ST NO EER	34.591	36.42	35.505	65.409	63.58	5.633
41	SM BN FB EER	38.225	36.127	37.176	61.775	63.873	5.181
42	SM BN CL AER_{min}	56.736	19.599	38.168	43.264	80.401	4.588
43	2CH DO FB EER	35.949	45.787	40.868	64.051	54.213	4.854

44	2CH DO FB AER_{min}	41.065	41.944	41.505	58.935	58.056	4.549
45	SM BN NO EER	45.016	42.978	43.997	54.984	57.022	6.479
46	SM BN CL EER	45.748	46.62	46.184	54.252	53.38	6.331
47	SM ST NO FRR_{zero}	40.17	55.571	47.87	59.83	44.429	2.392
48	SM DO FB AER_{min}	45.332	52.052	48.692	54.668	47.948	1.919
49	SM DO FB FAR_{zero}	0.0	97.978	48.989	100.0	2.022	1.892
50	SM ST CL FRR_{zero}	99.105	0.0	49.552	0.895	100.0	0.39
51	SM DO NO FRR_{zero}	99.452	0.0	49.726	0.548	100.0	0.394
52	SM ST FB FRR_{zero}	55.077	44.444	49.761	44.923	55.556	0.357
53	SM DO CL FRR_{zero}	88.673	11.127	49.9	11.327	88.873	0.153
54	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
55	2CH ST NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
56	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
57	2CH BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
58	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
59	2CH DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
60	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
61	2CH BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
62	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
63	2CH ST CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
64	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
65	2CH BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
66	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
67	2CH DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
68	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
70	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH ST FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
72	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
74	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH DO FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
76	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
77	2CH BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
78	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
81	SM DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

82	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
83	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
84	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
85	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
87	SM DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
88	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
90	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
93	SM DO FB FRR_{zero}	44.444	55.556	50.0	55.556	44.444	0.0
94	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
95	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
96	SM DO FB EER	67.045	35.957	51.501	32.955	64.043	3.871

Table A.5: The results of all 96 systems ranked according to the AER within the context of the BME and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH BD CL EER	6.181	4.792	5.486	93.819	95.208	2.036
2	2CH BD CL AER_{min}	5.556	5.903	5.729	94.444	94.097	1.718
3	2CH BD NO AER_{min}	5.764	5.903	5.833	94.236	94.097	1.593
4	2CH BD NO EER	8.472	5.208	6.84	91.528	94.792	3.345
5	2CH BN CL AER_{min}	4.236	9.583	6.91	95.764	90.417	2.236
6	2CH ST NO AER_{min}	6.875	7.014	6.945	93.125	92.986	3.313
7	2CH BN CL EER	6.319	7.639	6.979	93.681	92.361	2.165
8	2CH ST NO EER	8.056	6.042	7.049	91.944	93.958	3.405
9	2CH BN NO EER	5.556	9.444	7.5	94.444	90.556	3.115
10	2CH BD FB EER	11.181	3.958	7.569	88.819	96.042	3.556
11	2CH BD FB AER_{min}	11.042	5.625	8.333	88.958	94.375	4.538
12	2CH BN NO AER_{min}	4.861	12.222	8.542	95.139	87.778	3.697
13	2CH BN FB EER	9.167	9.792	9.479	90.833	90.208	4.893
14	2CH BN FB AER_{min}	7.847	11.181	9.514	92.153	88.819	4.375
15	2CH DO NO EER	12.292	9.375	10.833	87.708	90.625	4.658
16	2CH ST CL EER	10.139	12.014	11.076	89.861	87.986	1.539

17	SM DO CL EER	5.694	16.944	11.319	94.306	83.056	4.218
18	SM DO CL AER_{min}	5.694	16.944	11.319	94.306	83.056	4.218
19	2CH DO NO AER_{min}	11.319	12.986	12.153	88.681	87.014	4.738
20	2CH ST CL AER_{min}	11.181	13.264	12.222	88.819	86.736	2.331
21	2CH ST FB EER	11.875	14.583	13.229	88.125	85.417	7.12
22	SM DO NO EER	8.194	18.75	13.472	91.806	81.25	4.301
23	SM DO NO AER_{min}	8.194	18.75	13.472	91.806	81.25	4.301
24	2CH ST FB AER_{min}	10.972	21.597	16.285	89.028	78.403	8.437
25	2CH DO CL EER	19.444	13.264	16.354	80.556	86.736	5.896
26	SM ST FB AER_{min}	24.722	15.278	20.0	75.278	84.722	3.301
27	SM ST FB EER	24.653	16.528	20.59	75.347	83.472	3.893
28	2CH DO CL AER_{min}	36.458	7.153	21.806	63.542	92.847	11.607
29	2CH DO FB EER	21.944	30.833	26.389	78.056	69.167	8.744
30	SM ST NO AER_{min}	42.431	11.528	26.979	57.569	88.472	4.705
31	2CH DO FB AER_{min}	30.903	27.083	28.993	69.097	72.917	7.348
32	SM ST CL AER_{min}	55.625	5.417	30.521	44.375	94.583	7.196
33	SM BN FB EER	40.833	27.778	34.306	59.167	72.222	8.214
34	SM BD CL EER	70.139	2.639	36.389	29.861	97.361	5.911
35	SM BD NO EER	72.569	1.111	36.84	27.431	98.889	3.74
36	SM BN FB AER_{min}	74.653	2.222	38.437	25.347	97.778	5.007
37	SM BD CL AER_{min}	76.736	0.417	38.576	23.264	99.583	6.118
38	SM BD NO AER_{min}	76.944	0.556	38.75	23.056	99.444	3.566
39	SM BD FB EER	76.875	1.111	38.993	23.125	98.889	7.785
40	SM ST CL EER	50.694	28.889	39.792	49.306	71.111	14.561
41	SM BD FB AER_{min}	79.444	1.111	40.278	20.556	98.889	4.784
42	SM BN CL AER_{min}	82.847	3.056	42.952	17.153	96.944	5.928
43	SM ST NO EER	39.722	48.056	43.889	60.278	51.944	10.594
44	SM BN NO AER_{min}	88.333	0.139	44.236	11.667	99.861	3.591
45	SM DO FB AER_{min}	43.264	47.222	45.243	56.736	52.778	5.864
46	SM ST NO FRR_{zero}	35.347	55.556	45.451	64.653	44.444	6.008
47	SM DO FB FAR_{zero}	12.569	84.444	48.507	87.431	15.556	4.223
48	SM DO NO FRR_{zero}	97.431	0.0	48.715	2.569	100.0	1.362
49	SM ST FB FRR_{zero}	54.375	44.583	49.479	45.625	55.417	0.807
50	SM ST CL FRR_{zero}	99.722	0.0	49.861	0.278	100.0	0.393
51	SM DO CL FRR_{zero}	88.681	11.111	49.896	11.319	88.889	0.295
52	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
53	2CH ST NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
54	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

55	2CH BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
56	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
57	2CH DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
58	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
59	2CH BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
60	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
61	2CH ST CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
62	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
63	2CH BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
64	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
65	2CH DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
66	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
67	2CH BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
68	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH ST FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
70	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
72	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH DO FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
74	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
76	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
77	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
79	SM DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
82	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
83	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
85	SM DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
87	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
88	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
90	SM BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
91	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0

93	SM DO FB FRR_{zero}	19.583	85.556	52.569	80.417	14.444	4.892
94	SM BN CL EER	43.056	62.917	52.986	56.944	37.083	7.887
95	SM DO FB EER	65.486	42.222	53.854	34.514	57.778	8.394
96	SM BN NO EER	57.986	66.111	62.049	42.014	33.889	3.926

A.4 The augmented benchmark experiment

The augmented BME is an additional experiment conducted in this study for the purpose of gauging the proficiency of the proposed systems within the context of the BME in the event that the data augmentation strategies proposed in Sections 7.3.1 and 7.3.2 are employed in order to generate a sufficient number of *actual* hand vein samples. This experiment is motivated by the notion that the number of available samples within the context of the BME is arguably insufficient for the task at hand, especially for the Wilches database, in which case *only* 960 training pairs are available (see Section 7.5.1.1).

