UNIVERSITÀ DEGLI STUDI DI UDINE
POLYTECHNIC DEPARTMENT OF ENGINEERING AND ARCHITECTURE

DOCTORAL PROGRAMME IN
INDUSTRIAL AND INFORMATION ENGINEERING

DOCTORAL THESIS

A dynamical system approach for pattern recognition and image analysis in biometrics and phytopathology

SUPERVISOR:
Prof. Pier Luca Montessoro

CANDIDATE:
David Palma

CO-SUPERVISOR:
Prof. Franco Blanchini

DOCTORATE COORDINATOR:
Prof. David Esseni

Doctoral cycle XXXIII

March 31, 2021

Author: David Palma

Author's web page: `https://david-palma.github.io`

Author's e-mail: palma.david (at) spes.uniud.it

University of Udine

Polytechnic Department of Engineering and Architecture

Via delle Scienze, 206

33100 Udine

Italy

To my son, Leonardo

Abstract

The research presented in this thesis concerns the efficient application of the positive dynamical systems theory to problems arising in pattern recognition and image analysis, specifically, in the biometric security and plant pathology areas, providing both theoretical and experimental results. Thus, a novel approach to this kind of problems has been investigated. With this in mind, the principle contributions of this thesis can be summarised within the context of the above overlapping lines of research. In the first part, an introduction to the field of biometrics is given in order to present the concepts and primitives of performance metrics due to their impact on secure biometric systems. Thus, has been presented an overview to describe the main biometric traits along with their properties as well as the various biometric system operating modalities. Finally, the criteria for performance evaluation have been defined to determine the system accuracy and security which are related to the applicability in real-world deployments. Secondly, it has been investigated the feasibility of the proposed approach in biometrics. This study has led to the definition of a unified method for line-like feature matching that relies on a recursive algorithm based on a monotone dynamical system whose output converges either to zero, when a deep mismatch exists between the samples to be compared, or to a high value, when a good matching is observed, thus allowing the system to be employed in several applications, including all possible vascular-based biometric security systems based on blood vessel pattern matching. Thirdly, to consolidate the theoretical results, two examples of biometric security systems have been developed. In particular, it has been considered the case of hand palm-based human recognition first using the samples acquired in the visible spectrum and then those acquired in the near-infrared spectrum. Indeed, in highly security sensitive applications, vascular biometrics is more often used, since the involved patterns are considered to be very stable over time, difficult to forge, and false match rates are the lowest of all biometric recognition methods. In this context, it has been applied the proposed unified method, successfully testing carefully designed experiments to assess its performance, even in adverse conditions (e.g., in presence of noise). In the second part, at first is given an introduction to the field of phytopathology oriented to image-based diagnosis of plant disease symptoms in order to present the concepts and primitives of performance metrics due to their impact on such systems. Thus, an overview is provided to describe the main grapevine pathogenic diseases along with their properties as well as the analysis of visual symptoms used for the assessment of disease severity. Finally, the criteria for performance evaluation have been defined to determine the system accuracy and reliability which are related to the applicability in real-world deployments. After that, it has been investigated the feasibility of the proposed approach in plant pathology. In the vineyard, symptoms are visually most noticeable on leaves, which may include a detectable change in color, shape, or function of the plant as it responds to the pathogen. Hence, to detect potential plant pathogens as quickly as possible in order to reduce the likelihood of an infection spreading, it has been proposed a unified method based on the positive dynamical systems theory

that allows the detection and severity estimation of grape diseases regardless of disease type. The idea behind the algorithm is to recursively spread the disease to fill the infected regions of the leaf only if there are symptoms of the condition itself, otherwise the leaf will not be affected by any changes. Lastly, to consolidate the theoretical results, an example of grape leaf disease detection and severity estimation has been developed. In particular, it has been considered the case of a specific disease-causing agent due to biotic factors (i.e., those caused by living components such as pathogens). In this context, it has been applied the proposed unified method and the experiments have assessed the system ability to generalise symptoms beyond any previously seen conditions, also achieving promising results, even in adverse conditions (e.g., in presence of noise). In both the proposed unified methods, the main advantage rely in the robustness when dealing with low-resolution and noisy images. Indeed, an essential issue related to digital image processing is to effectively reduce noise from an image whilst keeping its features intact. The impact of noise (e.g., signal independent and uncorrelated noise) is effectively reduced and does not affect the final result allowing the proposed systems to ensure a high accuracy and reliability.

Acknowledgements

This thesis is the result of a very long journey, an accomplishment that would not have had the same value without the help of some people who deserve my gratitude.

First and foremost, I would like to express my deepest and most sincere gratitude to my research supervisors Pier Luca Montessoro and Franco Blanchini, for their invaluable guidance, unwavering support, and belief in me. Their immense knowledge and plentiful experience have encouraged me in all the time of my academic research and daily life. I would like to thank you very much for your support and understanding over the past years.

I am also grateful to all the people from several institutions and companies with whom I have had the opportunity to interact and collaborate. I would like to thank all my co-authors for the fruitful research discussions that gave us the opportunity to combine our topics and work on papers together.

I would also like to express my appreciation to Thomas Parisini, who hosted me at the Electrical and Electronic Engineering Department of the Imperial College London, and all the members of the Control and Power (CAP) Research Group with whom I had stimulating discussions.

I also have to thank the doctoral board and in particular David Esseni, the coordinator of the IIE doctoral programme at the University of Udine, for organising a wide range of events and training activities. Furthermore, I express my thanks to the reviewers of my doctoral thesis, for the insightful comments and suggestions that really helped me improve this work.

Getting through my dissertation required more than academic support, and I have many, many people to thank for listening to and, at times, having to tolerate me over the past three and a half years. From my closest friends to my teachers and boxing coaches (from both the Boxing Club in Udine and the Moreno Boxing Club in London), you know well who you are. I would like to say a very warm thank you to all of you for your thoughts and your ways to brighten my daily routine.

Finally, I would like to express my gratitude to my parents and my sister's family for all their love and support. Most importantly, I am exceptionally grateful to my one true love, Jessica, and my loving son, Leonardo, who are a source of joy and inspiration. I am forever indebted to you for teaching me to chase my dreams.

David Palma
Udine, March 31, 2021

Contents

List of figures

List of tables

Nomenclature

Common sets

$\mathbb{A}$	A set
$\mathbb{N}$	The set of natural numbers, including 0
$\mathbb{Z}$	The set of integer numbers
$\{0, 1\}$	The set containing 0 and 1
$\{0, 1, \dots, n\}$	The set of all integers between 0 and n
$\mathbb{R}$	The set of real numbers
$\mathbb{R}_+$ $(\mathbb{R}_{++})$	The set of non-negative (positive) real numbers
$\mathbb{R}^{n\times m}$	The set of real $n \times m$ matrices
$\mathbb{S}^n$	The set of real symmetric $n \times n$ matrices
$\mathbb{S}^n_+$ $\left(\mathbb{S}^n_{++}\right)$	The set of non-negative (positive) symmetric $n \times n$ matrices
$\mathbb{M}$	The set of Metzler matrices

Numbers, vectors, and matrices

a	A scalar
$\mathbf{a}$	A vector
$\mathbf{A}$	A matrix
$\mathcal{A}$	A tensor
$\mathbf{I}_n$	Identity matrix with n rows and n coloumns
$\mathbf{I}$	Identity matrix with dimensionality implied by context
$\mathbf{O}$	Zero matrix with dimensionality implied by context
$\operatorname{diag}(\mathbf{a})$	A square, diagonal matrix with diagonal entries given by $\mathbf{a}$
$\operatorname{diag}(\mathbf{A})$	A vector, with elements given by diagonal matrix $\mathbf{A}$

Indexing

a_i	Element i of vector $\mathbf{a}$, with indexing starting at 1
a_{-i}	All elements of vector $\mathbf{a}$ except for element i
$A_{i,j}$	Element (i, j) of matrix $\mathbf{A}$
$\mathbf{A}_{i,:}$	Row i of matrix $\mathbf{A}$
$\mathbf{A}_{:,j}$	Coloumn j of matrix $\mathbf{A}$
$A_{i,j,k}$	Element (i, j, k) of a 3-dimensional tensor $\mathcal{A}$
$\mathbf{A}_{:,:,k}$	2-dimensional slice of a 3-dimensional tensor $\mathcal{A}$

Linear algebra operations

$\boldsymbol{A}^\top$	Transpose of matrix $\boldsymbol{A}$
$\det(\boldsymbol{A})$	Determinant of $\boldsymbol{A}$
$\operatorname{tr}(\boldsymbol{A})$	Trace of the matrix $\boldsymbol{A}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Element-wise (Hadamard) product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \otimes \boldsymbol{B}$	Kronecker product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\lvert\boldsymbol{x}\rvert$	Absolute value (applies element-wise) of $\boldsymbol{x}$
$\lVert\boldsymbol{x}\rVert$	L^2 norm of $\boldsymbol{x}$
$\lVert\boldsymbol{x}\rVert_p$	L^p norm of $\boldsymbol{x}$