The results for all 96 systems within the context of the Bosphorus and Wilches databases and the augmented BME are depicted in Tables A.6 and A.7 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Table A.6: The results of all 96 systems ranked according to the AER within the context of the augmented BME and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	SM BD NO AER_{min}	26.3	20.52	23.41	73.7	79.48	3.089
2	SM BD NO EER	24.551	22.738	23.644	75.449	77.262	2.92
3	2CH BN NO EER	22.196	26.087	24.141	77.804	73.913	6.176
4	2CH BN NO AER_{min}	22.1	26.627	24.363	77.9	73.373	6.183
5	2CH BD NO EER	22.927	27.74	25.333	77.073	72.26	5.432
6	SM DO NO EER	15.849	34.849	25.349	84.151	65.151	3.202
7	SM DO NO AER_{min}	15.849	34.849	25.349	84.151	65.151	3.202
8	2CH BD NO AER_{min}	22.949	27.758	25.353	77.051	72.242	5.21
9	SM ST NO AER_{min}	28.829	23.398	26.113	71.171	76.602	2.017
10	SM BN NO AER_{min}	33.896	18.556	26.226	66.104	81.444	3.011
11	2CH BN CL EER	24.542	27.916	26.229	75.458	72.084	5.912
12	2CH BN CL AER_{min}	23.822	29.24	26.531	76.178	70.76	6.067
13	SM ST CL AER_{min}	27.6	25.627	26.613	72.4	74.373	2.364
14	2CH BD CL EER	25.313	28.504	26.909	74.687	71.496	6.966
15	SM DO CL AER_{min}	22.044	32.024	27.034	77.956	67.976	2.022

16	SM ST NO EER	27.184	27.002	27.093	72.816	72.998	2.567
17	SM DO CL EER	21.731	32.498	27.114	78.269	67.502	1.947
18	SM ST CL EER	26.022	28.776	27.399	73.978	71.224	2.486
19	SM BD CL AER_{min}	31.527	23.422	27.474	68.473	76.578	1.996
20	SM BN CL AER_{min}	37.331	17.638	27.484	62.669	82.362	2.931
21	2CH BD CL AER_{min}	23.589	31.553	27.571	76.411	68.447	6.708
22	SM BD CL EER	27.58	27.718	27.649	72.42	72.282	1.867
23	SM BN CL EER	29.113	27.12	28.117	70.887	72.88	2.657
24	SM BN NO EER	27.671	28.753	28.212	72.329	71.247	3.69
25	SM ST FB AER_{min}	28.891	30.273	29.582	71.109	69.727	1.932
26	2CH BN FB EER	23.709	35.589	29.649	76.291	64.411	4.1
27	2CH BN FB AER_{min}	27.402	31.993	29.698	72.598	68.007	4.32
28	2CH ST NO EER	26.764	32.687	29.726	73.236	67.313	6.878
29	2CH ST NO AER_{min}	30.936	28.798	29.867	69.064	71.202	7.259
30	SM BD FB AER_{min}	33.029	27.242	30.136	66.971	72.758	3.611
31	SM ST FB EER	26.836	33.462	30.149	73.164	66.538	2.158
32	SM BD FB EER	30.624	29.798	30.211	69.376	70.202	3.529
33	SM DO FB EER	20.973	39.473	30.223	79.027	60.527	1.77
34	SM DO FB AER_{min}	20.973	39.473	30.223	79.027	60.527	1.77
35	SM BN FB AER_{min}	35.113	26.513	30.813	64.887	73.487	2.056
36	SM BN FB EER	30.398	32.122	31.26	69.602	67.878	1.423
37	2CH BD FB AER_{min}	28.658	37.844	33.251	71.342	62.156	5.312
38	2CH BD FB EER	29.236	37.282	33.259	70.764	62.718	5.121
39	2CH ST CL EER	37.709	33.282	35.496	62.291	66.718	4.471
40	2CH DO NO AER_{min}	34.533	36.896	35.714	65.467	63.104	6.14
41	2CH DO NO EER	36.564	35.002	35.783	63.436	64.998	6.234
42	2CH ST CL AER_{min}	41.642	31.084	36.363	58.358	68.916	4.476
43	2CH DO CL EER	37.604	37.849	37.727	62.396	62.151	7.291
44	2CH DO CL AER_{min}	43.96	32.222	38.091	56.04	67.778	7.933
45	2CH ST FB AER_{min}	32.931	45.349	39.14	67.069	54.651	6.313
46	2CH ST FB EER	38.464	40.827	39.646	61.536	59.173	6.412
47	2CH DO FB AER_{min}	47.689	47.636	47.662	52.311	52.364	5.01
48	2CH DO FB EER	46.42	49.464	47.942	53.58	50.536	5.604
49	SM ST NO FRR_{zero}	9.971	88.889	49.43	90.029	11.111	1.612
50	SM ST FB FRR_{zero}	65.884	33.449	49.667	34.116	66.551	0.487
51	SM ST CL FRR_{zero}	10.82	88.889	49.854	89.18	11.111	0.412
52	SM DO NO FRR_{zero}	77.776	22.222	49.999	22.224	77.778	0.003
53	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

54	2CH ST NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
55	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
56	2CH BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
57	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
58	2CH DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
59	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
60	2CH BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
61	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
62	2CH ST CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
63	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
64	2CH BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
65	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
66	2CH DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
67	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
68	2CH BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
69	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH ST FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
71	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
72	2CH BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
73	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH DO FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
75	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
77	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
80	SM DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
82	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
83	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
85	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
86	SM DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
87	SM DO CL FRR_{zero}	88.889	11.111	50.0	11.111	88.889	0.0
88	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
90	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

92	SM BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
93	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
94	SM DO FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
95	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
96	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0

Table A.7: The results of all 96 systems ranked according to the AER within the context of the augmented BME and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH BD CL EER	7.242	12.236	9.739	92.758	87.764	5.104
2	2CH BD CL AER_{min}	6.742	12.876	9.809	93.258	87.124	5.283
3	2CH BD NO AER_{min}	10.478	9.944	10.211	89.522	90.056	3.688
4	2CH BD NO EER	9.876	10.629	10.252	90.124	89.371	3.346
5	2CH BN NO AER_{min}	10.327	11.047	10.687	89.673	88.953	3.998
6	2CH BN NO EER	9.891	11.504	10.698	90.109	88.496	3.991
7	2CH BN CL AER_{min}	11.753	12.433	12.093	88.247	87.567	6.301
8	2CH BN CL EER	10.747	13.493	12.12	89.253	86.507	6.4
9	2CH BN FB AER_{min}	13.444	15.349	14.397	86.556	84.651	5.052
10	SM BD NO AER_{min}	17.051	11.891	14.471	82.949	88.109	2.438
11	2CH BN FB EER	13.858	15.207	14.532	86.142	84.793	5.383
12	2CH BD FB AER_{min}	14.953	14.556	14.754	85.047	85.444	5.129
13	2CH BD FB EER	15.758	14.036	14.897	84.242	85.964	4.894
14	SM BD NO EER	14.573	15.611	15.092	85.427	84.389	2.845
15	2CH ST NO EER	14.129	16.229	15.179	85.871	83.771	4.933
16	2CH ST NO AER_{min}	15.333	15.076	15.204	84.667	84.924	4.565
17	SM DO NO EER	10.324	25.907	18.116	89.676	74.093	2.136
18	SM DO NO AER_{min}	10.324	25.907	18.116	89.676	74.093	2.136
19	SM ST CL AER_{min}	22.02	17.573	19.797	77.98	82.427	2.614
20	SM DO CL EER	9.798	30.744	20.271	90.202	69.256	2.833
21	SM DO CL AER_{min}	9.798	30.744	20.271	90.202	69.256	2.833
22	SM BD CL AER_{min}	26.231	14.596	20.413	73.769	85.404	5.698
23	SM ST NO AER_{min}	20.967	19.953	20.46	79.033	80.047	2.446
24	2CH ST CL EER	22.311	18.762	20.537	77.689	81.238	7.554
25	SM BD CL EER	23.587	17.622	20.604	76.413	82.378	5.344
26	SM ST FB AER_{min}	22.418	19.629	21.023	77.582	80.371	3.163

27	2CH ST CL AER_{min}	25.453	17.12	21.287	74.547	82.88	8.517
28	SM ST CL EER	20.989	21.904	21.447	79.011	78.096	3.394
29	SM BN NO AER_{min}	35.478	8.902	22.19	64.522	91.098	2.62
30	SM BD FB AER_{min}	22.129	22.784	22.457	77.871	77.216	3.609
31	SM ST FB EER	20.144	26.578	23.361	79.856	73.422	3.66
32	SM ST NO EER	19.791	26.951	23.371	80.209	73.049	4.775
33	SM BD FB EER	19.356	27.664	23.51	80.644	72.336	3.134
34	2CH DO NO EER	26.253	21.649	23.951	73.747	78.351	11.893
35	2CH DO NO AER_{min}	25.304	22.942	24.123	74.696	77.058	12.025
36	SM DO FB EER	11.84	39.084	25.462	88.16	60.916	4.022
37	SM DO FB AER_{min}	11.84	39.084	25.462	88.16	60.916	4.022
38	SM BN FB AER_{min}	35.958	17.198	26.578	64.042	82.802	4.185
39	SM BN CL AER_{min}	38.762	15.682	27.222	61.238	84.318	2.812
40	SM BN NO EER	27.12	27.389	27.254	72.88	72.611	4.166
41	SM BN FB EER	29.918	26.651	28.284	70.082	73.349	3.196
42	2CH ST FB AER_{min}	28.733	27.913	28.323	71.267	72.087	8.758
43	2CH ST FB EER	30.636	26.72	28.678	69.364	73.28	9.105
44	SM BN CL EER	31.271	26.871	29.071	68.729	73.129	2.381
45	2CH DO CL AER_{min}	38.602	25.967	32.284	61.398	74.033	9.532
46	2CH DO CL EER	26.233	38.607	32.42	73.767	61.393	8.722
47	2CH DO FB AER_{min}	52.309	34.338	43.323	47.691	65.662	11.604
48	2CH DO FB EER	43.942	43.171	43.557	56.058	56.829	11.973
49	SM ST FB FRR_{zero}	29.013	66.76	47.887	70.987	33.24	3.114
50	SM ST NO FRR_{zero}	8.078	89.027	48.552	91.922	10.973	4.095
51	SM ST CL FRR_{zero}	8.551	88.911	48.731	91.449	11.089	3.589
52	SM DO NO FRR_{zero}	99.54	0.0	49.77	0.46	100.0	0.345
53	SM DO CL FRR_{zero}	99.729	0.0	49.864	0.271	100.0	0.331
54	SM DO FB FRR_{zero}	99.916	0.011	49.963	0.084	99.989	0.064
55	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
56	2CH ST NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
57	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
58	2CH BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
59	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
60	2CH DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
61	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
62	2CH BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
63	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
64	2CH ST CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0

65	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
66	2CH BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
67	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
68	2CH DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
69	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
71	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
72	2CH ST FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
73	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
75	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH DO FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
77	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	2CH BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
79	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
82	SM DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
83	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
85	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
87	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
88	SM DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
90	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
91	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
93	SM BN FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
94	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
95	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
96	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0

When the results depicted in Tables A.4 and A.5 are compared to the results depicted in Tables A.6 and A.7, it is clear that the proficiency of the proposed systems is significantly reduced within the context of the augmented BME. The networks invariably become *more* biased towards the training individuals within the context of the augmented BME due to (1) the increased number of training samples associated with the *same* insufficient number of *different* individuals and (2) the increased intraclass variance between samples associated with the same individual. This further strengthens the fact that the *main* problem within the context of the IIS is the availability of a sufficient number of *different individuals*, and *not* merely the availability of a sufficient number of *different samples* associated with a non-representative set of individuals.