Probability and information theory

$a \sim P$	Random variable a has distribution P
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2

Calculus

$\frac{d\psi}{dx}$	Derivative of function ψ with respect to x
$\psi'(x)$	Derivative of function ψ with respect to x
$\frac{\partial\psi}{\partial x}$	Partial derivative of function ψ with respect to x
$\nabla\psi$	Gradient of function ψ (vector of first partial derivatives)
$\nabla^2\psi$	Laplacian of function ψ (sum of all the unmixed second partial derivatives)
$\boldsymbol{H}(\psi)$	Hessian matrix of function ψ (matrix of second partial derivatives)

Logical connectives

$\neg\boldsymbol{A}$	Logical complement, also known as "not operator"
$\boldsymbol{A} \wedge \boldsymbol{B}$	Truth-functional operator of logical conjunction, also known as "and operator"
$\boldsymbol{A} \vee \boldsymbol{B}$	Truth-functional operator of logical disjunction, also known as "or operator"
$\boldsymbol{A} \nleftrightarrow \boldsymbol{B}$	Truth-functional operator of logical exclusive disjunction, also known as "xor operator"
$\boldsymbol{A} \leftrightarrow \boldsymbol{B}$	Truth-functional operator of logical biconditional, also known as "xnor operator"

Acronyms

Some of the acronyms are as follows.

AWGN	additive white Gaussian noise	**BDOC**	block dominant orientation code

BHOG	block-based histogram of oriented gradient
CCD	charge-coupled device
CNN	convolutional neural network
DBNN	decision-based neural network
DET	detection error trade-off
DFT	discrete Fourier transform
DNA	deoxyribonucleic acid
DOC	double-orientation code
DoG	difference of Gaussians
dpi	dots per inch
DPVM	dynamic palm vein matching
DS	disease severity
ECG	electrocardiogram
EER	equal error rate
FAR	false acceptance rate
FBI	Federal Bureau of Investigation
FDA	Fisher discriminated analysis
FIR	finite impulse response
FN	false negative
FP	false positive
FRR	false rejection rate
GAR	genuine acceptance rate
HMM	hidden Markov model
ICA	independent component analysis
IEC	international electrotechnical commission
ISO	international organization for standardization
KDDA	kernel direct discriminant analysis
LBP	local binary pattern
LCE	linear contrast enhancement
LDA	linear discriminant analysis
LoG	Laplacian of Gaussian
LTI	linear time-invariant
MCC	Matthews correlation coefficient
MFRAT	modified finite Radon transform
NGI	Next Generation Identification
NIR	near-infrared
NMRT	neighbourhood matching Radon transform
NN	nearest neighbour
PA	precision agriculture
PCA	principal component analysis
PCF	phase-correlation function
PDF	probability density function
PIN	personal identification number
PNN	probabilistic neural network
PPV	positive predictive value
PSD	power spectral density
PSO	particle swarm optimisation
RMS	root mean square
RNA	ribonucleic acid
ROC	receiver operating characteristic
ROI	region of interest

RootSIFT	root of the scale invariant feature transform
SNR	signal-to-noise ratio
SRKDA	spectral regression kernel discriminant analysis
STR	short tandem repeat
SURF	speeded up robust features
SVM	support vector machine
TN	true negative
TP	true positive
TPR	true positive rate
UAV	unmanned aerial vehicle
WAT	wave atom transform
WS	wavelet scattering

1

Introduction

> The theory of positive systems is deep and elegant – and yet pleasantly consistent with intuition. [...] It is for positive systems, therefore, that dynamic systems theory assumes one of its most potent forms.
>
> David Luenberger, in Introduction to Dynamic Systems, Wiley, 1979 [95].

Positive dynamical systems are an important class of systems that arise naturally in many fields of science where the state-variables represent quantities that can only be positive (or at least non-negative) in value at all times. The explicit definition of a positive system is that its state and output are always non-negative for any non-negative initial state and any non-negative input. This non-negative restriction on system variables provides some remarkable outcomes that are available only for positive dynamical systems [13, 43, 133].

1.1 Thesis overview and contributions

Motivated by the importance of positivity in systems theory, this thesis presents a study based on a dynamical system approach to address challenging pattern recognition and image analysis tasks from both a theorethical and practical perspective, along with an interesting application of this kind of systems to the biometrics and phytopathology areas[1]. The purpose of this chapter is to provide an overview of the aforementioned fields of study in such a way so as to contextualise the contributions of this thesis, as well as equip the reader with background information sufficient to understand the technical developments of the succeeding chapters.

The central theme of this thesis is the efficient application of the positive dynamical systems theory to problems arising in pattern recognition and image analysis, specifically, in the biometric security and plant pathology areas. With this in mind, the principle contributions of this thesis can be summarised within the context of the above overlapping lines of research. Reflecting this twofold objective, the thesis is divided into two parts.

[1] Most of the works described in this thesis have been published in peer-reviewed conference or journal, or are currently under review or in preparation, however, the corresponding publications are always referenced; for the sake of completeness, almost all of the methods have been either reported or sketched in the thesis as well.

1. Biometrics

Biometric technologies refer to all processes used to recognise, authenticate, and identify individuals based on their biological and/or behavioural characteristics. Whether for security, access, or fraud prevention, biometrics come in many forms, and the number of deployed biometrics solutions is evolving quickly, as well. The global biometrics market is growing at a fast pace, initially due to the need to combat the rising security challenges. Nowadays, with facial recognition and fingerprint applications dripping down to consumer market through smartphones, broader applications of biometrics will be highly expected for the future digital world. Trends indicate that existing applications will expand and new ones will emerge, meaning that biometrics will become increasingly ubiquitous and powerful [56]. Accurate authentication or identification is fundamental to physical security, cyber security, financial transactions, contracts and employment, public services, criminal justice, national security and more. The immense interest in the theory, technology, applications, and social implications of biometric systems has created an imperative need for the systematic study of the use of biometrics in security and surveillance infrastructures [18]. The approaches that have been proposed in literature depend on the type and the number of the underlying biometric traits, which, in general, cannot be easily transferred between people, and thereby represent a highly secure unique identifier. Physical traits include the anatomical components and physiological functioning of the human body, whilst behavioural traits describe the way an individual reacts or moves within the environment. Biometric systems are pattern recognition systems that automate the verification and identification processes by capturing characteristics of individuals, extracting measurable features, and comparing the data against the stored template of biometric datasets from a prior enrollment procedure. Biometrics data is captured as digital images, known as *samples* and converted to *templates* that can be mathematically compared to each other by means of various biometric algorithms. In particular, in highly security sensitive applications, vascular biometrics is more often used, since the involved patterns are considered to be very stable over time, difficult to forge, and false match rates are the lowest of all biometric identification methods. In this context, a novel unified model for robust line-like feature matching, which can be employed in all vascular-based biometric security applications, has been perfected testing carefully studied experiments to assess its performance even in adverse conditions (e.g., in presence of noise). This model relies on a recursive algorithm based on a monotone dynamical system whose evolution depends on the two matrices representing the templates to be compared. The main advantage of the proposed approach is its robustness when dealing with low-resolution and noisy images. The impact of noise (e.g., impulse noise) is effectively reduced. Moreover, the proposed method can admit several extensions and applications (e.g., vascular-based biometric security applications, line-like feature matching on aerial imagery, and so on). The unified method described in Chapter 4 has been proposed in two versions, the first of which has been applied to palm line features acquired in the visible electromagnetic spectrum, whilst the enhanced version has been applied to subcutaneous palm vascular patterns in the near-infrared spectrum, both operating in contact-less mode, ensuring high user acceptability as well as hygienic conditions without discomfort, also achieving state-of-the-art performance [15].

2. **Phytopathology**

The demand for high level of safety and superior quality in agricultural products is of prime concern. The introduction of new technologies for supporting crop management allows the efficiency and quality of production to be improved and, at the same time, reduces the environmental impact. Common strategies to disease control are mainly oriented on spraying pesticides uniformly over cropping areas at different times during the growth cycle. Even though these methodologies can be effective, they present a negative impact in ecological and economic terms, introducing new pests and elevating resistance of the pathogens [106]. Recent technological developments have allowed useful tools to automatically detect the symptoms that appear on the leaves and stem of a plant, thus helping in the cultivation of healthy plants and improving their quality [104]. Severity estimation of plant disease is an important procedure to measure the degree of disease and thus can be used to recommend treatment and predict yield, helping to reduce crop losses [14]. The studies of plant disease diagnosis refer to studying the visually observable patterns (symptoms) of a specific plant. Since the first step to detect when a plant has any type of deficiency, pest, or pathogenic disease is to observe its stems, its grapes, and/or its leaves, the idea is that, once the digital images are automatically acquired from the environment, e.g., by means of a unmanned aerial vehicle (UAV), both image processing and soft computing techniques are applied following a pattern recognition system scheme. Many of these image-based assessment methods for plant diseases, indeed, rely on the same basic procedure [9, 10]. Firstly, preprocessing techniques are employed for background removal and segmentation of the lesion tissue of the infected plants. Then, discriminative features are extracted for further analysis. At last, supervised classification algorithms or unsupervised cluster algorithms are used to classify features according to the specific task [141]. Studies using visible features imaged with conventional RGB cameras have shown the ability for autometd systems to recognise the presence of known plant disease using deep convolutional neural network models [110]. Deep learning models rely on large neural networks that typically require a very time-consuming training, moreover to accurately generalise predictions (classification among different diseases) is needed a very large number of verified images of diseased and healthy plants. However, the fine-grained disease severity estimation is much more challenging, as there exist large intraclass similarity and small interclass variance [152]. In this context, to detect potential plant pathogens as quickly as possible in order to reduce the likelihood of an infection spreading, a different and highly efficient unified method relying on a recursive algorithm based on a monotone dynamical system has been presented, allowing the automatic detection and severity estimation of pathogenic diseases regardless of disease type (i.e., not suitable for disease classfication). The unified method described in Chapter 6 does not require any training to automatically discover the discriminative features for fine-grained estimation, which enables the end-to-end pipeline for diagnosing plant disease severity. In Section 6.3, a fully automatic plant disease detection and severity estimation system has been presented and the proposed experimental setup has allowed to assess the system ability to generalise symptoms beyond any previously seen conditions, also achieving excellent results, even in adverse conditions (e.g., in presence of noise).