A.5 The first GenVeins experiment

The results for all 96 systems within the context of the Bosphorus and Wilches databases and the FGE are depicted in Tables A.8 and A.9 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Table A.8: The results of all 96 systems ranked according to the AER within the context of the FGE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	System Design	FAR (%)	FRR (%)	AER (%)	SPE (%)	SEN (%)	STD (%)
1	2CH DO NO AER_{min}	4.542	25.669	15.105	95.458	74.331	0.679
2	2CH ST NO AER_{min}	7.862	22.64	15.251	92.138	77.36	0.757
3	2CH BD CL AER_{min}	9.809	21.336	15.573	90.191	78.664	0.698
4	2CH DO NO FRR_{zero}	15.219	16.656	15.937	84.781	83.344	2.754
5	2CH BD CL EER	5.216	26.818	16.017	94.784	73.182	0.916
6	2CH ST NO EER	4.151	28.012	16.082	95.849	71.988	0.79
7	2CH BN CL EER	13.345	18.955	16.15	86.655	81.045	0.631
8	2CH BN CL AER_{min}	20.858	14.831	17.844	79.142	85.169	1.236
9	2CH BD FB FRR_{zero}	5.697	30.553	18.125	94.303	69.447	0.79
10	2CH DO CL FRR_{zero}	2.099	34.194	18.146	97.901	65.806	1.823
11	2CH BD CL FRR_{zero}	21.861	14.618	18.239	78.139	85.382	3.199
12	2CH DO NO EER	1.431	35.836	18.633	98.569	64.164	1.074
13	2CH ST CL FRR_{zero}	14.377	25.401	19.889	85.623	74.599	10.672
14	2CH DO CL AER_{min}	1.075	38.917	19.996	98.925	61.083	1.27
15	2CH ST NO FRR_{zero}	28.222	12.843	20.532	71.778	87.157	10.72
16	2CH ST CL AER_{min}	0.908	41.48	21.194	99.092	58.52	1.752
17	2CH BN FB AER_{min}	1.849	40.963	21.406	98.151	59.037	1.129
18	2CH BD FB AER_{min}	1.294	43.789	22.541	98.706	56.211	1.661
19	2CH BN FB EER	1.218	44.789	23.004	98.782	55.211	0.759
20	SM DO NO EER	26.527	20.088	23.307	73.473	79.912	2.492
21	SM DO NO AER_{min}	28.061	18.605	23.333	71.939	81.395	2.558
22	SM BD NO AER_{min}	22.801	24.215	23.508	77.199	75.785	1.968
23	2CH DO CL EER	0.409	46.793	23.601	99.591	53.207	0.973

24	2CH ST FB AER_{min}	1.078	47.088	24.083	98.922	52.912	1.179
25	2CH ST CL EER	0.396	47.922	24.159	99.604	52.078	1.576
26	2CH DO FB AER_{min}	0.763	47.786	24.274	99.237	52.214	1.059
27	2CH BD FB EER	0.807	47.885	24.346	99.193	52.115	0.62
28	SM BD CL AER_{min}	17.103	32.427	24.765	82.897	67.573	1.162
29	2CH BN CL FRR_{zero}	41.028	8.76	24.894	58.972	91.24	4.544
30	2CH ST FB EER	0.901	49.107	25.004	99.099	50.893	1.714
31	2CH BN FB FRR_{zero}	27.035	23.338	25.186	72.965	76.662	13.292
32	SM DO CL EER	8.198	43.203	25.701	91.802	56.797	0.897
33	SM DO CL AER_{min}	8.198	43.203	25.701	91.802	56.797	0.897
34	2CH DO FB EER	0.46	51.576	26.018	99.54	48.424	0.878
35	SM BD FB AER_{min}	10.131	45.34	27.735	89.869	54.66	1.812
36	SM BD NO EER	10.195	46.887	28.541	89.805	53.113	4.07
37	SM BD CL EER	9.191	48.441	28.816	90.809	51.559	1.898
38	2CH DO FB FRR_{zero}	36.698	22.381	29.539	63.302	77.619	14.503
39	SM BD FB EER	6.97	53.435	30.203	93.03	46.565	1.097
40	2CH BD NO EER	57.823	3.854	30.838	42.177	96.146	4.802
41	SM ST FB AER_{min}	15.755	48.847	32.301	84.245	51.153	1.041
42	2CH ST FB FRR_{zero}	47.64	17.96	32.8	52.36	82.04	15.419
43	SM ST CL AER_{min}	35.166	31.876	33.521	64.834	68.124	2.533
44	SM ST FB EER	13.968	55.469	34.719	86.032	44.531	0.929
45	SM ST CL EER	32.914	37.493	35.204	67.086	62.507	1.967
46	SM BN NO AER_{min}	30.984	39.806	35.395	69.016	60.194	6.786
47	SM ST NO AER_{min}	42.202	28.887	35.544	57.798	71.113	2.712
48	SM BN CL AER_{min}	13.261	58.401	35.831	86.739	41.599	3.939
49	2CH BD NO AER_{min}	70.774	2.218	36.496	29.226	97.782	4.415
50	2CH BN NO EER	76.473	1.65	39.061	23.527	98.35	2.059
51	SM ST NO EER	38.29	40.692	39.491	61.71	59.308	1.296
52	SM BN FB AER_{min}	27.889	51.232	39.561	72.111	48.768	2.108
53	SM BN FB EER	26.555	55.851	41.203	73.445	44.149	1.49
54	SM BN FB FRR_{zero}	74.322	8.377	41.35	25.678	91.623	7.805
55	SM ST CL FRR_{zero}	77.417	5.351	41.384	22.583	94.649	9.72
56	SM BN CL EER	10.05	73.053	41.552	89.95	26.947	0.942
57	2CH BN NO AER_{min}	82.745	1.123	41.934	17.255	98.877	1.941
58	2CH BD NO FRR_{zero}	83.462	1.125	42.294	16.538	98.875	5.479
59	SM ST NO FRR_{zero}	82.605	3.164	42.884	17.395	96.836	8.024
60	SM BN NO EER	26.495	61.365	43.93	73.505	38.635	2.318
61	SM BD FB FRR_{zero}	81.831	6.881	44.356	18.169	93.119	10.561

62	SM DO FB AER_{min}	42.12	50.232	46.176	57.88	49.768	4.276
63	2CH BN NO FRR_{zero}	93.444	0.324	46.884	6.556	99.676	2.002
64	SM BD CL FRR_{zero}	92.845	1.753	47.299	7.155	98.247	7.64
65	SM ST FB FRR_{zero}	91.329	3.479	47.404	8.671	96.521	7.316
66	SM DO FB EER	42.12	53.069	47.594	57.88	46.931	5.76
67	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
68	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
72	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
77	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
82	SM DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
83	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
85	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
87	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
88	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
90	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
93	SM DO CL FRR_{zero}	100.0	0.002	50.001	0.0	99.998	0.003
94	SM DO FB FRR_{zero}	33.333	66.669	50.001	66.667	33.331	0.003
95	SM DO CL FAR_{zero}	0.122	99.925	50.024	99.878	0.075	0.04
96	SM DO NO FAR_{zero}	0.179	99.945	50.062	99.821	0.055	0.082

Table A.9: The results of all 96 systems ranked according to the AER within the context of the FGE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	2CH ST NO EER	1.885	6.657	4.271	98.115	93.343	0.447
2	2CH DO NO AER_{min}	2.907	5.893	4.4	97.093	94.107	0.751
3	2CH ST NO AER_{min}	4.177	4.782	4.479	95.823	95.218	0.645
4	2CH DO NO EER	0.823	9.018	4.921	99.177	90.982	0.798
5	2CH DO FB AER_{min}	2.202	8.105	5.154	97.798	91.895	0.51
6	2CH BN FB EER	2.996	7.55	5.273	97.004	92.45	0.596
7	2CH BD FB AER_{min}	2.708	7.956	5.332	97.292	92.044	0.436
8	2CH BN FB AER_{min}	4.434	6.359	5.397	95.566	93.641	0.631
9	2CH DO FB EER	1.24	9.633	5.437	98.76	90.367	0.697
10	2CH BD FB EER	1.567	9.712	5.64	98.433	90.288	0.564
11	2CH ST FB AER_{min}	4.206	7.302	5.754	95.794	92.698	0.583
12	2CH ST FB EER	3.403	8.234	5.818	96.597	91.766	0.438
13	2CH DO CL EER	8.661	3.115	5.888	91.339	96.885	0.774
14	2CH DO NO FRR_{zero}	11.032	3.294	7.163	88.968	96.706	4.036
15	2CH BD CL EER	12.55	2.976	7.763	87.45	97.024	0.549
16	2CH ST CL EER	12.946	2.827	7.887	87.054	97.173	1.757
17	2CH BD FB FRR_{zero}	12.688	3.482	8.085	87.312	96.518	1.853
18	2CH DO CL AER_{min}	15.873	2.113	8.993	84.127	97.887	2.094
19	SM DO CL EER	8.829	9.385	9.107	91.171	90.615	1.156
20	SM DO CL AER_{min}	8.829	9.385	9.107	91.171	90.615	1.156
21	2CH BN CL EER	17.193	2.113	9.653	82.807	97.887	1.834
22	2CH ST CL AER_{min}	18.621	1.964	10.293	81.379	98.036	2.105
23	2CH BD CL AER_{min}	19.772	2.143	10.957	80.228	97.857	1.655
24	SM BD CL EER	9.653	13.492	11.573	90.347	86.508	1.117
25	SM BD CL AER_{min}	17.639	5.933	11.786	82.361	94.067	1.222
26	2CH DO CL FRR_{zero}	22.321	1.538	11.929	77.679	98.462	4.384
27	2CH ST NO FRR_{zero}	23.73	1.915	12.823	76.27	98.085	13.481
28	SM BD FB AER_{min}	13.542	12.738	13.14	86.458	87.262	1.093
29	2CH BN CL AER_{min}	25.089	1.419	13.254	74.911	98.581	2.505
30	SM BD FB EER	8.968	19.583	14.276	91.032	80.417	1.148
31	SM BD NO AER_{min}	23.591	5.774	14.683	76.409	94.226	3.197
32	SM BD NO EER	10.347	19.147	14.747	89.653	80.853	3.322
33	SM DO NO EER	27.877	4.365	16.121	72.123	95.635	2.3

34	SM DO NO AER_{min}	28.899	3.889	16.394	71.101	96.111	2.562
35	2CH BD CL FRR_{zero}	35.297	1.2	18.249	64.703	98.8	4.976
36	2CH BN FB FRR_{zero}	35.427	2.183	18.804	64.573	97.817	16.819
37	SM BN CL AER_{min}	19.335	23.552	21.443	80.665	76.448	4.25
38	2CH BD NO EER	43.046	0.903	21.974	56.954	99.097	5.23
39	2CH DO FB FRR_{zero}	41.696	2.44	22.068	58.304	97.56	19.823
40	2CH ST CL FRR_{zero}	43.998	0.883	22.441	56.002	99.117	10.155
41	SM ST FB AER_{min}	28.73	17.421	23.075	71.27	82.579	2.13
42	SM BN NO AER_{min}	29.078	17.718	23.398	70.922	82.282	8.693
43	2CH BN CL FRR_{zero}	46.944	0.794	23.869	53.056	99.206	5.437
44	SM ST FB EER	25.02	26.945	25.982	74.98	73.055	2.06
45	SM BN FB AER_{min}	30.516	24.206	27.361	69.484	75.794	2.623
46	2CH ST FB FRR_{zero}	53.542	1.865	27.703	46.458	98.135	20.153
47	SM ST CL AER_{min}	42.262	15.218	28.74	57.738	84.782	2.543
48	2CH BD NO AER_{min}	57.818	0.456	29.137	42.182	99.544	4.966
49	SM BN FB EER	28.76	31.845	30.303	71.24	68.155	1.676
50	SM ST NO AER_{min}	47.689	12.956	30.322	52.311	87.044	2.437
51	SM BN CL EER	15.06	45.794	30.426	84.94	54.206	1.335
52	SM ST CL EER	39.415	25.397	32.406	60.585	74.603	1.841
53	2CH BN NO EER	65.952	0.208	33.08	34.048	99.792	3.166
54	SM BN NO EER	24.693	45.595	35.144	75.307	54.405	3.274
55	SM ST NO EER	43.006	28.948	35.977	56.994	71.052	1.375
56	2CH BN NO AER_{min}	74.544	0.139	37.341	25.456	99.861	2.553
57	2CH BD NO FRR_{zero}	76.26	0.208	38.234	23.74	99.792	7.638
58	SM BN FB FRR_{zero}	81.141	0.794	40.967	18.859	99.206	8.663
59	SM ST CL FRR_{zero}	82.857	0.675	41.766	17.143	99.325	9.392
60	SM DO FB AER_{min}	43.859	40.02	41.939	56.141	59.98	5.731
61	SM BD FB FRR_{zero}	83.8	0.893	42.346	16.2	99.107	14.326
62	SM ST NO FRR_{zero}	86.181	0.317	43.249	13.819	99.683	7.695
63	2CH BN NO FRR_{zero}	90.615	0.08	45.347	9.385	99.92	3.063
64	SM DO FB EER	43.859	48.353	46.106	56.141	51.647	11.608
65	SM BD CL FRR_{zero}	92.202	0.298	46.25	7.798	99.702	10.607
66	SM ST FB FRR_{zero}	93.641	0.456	47.049	6.359	99.544	8.348
67	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
68	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

72	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
77	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BN NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
82	SM DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
83	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BD NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
85	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
87	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
88	SM DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
89	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
90	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
93	SM DO FB FRR_{zero}	33.333	66.667	50.0	66.667	33.333	0.0
94	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
95	SM DO CL FAR_{zero}	0.099	100.0	50.05	99.901	0.0	0.11
96	SM DO NO FAR_{zero}	0.784	99.94	50.362	99.216	0.06	0.357

A.6 The second GenVeins experiment

The results for all 96 systems within the context of the Bosphorus and Wilches databases and the SGE are depicted in Tables A.10 and A.11 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Table A.10: The results of all 96 systems ranked according to the AER within the context of the SGE and the Bosphorus database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**

1	2CH DO CL AER_{min}	4.462	9.593	7.028	95.538	90.407	0.398
2	2CH ST CL AER_{min}	4.439	10.291	7.365	95.561	89.709	0.546
3	2CH DO CL EER	2.12	13.453	7.786	97.88	86.547	0.77
4	2CH DO NO EER	9.978	5.655	7.816	90.022	94.345	1.438
5	2CH ST CL EER	2.403	13.442	7.923	97.597	86.558	0.771
6	2CH BN FB AER_{min}	5.074	12.52	8.797	94.926	87.48	0.689
7	2CH BD FB AER_{min}	4.125	13.542	8.833	95.875	86.458	0.684
8	2CH BN FB EER	3.266	15.139	9.203	96.734	84.861	0.841
9	2CH BD FB EER	2.448	16.194	9.321	97.552	83.806	0.68
10	2CH DO FB AER_{min}	2.609	17.03	9.82	97.391	82.97	0.899
11	2CH ST FB AER_{min}	3.299	16.66	9.979	96.701	83.34	0.817
12	2CH ST FB EER	2.128	19.372	10.75	97.872	80.628	1.138
13	2CH DO FB EER	1.565	20.268	10.916	98.435	79.732	1.02
14	2CH ST NO EER	18.628	3.548	11.088	81.372	96.452	2.425
15	2CH BD FB FRR_{zero}	18.611	6.22	12.416	81.389	93.78	2.421
16	2CH BD CL EER	21.473	4.087	12.78	78.527	95.913	1.756
17	2CH DO NO AER_{min}	22.247	3.317	12.782	77.753	96.683	3.661
18	2CH ST NO AER_{min}	26.528	2.563	14.545	73.472	97.437	2.854
19	2CH DO CL FRR_{zero}	25.256	4.57	14.913	74.744	95.43	12.599
20	2CH ST CL FRR_{zero}	25.238	4.891	15.065	74.762	95.109	12.683
21	SM DO CL EER	16.558	14.484	15.521	83.442	85.516	1.513
22	SM DO CL AER_{min}	17.165	14.087	15.626	82.835	85.913	1.684
23	SM BD CL AER_{min}	19.296	13.274	16.285	80.704	86.726	1.001
24	SM BD FB AER_{min}	17.051	17.1	17.076	82.949	82.9	0.803
25	SM BD CL EER	10.923	24.289	17.606	89.077	75.711	1.575
26	SM BD FB EER	11.739	24.154	17.946	88.261	75.846	1.503
27	2CH BD CL AER_{min}	35.962	2.589	19.276	64.038	97.411	2.416
28	SM BD NO EER	17.949	22.503	20.226	82.051	77.497	3.22
29	SM BD NO AER_{min}	34.306	6.786	20.546	65.694	93.214	2.251
30	SM DO FB AER_{min}	19.025	23.366	21.196	80.975	76.634	6.695
31	SM BN CL AER_{min}	22.929	19.795	21.362	77.071	80.205	2.718
32	2CH DO NO FRR_{zero}	42.574	1.478	22.026	57.426	98.522	4.781
33	2CH BN CL EER	42.245	2.335	22.29	57.755	97.665	1.788
34	SM DO FB EER	15.533	29.527	22.53	84.467	70.473	6.218
35	SM DO NO EER	44.938	4.064	24.501	55.062	95.936	3.265
36	SM DO NO AER_{min}	44.938	4.064	24.501	55.062	95.936	3.265
37	2CH BN FB FRR_{zero}	46.572	4.124	25.348	53.428	95.876	17.477
38	SM ST FB AER_{min}	39.05	14.038	26.544	60.95	85.962	1.443