1.2 Structure of the thesis

The thesis is articulated as illustrated in Figure 1.1 and described below.

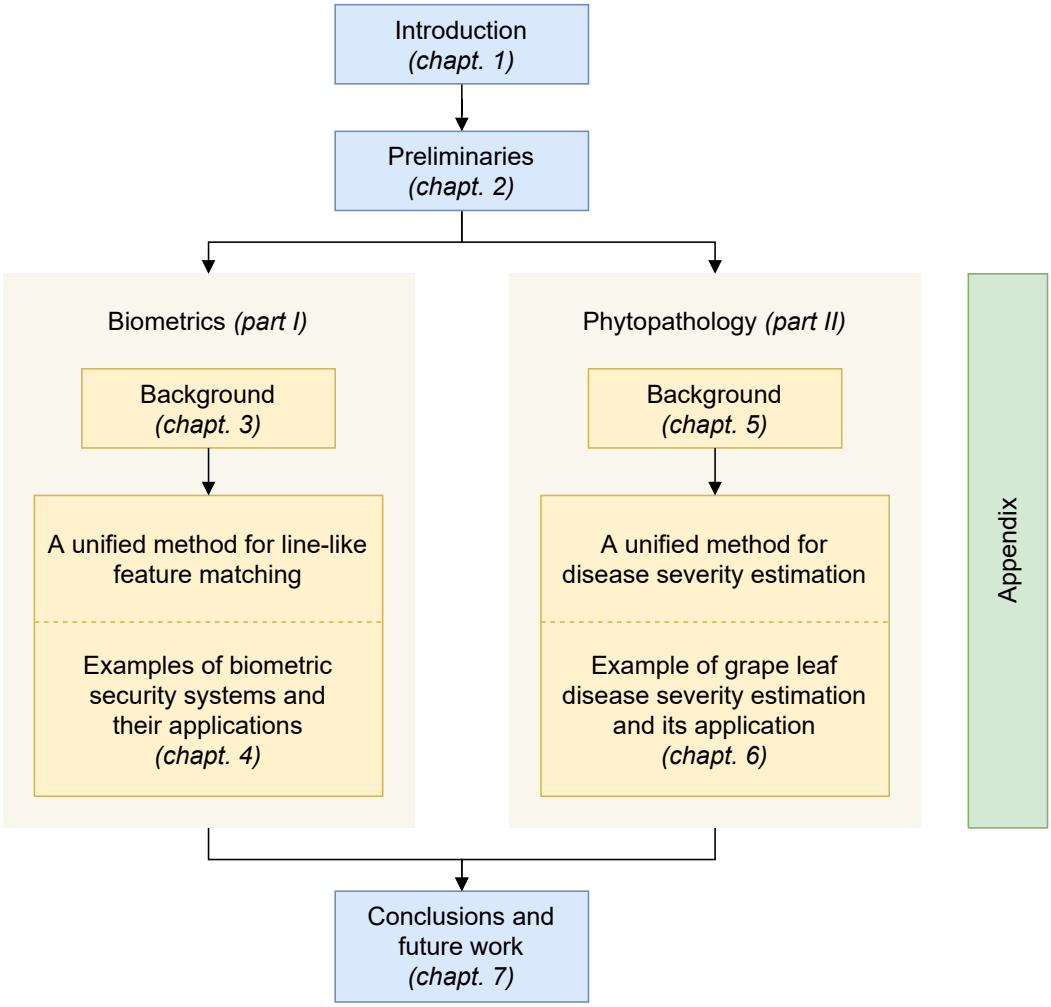


Figure 1.1: Structure of the thesis.