39	SM ST FB EER	36.317	17.874	27.095	63.683	82.126	1.552
40	2CH BN CL AER_{min}	55.183	1.531	28.357	44.817	98.469	2.607
41	2CH BD CL FRR_{zero}	55.967	1.399	28.683	44.033	98.601	4.296
42	2CH ST NO FRR_{zero}	57.775	0.863	29.319	42.225	99.137	11.287
43	SM BN CL EER	11.721	48.003	29.862	88.279	51.997	1.231
44	SM BN NO AER_{min}	43.715	17.014	30.365	56.285	82.986	4.291
45	SM BN FB AER_{min}	35.441	27.659	31.55	64.559	72.341	4.565
46	SM ST CL AER_{min}	54.529	10.367	32.448	45.471	89.633	1.912
47	SM ST CL EER	52.098	13.287	32.693	47.902	86.713	1.94
48	2CH ST FB FRR_{zero}	63.262	3.294	33.278	36.738	96.706	18.707
49	SM BN FB EER	32.904	34.524	33.714	67.096	65.476	3.901
50	SM BN NO EER	36.905	33.032	34.969	63.095	66.968	1.756
51	SM ST NO AER_{min}	65.027	6.313	35.67	34.973	93.687	1.608
52	SM ST NO EER	60.794	11.934	36.364	39.206	88.066	1.568
53	2CH DO FB FRR_{zero}	69.479	3.37	36.424	30.521	96.63	19.2
54	2CH BN CL FRR_{zero}	76.84	0.615	38.728	23.16	99.385	5.94
55	SM ST FB FRR_{zero}	82.626	1.561	42.094	17.374	98.439	9.106
56	SM ST CL FRR_{zero}	83.968	1.052	42.51	16.032	98.948	6.892
57	SM BN FB FRR_{zero}	85.68	1.445	43.562	14.32	98.555	9.242
58	SM ST NO FRR_{zero}	89.397	0.559	44.978	10.603	99.441	4.583
59	2CH BD NO EER	90.325	0.192	45.258	9.675	99.808	1.551
60	2CH BD NO AER_{min}	94.608	0.109	47.359	5.392	99.891	1.317
61	2CH BN NO EER	95.203	0.136	47.669	4.797	99.864	0.601
62	SM BD NO FRR_{zero}	96.29	0.136	48.213	3.71	99.864	5.055
63	2CH BN NO AER_{min}	97.143	0.083	48.613	2.857	99.917	0.455
64	SM BN NO FRR_{zero}	97.852	0.109	48.981	2.148	99.891	2.883
65	2CH BD NO FRR_{zero}	97.954	0.03	48.992	2.046	99.97	1.131
66	2CH BN NO FRR_{zero}	99.687	0.0	49.844	0.313	100.0	0.108
67	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
68	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
72	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0

77	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
82	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
83	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
85	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
86	SM DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
87	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
88	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
89	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
90	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM DO FB FRR_{zero}	88.889	11.111	50.0	11.111	88.889	0.0
93	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
94	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
95	SM DO CL FAR_{zero}	0.367	99.792	50.079	99.633	0.208	0.075
96	SM DO NO FAR_{zero}	1.632	99.226	50.429	98.368	0.774	0.647

Table A.11: The results of all 96 systems ranked according to the AER within the context of the SGE and the Wilches database. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE, preprocessing protocol and probabilistic threshold criterion. Refer to Table A.1 for the descriptions of the abbreviations.

Rank	**System Design**	**FAR (%)**	**FRR (%)**	**AER (%)**	**SPE (%)**	**SEN (%)**	**STD (%)**
1	2CH DO NO EER	3.228	0.0	1.614	96.772	100.0	0.751
2	2CH ST FB EER	4.074	0.0	2.037	95.926	100.0	1.215
3	2CH DO FB EER	5.026	0.0	2.513	94.974	100.0	1.425
4	2CH BD FB EER	6.138	0.0	3.069	93.862	100.0	1.461
5	2CH BN FB EER	6.667	0.0	3.333	93.333	100.0	1.728
6	2CH ST NO EER	6.984	0.0	3.492	93.016	100.0	0.761
7	2CH ST FB AER_{min}	7.09	0.0	3.545	92.91	100.0	1.613
8	2CH DO FB AER_{min}	7.936	0.0	3.968	92.064	100.0	1.796
9	SM BN CL EER	8.73	0.0	4.365	91.27	100.0	3.389
10	2CH DO NO AER_{min}	9.206	0.0	4.603	90.794	100.0	2.772
11	2CH BD FB AER_{min}	9.788	0.0	4.894	90.212	100.0	2.459

12	2CH BN FB AER_{min}	10.159	0.0	5.079	89.841	100.0	1.71
13	2CH ST NO AER_{min}	11.587	0.0	5.794	88.413	100.0	1.285
14	SM DO CL EER	12.699	0.0	6.349	87.301	100.0	2.564
15	SM BD NO EER	12.857	0.0	6.429	87.143	100.0	1.928
16	SM DO CL AER_{min}	13.386	0.0	6.693	86.614	100.0	2.84
17	SM BD FB EER	15.132	0.0	7.566	84.868	100.0	2.939
18	SM BD CL EER	15.979	0.0	7.989	84.021	100.0	2.15
19	SM DO FB EER	17.354	0.0	8.677	82.646	100.0	3.358
20	SM BN CL AER_{min}	17.831	0.0	8.915	82.169	100.0	4.129
21	2CH DO CL EER	20.847	0.0	10.423	79.153	100.0	2.988
22	SM BD FB AER_{min}	21.217	0.0	10.609	78.783	100.0	3.569
23	2CH DO NO FRR_{zero}	21.534	0.0	10.767	78.466	100.0	2.588
24	SM DO FB AER_{min}	21.905	0.0	10.953	78.095	100.0	3.367
25	2CH ST CL EER	23.704	0.0	11.852	76.296	100.0	2.984
26	SM BD CL AER_{min}	23.757	0.0	11.878	76.243	100.0	2.757
27	SM BD NO AER_{min}	25.185	0.0	12.592	74.815	100.0	3.62
28	2CH BD CL EER	26.72	0.0	13.36	73.28	100.0	4.027
29	SM BN FB EER	28.148	0.0	14.074	71.852	100.0	2.621
30	2CH DO CL AER_{min}	30.794	0.0	15.397	69.206	100.0	3.352
31	SM BN FB AER_{min}	31.746	0.0	15.873	68.254	100.0	4.435
32	2CH ST CL AER_{min}	31.958	0.0	15.979	68.042	100.0	2.293
33	2CH BD FB FRR_{zero}	32.328	0.0	16.164	67.672	100.0	5.393
34	SM BN NO EER	34.497	0.0	17.249	65.503	100.0	5.692
35	2CH BN CL EER	35.555	0.0	17.778	64.445	100.0	2.097
36	SM ST FB EER	37.407	0.0	18.704	62.593	100.0	3.479
37	SM DO NO EER	37.566	0.0	18.783	62.434	100.0	3.762
38	SM DO NO AER_{min}	37.566	0.0	18.783	62.434	100.0	3.762
39	SM BN NO AER_{min}	39.63	0.0	19.815	60.37	100.0	5.691
40	SM ST FB AER_{min}	41.323	0.0	20.661	58.677	100.0	2.911
41	2CH BD CL AER_{min}	41.429	0.0	20.714	58.571	100.0	4.186
42	SM ST CL EER	41.535	0.0	20.767	58.465	100.0	2.837
43	2CH ST NO FRR_{zero}	42.857	0.0	21.428	57.143	100.0	15.351
44	SM ST CL AER_{min}	44.603	0.0	22.302	55.397	100.0	2.256
45	2CH BN CL AER_{min}	47.884	0.0	23.942	52.116	100.0	3.858
46	SM ST NO EER	53.968	0.0	26.984	46.032	100.0	3.218
47	SM ST NO AER_{min}	58.73	0.0	29.365	41.27	100.0	3.458
48	2CH ST CL FRR_{zero}	58.942	0.0	29.471	41.058	100.0	9.176
49	2CH BD CL FRR_{zero}	59.048	0.0	29.524	40.952	100.0	6.313

50	2CH BN FB FRR_{zero}	59.206	0.0	29.603	40.794	100.0	14.766
51	2CH DO CL FRR_{zero}	59.735	0.0	29.868	40.265	100.0	8.865
52	2CH BD NO EER	70.053	0.0	35.027	29.947	100.0	4.269
53	2CH ST FB FRR_{zero}	70.212	0.0	35.106	29.788	100.0	16.788
54	2CH BN CL FRR_{zero}	72.328	0.0	36.164	27.672	100.0	4.859
55	2CH DO FB FRR_{zero}	73.651	0.0	36.825	26.349	100.0	18.641
56	SM ST CL FRR_{zero}	80.053	0.0	40.026	19.947	100.0	9.081
57	2CH BD NO AER_{min}	80.899	0.0	40.45	19.101	100.0	3.484
58	SM ST FB FRR_{zero}	85.45	0.0	42.725	14.55	100.0	8.384
59	2CH BN NO EER	87.566	0.0	43.783	12.434	100.0	1.659
60	SM ST NO FRR_{zero}	88.413	0.0	44.206	11.587	100.0	6.022
61	SM BN FB FRR_{zero}	90.317	0.0	45.159	9.683	100.0	7.065
62	2CH BN NO AER_{min}	91.111	0.0	45.556	8.889	100.0	1.518
63	2CH BD NO FRR_{zero}	92.381	0.0	46.19	7.619	100.0	2.623
64	SM BD NO FRR_{zero}	95.397	0.0	47.698	4.603	100.0	6.51
65	SM BN NO FRR_{zero}	97.249	0.0	48.624	2.751	100.0	3.891
66	2CH BN NO FRR_{zero}	98.466	0.0	49.233	1.534	100.0	0.536
67	SM DO CL FAR_{zero}	0.0	99.894	49.947	100.0	0.106	0.15
68	2CH ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
69	2CH BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
70	2CH DO NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
71	2CH BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
72	2CH ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
73	2CH BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
74	2CH DO CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
75	2CH BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
76	2CH ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
77	2CH BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
78	2CH DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
79	2CH BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
80	SM ST NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
81	SM BN NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
82	SM DO NO FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
83	SM BD NO FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
84	SM ST CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
85	SM BN CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
86	SM BN CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
87	SM DO CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0

88	SM BD CL FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
89	SM BD CL FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
90	SM ST FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
91	SM BN FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
92	SM DO FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
93	SM DO FB FRR_{zero}	88.889	11.111	50.0	11.111	88.889	0.0
94	SM BD FB FAR_{zero}	0.0	100.0	50.0	100.0	0.0	0.0
95	SM BD FB FRR_{zero}	100.0	0.0	50.0	0.0	100.0	0.0
96	SM DO NO FAR_{zero}	2.645	99.048	50.847	97.355	0.952	1.27

Appendix B

Execution times

B.1 Introduction

A summary of the execution times of 24 out of the total number of 96 systems proposed in this study within the context of the IDS and IIS is presented and discussed in this appendix. These 24 systems exclude the four different probabilistic threshold selection criteria (see Section 7.6.2), since they have no effect on execution time. The summary comprises of (1) training data loading time (TDLT), (2) validation data loading time (VDLT), (3) training time (TT), (4) validation time (VT) and (5) model response time (MRT). The preprocessing time is *included* in the MRT, while *all* training and validation images are already preprocessed within the context of the TT and the VT.

The computational resources employed for the purpose of determining the aforementioned execution times are as follows:

Description	**Specification**
CPU	2× Intel(R) Xeon(R) Gold 6150 CPU @ 2.70GHz (32 threads)
GPU	NVIDIA Tesla V100 PCIe 32 GB
RAM	157GB DDR4 2666MHz

Table B.1: Computational resources employed for the purpose of determining the execution times of the 24 proposed systems.

B.2 Individual dependent scenario

The execution times of the 24 systems within the context of the Bosphorus and Wilches databases and the individual dependent experiment (IDE) are presented in Tables B.2 and B.3 respectively. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations. It is important to note that the reported execution times are *averaged* over the three inner cross-validation folds associated with a *single* random individual and rounded to the nearest *second* for brevity, except within the context of the model response time (MRT), which is rounded to the third decimal.

Table B.2: The execution times of the 24 systems within the context of the Bosphorus database and the IDE. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations.

System Design	TDLT (s)	VDLT (s)	TT (s)	VT (s)	MRT (s)
2CH ST NO	16	15	288	19	4.292
2CH ST CL	16	15	288	19	4.466
2CH ST FB	16	15	288	19	5.213
2CH BN NO	16	15	289	19	4.466
2CH BN CL	16	15	289	19	4.471
2CH BN FB	16	15	289	19	5.213
2CH DO NO	16	15	288	19	4.292
2CH DO CL	16	15	288	19	4.466
2CH DO FB	16	15	288	19	5.213
2CH BD NO	16	15	289	19	4.292
2CH BD CL	16	15	289	19	4.466
2CH BD FB	16	15	289	19	5.213
SM ST NO	16	15	319	22	4.292
SM ST CL	16	15	319	22	4.466
SM ST FB	16	15	319	22	5.213
SM BN NO	16	15	321	22	4.292
SM BN CL	16	15	321	22	4.466
SM BN FB	16	15	321	22	5.213
SM DO NO	16	15	320	22	4.292
SM DO CL	16	15	320	22	4.466
SM DO FB	16	15	320	22	5.213
SM BD NO	16	15	322	22	4.292
SM BD CL	16	15	322	22	4.466
SM BD FB	16	15	322	22	5.213

Table B.3: The execution times of the 24 systems within the context of the Wilches database and the IDE. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations.

System Design	TDLT (s)	VDLT (s)	TT (s)	VT (s)	MRT (s)
2CH ST NO	16	15	288	19	0.005
2CH ST CL	16	15	288	19	0.023
2CH ST FB	16	15	288	19	0.042
2CH BN NO	16	15	289	19	0.005

2CH BN CL	16	15	289	19	0.023
2CH BN FB	16	15	289	19	0.042
2CH DO NO	16	15	288	19	0.005
2CH DO CL	16	15	288	19	0.023
2CH DO FB	16	15	288	19	0.042
2CH BD NO	16	15	289	19	0.005
2CH BD CL	16	15	289	19	0.023
2CH BD FB	16	15	289	19	0.042
SM ST NO	16	15	319	22	0.005
SM ST CL	16	15	319	22	0.023
SM ST FB	16	15	319	22	0.042
SM BN NO	16	15	321	22	0.005
SM BN CL	16	15	321	22	0.023
SM BN FB	16	15	321	22	0.042
SM DO NO	16	15	320	22	0.005
SM DO CL	16	15	320	22	0.023
SM DO FB	16	15	320	22	0.042
SM BD NO	16	15	322	22	0.005
SM BD CL	16	15	322	22	0.023
SM BD FB	16	15	322	22	0.042

The following conclusions may be reached from Tables B.2 and B.3:

1. The Siamese networks take significantly longer to train and validate due to their increased complexity when compared to the complexity of the 2CH networks.

2. The MRT is significantly worse within the context of the Bosphorus database, due to the increased complexity of the ROI extraction protocol. These response times may not be viable within the context of a real world scenario, and warrants investigation into the optimisation of the aforementioned ROI extraction protocol.

3. The average training and validation times for all the networks within the context of the IDS in which a tailor-made network is trained for each individual enrolled into the system is small enough so as to render this solution viable within the context of a real world scenario.

B.3 Individual independent scenario

The execution times of the 24 systems within the context of the Bosphorus database and the first GenVeins experiment (FGE) is presented in Table B.4. It is important to note that the execution times for *only* the FGE is presented and discussed within the context of the individual independent scenario (IIS). The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations. It is important to note that the reported execution times are averaged over the nine cross-validation folds and rounded to the nearest *second* for brevity, except within the context of the model response time (MRT), which is rounded to the third decimal.

Table B.4: The execution times of the 24 systems within the context of the Bosphorus database and the FGE. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations.

System Design	**TDLT (s)**	**VDLT (s)**	**TT (s)**	**VT (s)**	**MRT (s)**
2CH ST NO	157	124	1330	152	4.295
2CH ST CL	157	124	1330	152	4.471
2CH ST FB	157	124	1330	152	5.216
2CH BN NO	157	124	1387	162	4.295
2CH BN CL	157	124	1387	162	4.471
2CH BN FB	157	124	1387	162	5.216
2CH DO NO	157	124	1348	148	4.295
2CH DO CL	157	124	1348	148	4.471
2CH DO FB	157	124	1348	148	5.216
2CH BD NO	157	124	1443	157	4.295
2CH BD CL	157	124	1443	157	4.471
2CH BD FB	157	124	1443	157	5.216
SM ST NO	157	124	2427	314	4.295
SM ST CL	157	124	2427	314	4.471
SM ST FB	157	124	2427	314	5.216
SM BN NO	157	124	2629	324	4.295
SM BN CL	157	124	2629	324	4.471
SM BN FB	157	124	2629	324	5.216
SM DO NO	157	124	2523	282	4.295
SM DO CL	157	124	2523	282	4.471
SM DO FB	157	124	2523	282	5.216
SM BD NO	157	124	2695	328	4.295
SM BD CL	157	124	2695	328	4.471
SM BD FB	157	124	2695	328	5.216

Table B.5: The execution times of the 24 systems within the context of the Wilches database and the FGE. The order in which the system design is provided is as follows: neural network architecture, CNN-based FE and preprocessing protocol. Refer to Table A.1 for the descriptions of the abbreviations.