- **Chapter 1** provides an overview of the study in order to contextualise the contributions of this thesis, as well as equip the reader with background information sufficient to understand the technical developments of the succeeding chapters.
- **Chapter 2** intruduces the terminology, the notation, and presents a brief introduction to positive systems and monotonicity along with some basic properties of positive matrices and Metzler matrices.
- **Chapter 3** introduces the fundamental concepts, theory, design, and application of biometric characterisation of human beings, including the main biometric traits along with their properties and the various biometric system operating modalities, as well as the primitives of performance metrics to determine the system accuracy and security which are related to the applicability in real-world deployments.

- **Chapter 4** presents a noise-rejecting unified method for line-like feature matching, which relies on a recursive algorithm based on a positive linear dynamical system whose evolution is determined by the matching level between the two input images: its output converges to zero when the two images have a deep mismatch, while it reaches a high value in the case of good matching. The last part of this chapter illustrates two examples of biometric recognition systems where the first one is based on principal palm lines features acquired in the visible spectrum, whilst the second one is based on subcutaneous palm vascular patterns acquired in the near-infrared spectrum, proving the effectiveness and robustness of the proposed dynamical system by means of a massive campaign of experiments.

- **Chapter 5** introduces the fundamental concepts in the field of phytopathology oriented to image-based diagnosis of plant disease symptoms, including the main grapevine pathogenic diseases along with their properties and the analysis of visual symptoms used for the assessment of disease severity, as well as the primitives of performance metrics to determine the system accuracy and reliability which are related to the applicability in real-world deployments.

- **Chapter 6** presents a noise-rejecting unified method for automatic disease severity estimation, which relies on a recursive algorithm based on a positive non-linear dynamical system whose evolution is determined by the input image to be analysed: the system recursively spread the disease to fill the infected regions only if there are symptoms of the condition itself. The last part of this chapter illustrates an example of disease severity estimation system to automatically detect the symptoms that appear in the input image. To assess the performance and the effectiveness of the proposed, extensive experiments have been carried out, even in noisy conditions.

- **Chapter 7** concludes the thesis by discussing the main achievements of the presented work and suggests ideas for future work.

Appendix

- **A** provides essential mathemathical notions, techniques, and transformations in digital image processing and image analysis.

- **B** summarises the transformations between the CIE RGB, CIE XYZ, and CIE L*a*b* colour spaces.

- **C** provides analysis of the main experimental results for the parameter optimisation procedure relative to the system described in Chapter 4.

2

Preliminaries

In this chapter, we introduce terminology, positive systems, various other definitions, which will be essentially used for proving our main results.

2.1 Notation

For better or worse, the notation found in dynamical systems and control theory, pattern recognition, and image analysis tends to vary considerably from one field to another. A consistent notation has been used throughout the thesis, even though at times this means departing from some of the conventions used in the corresponding research literature. Variables or symbols used within limited contexts are described within their context and are not listed here.
Typically, lower case letters (e.g., x and y) will denote scalars, whilst vectors are denoted by lower case bold-face letters (e.g., $\boldsymbol{x}$ and $\boldsymbol{y}$) and all vectors are assumed to be column vectors (i.e., they post-multiply matrices) and where the i-th element of a vector $\boldsymbol{x}$ is denoted by $\boldsymbol{x}_i$. Matrices are denoted by upper case bold-face letters (e.g., $\boldsymbol{A}$ and $\boldsymbol{B}$), whilst sets and sequences are denoted by either blackboard bold or upper case bold-face letters (e.g., $\mathbb{A}$ or $\boldsymbol{A}$, respectively) however, the meaning should be clear from the context. The i-th row of a matrix is denoted by $\boldsymbol{A}_{i,:}$, the j-th column of a matrix is denoted by $\boldsymbol{A}_{:,j}$, whilst a single element $(i,j,)$ of a matrix is denoted as usual by $\boldsymbol{A}_{i,j}$. A superscript $\top$ denotes the transpose of a matrix or vector, so that $\boldsymbol{x}^\top$ represents a row vector. The $n \times n$ identity matrix is denoted $\mathbf{I}_n$, which will be abbreviated to $\mathbf{I}$ where there is not ambiguity about the dimensionality that is implied by context. Tensors are denoted by calligraphic bold-face letters (e.g., $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$). Besides the standard mathematical conventions, the following notations will be used throughout this thesis.