System Design	**TDLT (s)**	**VDLT (s)**	**TT (s)**	**VT (s)**	**MRT (s)**
2CH ST NO	157	124	1330	152	0.006
2CH ST CL	157	124	1330	152	0.026

2CH ST FB	157	124	1330	152	0.046
2CH BN NO	157	124	1387	162	0.006
2CH BN CL	157	124	1387	162	0.026
2CH BN FB	157	124	1387	162	0.046
2CH DO NO	157	124	1348	148	0.006
2CH DO CL	157	124	1348	148	0.026
2CH DO FB	157	124	1348	148	0.046
2CH BD NO	157	124	1443	157	0.006
2CH BD CL	157	124	1443	157	0.026
2CH BD FB	157	124	1443	157	0.046
SM ST NO	157	124	2427	314	0.006
SM ST CL	157	124	2427	314	0.026
SM ST FB	157	124	2427	314	0.046
SM BN NO	157	124	2629	324	0.006
SM BN CL	157	124	2629	324	0.026
SM BN FB	157	124	2629	324	0.046
SM DO NO	157	124	2523	282	0.006
SM DO CL	157	124	2523	282	0.026
SM DO FB	157	124	2523	282	0.046
SM BD NO	157	124	2695	328	0.006
SM BD CL	157	124	2695	328	0.026
SM BD FB	157	124	2695	328	0.046

The following conclusions may be reached from Tables B.4 and B.5:

1. The Siamese networks take significantly longer to train and validate due to their increased complexity when compared to the complexity of the 2CH networks, as is the case within the context of the IDE.

2. Batch normalisation layers are computationally more expensive than dropout layers, since they perform normalisation of all the mini-batches, whereas dropout layers simply omit certain neurons.

3. The MRT is significantly worse within the context of the Bosphorus database, as is the case within the context of the IDE.

4. The average training and validation times for all the networks within the context of the IIS in which a single network is trained in a once-off fashion prior to the enrolment of any clients is still relatively small when compared to multiple ANNs proposed in recent literature, which may take days to train.

When Tables B.2 and B.3 are compared with those depicted in Tables B.4 and B.5, it is clear that (1) the MRTs are slightly larger within the context of the IIS and (2) the networks take significantly longer to train. This is due to the fact that the number of network parameters were multiplied by 4 within the context of the IIS in order to account for the increased complexity of the optimisation function.

Bibliography

[1] Consumer sentinal network databook 2022. February 2022.
Available at: https://www.ftc.gov/reports/consumer-sentinel-network-data-book-2022

[2] Beukes, E.: *Hand vein-based biometric authentication with limited training samples.* Master's thesis, University of Stellenbosch, 2018.

[3] Lundberg, S.: Pytorch deep explainer mnist example. Available at:

[4] Black, J.: Simulated signatures: forgery by imitation. *Journal of forensic identification*, pp. 283–297, 1995.

[5] Upadhyay, S., Singh, J., Shukla, S. and Sareen, R.: A study of simulated forgery of signature. *Research J. Pharm. and Tech*, vol. 10, pp. 4215–4218, 2017.

[6] Mane, M.: Signs of traced signature in forgery. *Indian Journal of Applied Research*, vol. 4, pp. 213–215, 2014.

[7] Tarannum, A. and Singh, R.K.: Traced forgery - a case study. *Academic Journal of Forensic Sciences*, vol. 1, pp. 1–4, 2018.

[8] Watanabe, M. and Jain, A.: *Palm Vein*, pp. 1027–1033. Springer US, Boston, MA, 2009. ISBN 978-0-387-73003-5.
Available at: https://doi.org/10.1007/978-0-387-73003-5_129

[9] Yüksel, A., Akarun, L. and Sankur, B.: Biometric identification through hand vein patterns. In: *2010 International Workshop on Emerging Techniques and Challenges for Hand-Based Biometrics*, pp. 1–6. 2010.

[10] Grossi, E. and Buscema, M.: Introduction to artificial neural networks. *European journal of gastroenterology and hepatology*, vol. 19, pp. 1046–54, 01 2008.

[11] Bogazici University: Bosphorus hand vein database. 2010.
Available at: http://bosphorus.ee.boun.edu.tr/hand/Home.aspx

[12] Wilches-Bernal, F., Álvarez, B.N. and Vizcaya, P.: A database of dorsal hand vein images. *ArXiv*, vol. abs/1709.08761, 2020.
Available at: https://arxiv.org/abs/2012.05383

[13] Zuiderveld, K.: *VIII.5. - Contrast Limited Adaptive Histogram Equalization*, pp. 474–485. Academic Press, 1994. ISBN 978-0-12-336156-1.
Available at: https://www.sciencedirect.com/science/article/pii/B9780123361561500616

[14] Mukarambi, D.G.: Hybrid method for elimination of uneven illumination from camera-based document images. *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, 01 2020.

[15] O'Shea, K. and Nash, R.: An introduction to convolutional neural networks. 2015. `1511.08458`.

[16] Zagoruyko, S. and Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4353–4361. 2015.

[17] Ioffe, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv*, 2015.
Available at: `https://arxiv.org/abs/1502.03167`

[18] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.
Available at: `http://jmlr.org/papers/v15/srivastava14a.html`

[19] Garbin, C., Zhu, X. and Marques, O.: Dropout vs. batch normalization: an empirical study of their impact to deep learning. *Multimedia Tools and Applications*, vol. 79, pp. 12777–12815, 2020.

[20] Hu, J., Shen, L., Albanie, S., Sun, G. and Wu, E.: Squeeze-and-excitation networks. 2019. `1709.01507`.

[21] Gibson, P., Cano, J., Turner, J., Crowley, E.J., O Boyle, M. and Storkey, A.: Optimizing grouped convolutions on edge devices. In: *2020 IEEE 31st International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pp. 189–196. 2020.

[22] Thapar, D., Jaswal, G., Nigam, A. and Kanhangad, V.: Pvsnet: Palm vein authentication siamese network trained using triplet loss and adaptive hard mining by learning enforced domain specific features. In: *2019 IEEE 5th International Conference on Identity, Security, and Behavior Analysis (ISBA)*, pp. 1–8. January 2019.

[23] Zagoruyko, S. and Komodakis, N.: Deep compare: A study on using convolutional neural networks to compare image patches. *Computer Vision and Image Understanding*, vol. 164, pp. 38–55, 2017. ISSN 1077-3142. Deep Learning for Computer Vision.

[24] Beukes, E. and Coetzer, J.: Hand vein-based biometric authentication using two-channel similarity measure networks. In: *2020 International SAUPEC/RobMech/PRASA Conference*, pp. 1–6. 2020.

[25] Beukes, E. and Coetzer, J.: Hand vein-based biometric authentication with convolutional neural networks and support vector machines. *International Journal of Recent Research in Electrical and Electronics Engineering*, vol. 9, no. 3, pp. 12–31, August 2022.
Available at: `https://doi.org/10.5281/zenodo.6961864`

[26] Kumar, A. and Prathyusha, K.V.: Personal authentication using hand vein triangulation and knuckle shape. *IEEE Transactions on Image Processing*, vol. 18, no. 9, pp. 2127–2136, 2009.

[27] Narula, P., Srivastava, S., Hanmandlu, M. and Bhargava, S.: Robust authentication using dorsal hand vein images. *Intelligent Systems, IEEE*, vol. 34, pp. 25–35, November 2018.

[28] Krishnaveni, N.V., Sivasankari, K. and Vijayan, V.: Personal authentication using hand vein. *International Journal of Engineering Research and Technology (IJERT)*, vol. 03, no. 01, pp. 2333–2339, January 2014.

[29] Trabelsi, R., Alima, D. and Sellami, D.: A new multimodal biometric system based on finger vein and hand vein recognition. *International Journal of Engineering and Technology*, vol. 5, pp. 3175–3183, August 2013.

[30] Gourisankar, S. and Sankareswaran, U.M.: Personal authentication system using hand vein biometric. *International Journal of Computer Technology and Applications*, vol. 03, pp. 383–391, January 2012.

[31] Al-Roomi, S.A., Alshayeji, M.H. and Abed, S.: Efficient hand vein recognition using local keypoint descriptors and directional gradients. *Multimedia Tools and Applications*, vol. 81, no. 11, pp. 15687–15705, 2022.
Available at: `https://doi.org/10.1007/s11042-022-12608-6`

[32] Abdullah, N. and Elrefaei, L.: Dorsal hand vein recognition by convolutional neural networks: Feature learning and transfer learning approaches. *International Journal of Intelligent Engineering and Systems*, vol. 12, pp. 178–171, June 2019.

[33] Laghari, W.A., Gaik, T.K., Huong, A.K.C., Choy, Y.Y. and Chew, C.C.: Dorsal hand vein identification using transfer learning from alexnet. *International Journal of Integrated Engineering*, vol. 14, no. 3, pp. 111–119, June 2022.

[34] Chantaf, S., Hilal, A. and Elsaleh, R.: Palm vein biometric authentication using convolutional neural networks. In: Bouhlel, M.S. and Rovetta, S. (eds.), *Proceedings of the 8th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT'18), Vol.1*, pp. 352–363. Springer International Publishing, Cham, 2020.

[35] Obayya, M., El-Ghandour, M. and Alrowais, F.: Contactless palm vein authentication using deep learning with bayesian optimization. *IEEE Access*, vol. PP, pp. 1940–1957, December 2020.

[36] Qin, H., El Yacoubi, M., Lin, J. and Liu, B.: An iterative deep neural network for hand-vein verification. *IEEE Access*, vol. 7, pp. 34823–34837, February 2019.

[37] Shu, Z., Xie, Z. and Zhang, C.: Dorsal hand vein recognition based on transmission-type near infrared imaging and deep residual network with attention mechanism. *Optical Review*, vol. 29, no. 4, pp. 335–342, 2022. ISSN 1349-9432.
Available at: `https://doi.org/10.1007/s10043-022-00750-3`

[38] Bell, S. and Bala, K.: Learning visual similarity for product design with convolutional neural networks. *ACM Transactions on Graphics*, vol. 34, no. 4, pp. 1–10, July 2015. ISSN 0730-0301.
Available at: `https://doi.org/10.1145/2766959`

[39] Appalaraju, S. and Chaoji, V.: Image similarity using deep cnn and curriculum learning. *ArXiv*, vol. abs/1709.08761, 2017.
Available at: `https://arxiv.org/abs/1709.08761`

[40] Zhang, Y.: Learning near duplicate image pairs using convolutional neural networks. *International Journal of Performability Engineering*, vol. 14, pp. 168–177, January 2018.

[41] Fang, Y., Wu, Q. and Kang, W.: A novel finger vein verification system based on two-stream convolutional network learning. *Neurocomputing*, vol. 290, pp. 100–107, 2018. ISSN 0925-2312.
Available at: `https://www.sciencedirect.com/science/article/pii/S0925231218301772`

[42] Bagchi, S., Chanda, G., Agarwal, A. and Ratha, N.: On deep learning for dorsal hand vein recognition. In: *2022 IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, pp. 1–4. 2022.

[43] Swain, A., Marattukalam, F. and Abdulla, W.H.: N-shot palm vein verification using siamese networks. *CoRR*, vol. abs/2109.12808, 2021. `2109.12808`.
Available at: `https://arxiv.org/abs/2109.12808`

[44] Gao, X., Qin, H., Gong, C., Li, Y. and El-Yacoubi, M.A.: Label enhancement-based multi-scale transformer for palm-vein recognition. *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–17, 2023.

[45] El-Yacoubi, M.A., Qin, H., Hu, R., Li, Y. and Gao, X.: Local attention transformer-based full-view finger-vein identification. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 6, pp. 2767–2782, 2023.

[46] Garcia-Martin, R. and Sanchez-Reillo, R.: Vision transformers for vein biometric recognition. *IEEE Access*, vol. 11, pp. 22060–22080, 2023.

[47] Li, Y., Lu, H., Wang, Y., Gao, R. and Zhao, C.: Vit-cap: A novel vision transformer-based capsule network model for finger vein recognition. *Applied Sciences*, vol. 12, no. 20, 2022. ISSN 2076-3417.
Available at: `https://www.mdpi.com/2076-3417/12/20/10364`

[48] Otsu, N.: A threshold selection method from gray-level histograms. *IEEE Trans. Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[49] Gupta, S. and Kaur, Y.: Review of different local and global contrast enhancement techniques for a digital image. *International Journal of Computer Applications*, vol. 100, pp. 18–23, August 2014.

[50] Gonzales, R.C. and Woods, R.E.: *Digital Image Processing*. Pearson, 2010.

[51] McCulloch, W.S. and Pitts, W.: A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, December 1943. ISSN 1522-9602.
Available at: `https://doi.org/10.1007/BF02478259`

[52] Morris, R.: D.o. hebb: The organization of behavior, wiley: New york; 1949. *Brain Research Bulletin*, vol. 50, no. 5-6, p. 437, 1999. ISSN 0361-9230.
Available at: `https://doi.org/10.1016/s0361-9230(99)00182-3`

[53] McCulloch, W.S., Oettinger, A.G., Schmitt, O.H. and Rochester., N.: Symposium the design of machines to simulate the behavior of the human brain. *IRE Transactions on Electronic Computers*, vol. EC-5, no. 4, pp. 240–255, 1956.

[54] Rochester, N., McCarthy, J., Minsky, M.L. and Shannon, C.E.: A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI Magazine*, vol. 27, no. 4, p. 12, December 2006.
Available at: `https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/1904`

[55] Widrow, B. and Hoff, M.E.: *Associative Storage and Retrieval of Digital Information in Networks of Adaptive "Neurons"*, pp. 160–160. Springer US, Boston, MA, 1962. ISBN 978-1-4684-1716-6.
Available at: `https://doi.org/10.1007/978-1-4684-1716-6_25`

[56] Minsky, M. and Papert, S.: *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge, MA, USA, 1969.

[57] Werbos, P.J.: *Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Sciences*. Ph.D. thesis, Harvard University, 1975.

[58] Hopfield, J.: Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, pp. 2554–2558, May 1982.

[59] Reilly, D.L., Cooper, L.N. and Elbaum, C.: A neural model for category learning. *Biological Cybernetics*, vol. 45, no. 1, pp. 35–41, August 1982. ISSN 1432-0770.
Available at: `https://doi.org/10.1007/BF00387211`

[60] Rumelhart, D.E., Hinton, G.E. and Williams, R.J.: Learning representations by back-propagating errors. *Nature*, vol. 323, no. 6088, pp. 533–536, October 1986.
Available at: `https://doi.org/10.1038/323533a0`

[61] Marhon, S.A., Cameron, C.J.F. and Kremer, S.C.: *Recurrent Neural Networks*, pp. 29–65. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-36657-4.
Available at: `https://doi.org/10.1007/978-3-642-36657-4_2`

[62] Goodfellow, I., Bengio, Y. and Courville, A.: *Deep Learning*. MIT Press, 2016. Book in preparation for MIT Press.
Available at: `http://www.deeplearningbook.org`

[63] Harmon, M. and Klabjan, D.: Activation ensembles for deep neural networks. In: *2019 IEEE International Conference on Big Data (Big Data)*, pp. 206–214. IEEE Computer Society, Los Alamitos, CA, USA, December 2019.

[64] Glorot, X. and Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: *International Conference on Artificial Intelligence and Statistics*, vol. 9, pp. 249–256. 01 2010.

[65] He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778. June 2016.

[66] Krizhevsky, A., Sutskever, I. and Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C., Bottou, L. and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, vol. 25. Curran Associates, Inc., 2012.

[67] Beleites, C., Baumgartner, R., Bowman, C., Somorjai, R., Steiner, G., Salzer, R. and Sowa, M.G.: Variance reduction in estimating classification error using sparse datasets. *Chemometrics and Intelligent Laboratory Systems*, vol. 79, no. 1, pp. 91–100, 2005. ISSN 0169-7439.
Available at: `https://www.sciencedirect.com/science/article/pii/S0169743905000687`

[68] Liew, V.: An overview on various ways of bootstrap methods. *MPRA Paper*, 01 2008.

[69] Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

[70] Ross, A. and Willson, V.L.: *Independent Samples T-Test*, pp. 13–16. SensePublishers, Rotterdam, 2017. ISBN 978-94-6351-086-8.
Available at: `https://doi.org/10.1007/978-94-6351-086-8`$_3$

[71] Shorten, C. and Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019. ISSN 2196-1115.
Available at: `https://doi.org/10.1186/s40537-019-0197-0`

[72] Polyak, B.T. and Juditsky, A.B.: Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.

[73] von Stackelberg, P., Goedhart, R., Birbil, .l. and Does, R.J.M.M.: Comparison of threshold tuning methods for predictive monitoring. *Quality and Reliability Engineering International*, vol. 40, no. 1, pp. 499–512, 2024. `https://onlinelibrary.wiley.com/doi/pdf/10.1002/qre.3436`.
Available at: `https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.3436`

[74] Lundberg, S. and Lee, S.-I.: A unified approach to interpreting model predictions. *arXiv*, 2017. `1705.07874`.
Available at: `https://arxiv.org/abs/1705.07874`

[75] Shapley, L. and Corporation, R.: *Notes on the N-person Game*. No. v. 2 in Notes on the N-person Game. Rand Corporation, 1951.
Available at: `https://books.google.co.za/books?id=uMoWzwEACAAJ`

[76] Shrikumar, A., Greenside, P. and Kundaje, A.: Learning important features through propagating activation differences. *arXiv*, 2019. `1704.02685`.
Available at: `https://arxiv.org/abs/1704.02685`

[77] Wang, C., Peng, M., Xu, L. and Chen, T.: A single scale retinex based method for palm vein extraction. In: *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, pp. 75–78. 2016.

[78] Wang, J., Yuan, Y. and Li, G.: Multifeature contrast enhancement algorithm for digital media images based on the diffusion equation. *Advances in Mathematical Physics*, vol. 2022, pp. 1–11, 01 2022.

[79] Lefkovits, S., Emerich, S. and Lefkovits, L.: Boosting unsupervised dorsal hand vein segmentation with u-net variants. *Mathematics*, vol. 10, no. 15, 2022. ISSN 2227-7390. Available at: `https://www.mdpi.com/2227-7390/10/15/2620`

[80] Hinton, G.E., Frosst, N. and Sabour, S.: Dynamic routing between capsules. *CoRR*, vol. abs/1710.09829, 2017. `1710.09829`. Available at: `http://arxiv.org/abs/1710.09829`

[81] Niu, X.-X. and Suen, C.Y.: A novel hybrid cnnâsvm classifier for recognizing handwritten digits. *Pattern Recognition*, vol. 45, no. 4, pp. 1318–1325, 2012. ISSN 0031-3203. Available at: `https://www.sciencedirect.com/science/article/pii/S0031320311004006`

[82] Kingma, D.P. and Ba, J.: Adam: A method for stochastic optimization. *ArXiv*, 2017. `1412.6980`.