- The set of natural numbers, including 0, is represented by $\mathbb{N}$; the set of integer numbers is represented by $\mathbb{Z}$; the set of real numbers is represented by $\mathbb{R}$, whilst $\mathbb{R}_+$ $(\mathbb{R}_{++})$ is the set of non-negative (positive) real numbers.
- A n-square matrix $\boldsymbol{A}$ is called positive if $A_{i,j} > 0$ for $i,j \in \{0,1,\ldots,n\}$ and non-negative if the strict inequality is replaced by a weak inequality.
- A square, diagonal matrix with diagonal entries given by $\boldsymbol{a}$ will be denoted by $\mathrm{diag}(\boldsymbol{a})$.
- Given a square matrix $\boldsymbol{A}$, $\det(\boldsymbol{A})$ is its determinant, $\mathrm{tr}(\boldsymbol{A})$ is its trace (sum of diagonal elements), $\sigma(\boldsymbol{A})$ is its spectrum (set of the eigenvalues).

- If $\mathbf{A}$ and $\mathbf{B}$ are matrices (or vectors) of the same dimensions, then $\mathbf{A} > \mathbf{B}$, $\mathbf{A} \geqslant \mathbf{B}$, $\mathbf{A} < \mathbf{B}$, $\mathbf{A} \leqslant \mathbf{B}$, have to be intended componentwise ($A_{i,j} > B_{i,j}$, $A_{i,j} \geqslant B_{i,j}$, $A_{i,j} < B_{i,j}$, $A_{i,j} \leqslant B_{i,j}$, for all valid i and j).
- The symbols $\succ \succeq$, $\prec$, and $\preceq$ denote order relations induced by definiteness properties. In the space of symmetric matrices $\mathbf{Q} \succ \mathbf{P}$, $\mathbf{Q} \succeq \mathbf{P}$, $\mathbf{Q} \prec \mathbf{P}$, $\mathbf{Q} \preceq \mathbf{P}$, denote that $\mathbf{P} - \mathbf{Q}$ is positive definite, positive semi-definite, negative definite, and negative semi-definite.

With a slight abuse of notation, sometimes we refer to a function ψ whose argument is a scalar but apply it to a vector, matrix or tensor $\psi(\mathbf{x})$, $\psi(\mathbf{X})$, $\psi(\mathcal{X})$), denoting the application of the function to the array element-wise (e.g., if $\mathcal{Y} = \psi(\mathcal{X})$, then $Y_{i,j,k} = \psi(X_{i,j,k})$, for all valid i, j, and k).

The mathematical model of an image as a function of two real spatial parameters is enormously useful in both describing images and defining operations on them. In equation form, the representation of a $w \times h$ numerical array is

$$\psi(x, y) = \begin{bmatrix} \psi(0,0) & \psi(0,1) & \dots & \psi(0, w-1) \\ \psi(1,0) & \psi(1,1) & \dots & \psi(1, w-1) \\ \vdots & \vdots & & \vdots \\ \psi(h-1,0) & \psi(h-1,1) & \dots & \psi(h-1, w-1) \end{bmatrix}$$

however, throughout the thesis, a more traditional matrix notation has been used to denote a digital image and its elements

$$\mathbf{A} = \begin{bmatrix} A_{0,0} & A_{0,1} & \dots & A_{0,w-1} \\ A_{1,0} & A_{1,1} & \dots & A_{1,w-1} \\ \vdots & \vdots & & \vdots \\ A_{h-1,0} & A_{h-1,1} & \dots & A_{h-1,w-1} \end{bmatrix}.$$

If the elements of the matrix are vectors, then will be used the tensor notation. If $\mathcal{X}$ represnts an $w \times h$ digital image whose elements are represented by a triple of real values, then $X_{i,j,k}$ is the element (i, j, k) of $\mathcal{X}$ and $\mathbf{X}_{:,:,k}$ is the two-dimensional slice of the three-dimensional tensor $\mathcal{X}$ (i.e., the matrix represnting the monochrome channel k).

2.2 Metzler matrices

A class of matrices that is fundamental when dealing with monotone systems is known as Metzler matrices.

Definition 1. *A n-square matrix $\mathbf{A}$ is a* Metzler matrix *if it has non-negative off-diagonal entries, i.e., $A_{i,j} \geqslant 0, \forall i \neq j$.*

Definition 2. *A n-square matrix $\mathbf{A}$ is a* Metzler Hurwitz matrix *if it is a Metzler matrix and the real part of each element of the spectrum of $\mathbf{A}$ is negative.*

It is worth noticing that any Metzler matrix $\mathbf{A}$ can be represented as:

$$\mathbf{A} = \mathbf{A}_+ - \lambda \mathbf{I}_n \qquad \lambda \in \mathbb{R},\ \mathbf{A}_+ \in \mathbb{R}_+^{n \times n} \tag{2.1}$$

As a consequence, since the spectrums of $\mathbf{A}$ and $\mathbf{A}_+$ are linked to each other through the relation $\sigma(\mathbf{A}) = \sigma(\mathbf{A}_+) - \lambda$, then Equation (2.1) defines a Metzler Hurwitz matrix if and only if $\lambda > \lambda_{\max}$, where $\lambda_{\max}$ is the Frobenius eigenvalue of $\mathbf{A}_+$.

Some useful properties are listed in the following proposition[1].

Proposition 1. *Let $\mathbf{A} \in \mathbb{R}_+^{n\times n}$ be a Metzler matrix, then the following statements are equivalent:*

1. *the matrix $\mathbf{A}$ is a Metzler Hurwitz matrix;*
2. *the coefficients of the characteristic polynomial of $\mathbf{A}$ (i.e., $\Delta_{\mathbf{A}}(z)$) are all positive;*
3. *$\mathbf{A}$ is non-singular and $\mathbf{A}^{-1}$ is a negative matrix, i.e., $A_{i,j} < 0$ for all valid i and j;*
4. *the matrix $-\mathbf{A}$ is a P-matrix, i.e., all its principal minors are positive;*
5. *$\mathbf{A}$ has all negative diagonal elements and there exists a positive diagonal matrix $\mathbf{D}$ such that $-\mathbf{D}^{-1}\mathbf{A}\mathbf{D}$ is strictly diagonally dominant;*
6. *there exist lower and upper triangular matrices $\mathbf{L}$ and $\mathbf{U}$, with positive diagonals such that $\mathbf{A} = -\mathbf{L}\mathbf{U}$.*

2.3 Positive systems

A positive system is a state-space model of the form

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \end{aligned} \tag{2.2}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^p$ denote the state, input, and output, respectively. We can distinguish between two different notions of positivity, namely external and internal positivity [43].

Definition 3. *A linear sistem $\Sigma(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is said to be* externally positive *iff for every non-negative initial state and for every non-negative input its state and output are non-negative.*

Definition 4. *A linear sistem $\Sigma(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is said to be* internally positive *iff its state output (i.e., the output corresponding to a zero initial state) is non-negative for every non-negative input.*

Let us consider the homogeneous LTI discrete-time dynamic system

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) \tag{2.3}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, hence, the following theorem follows [13].

Definition 5. *The system is said positive if*

$$\mathbf{x}(0) \geqslant 0 \;\Rightarrow\; \mathbf{x}(k) \geqslant 0, \quad \forall k > 0. \tag{2.4}$$

where the inequalities above should be intended component-wise.

Theorem 1. *A (homogeneous) discrete-time linear system is positive if and only if the state matrix is non-negative (Metzler property of the state matrix in the continuous time case).*

[1] See [11] for a more exhaustive characterisation of M-matrices (rom which it is possible to easily derive the characterisation of Metzler Hurwitz matrices.

As far as asymptotic stability is concerned, it is worth noticing that asymptotic stability with respect to initial conditions in the positive orthant is equivalent to asymptotic stability with respect to arbitrary initial conditions in $\mathbb{R}^n$. Indeed, any initial state $\mathbf{x}(0) \in \mathbb{R}^n$ can be expressed as $\mathbf{x}(0) = \mathbf{x}(1) - \mathbf{x}(2)$ where $\mathbf{x}(1), \mathbf{x}(2) \in \mathbb{R}_{++}$. Hence, all the conditions for asymptotic stability of linear systems are valid for positive systems. Nevertheless, positive systems have some peculiar properties that are quite useful for the stability analysis, as shown in the following proposition.

Proposition 2. *Let $\mathbf{A} \in \mathbb{R}_+^{n\times n}$, then the following statements are equivalent:*

1. *all the eigenvalues λ_i of $\mathbf{A}$ have modulus less than 1, i.e., $|\lambda_i| < 1,\ i = 1, 2, \dots n$;*
2. *the matrix $\mathbf{A} - \mathbf{I}_n$ is a Metzler Hurwitz matrix;*
3. *the coefficients of the characteristic polynomial of $\mathbf{A} - \mathbf{I}_n$ (i.e., $\Delta_{\mathbf{A}-\mathbf{I}_n}(z)$) are all positive.*

A further property is the Perron-Frobenius theorem.

Theorem 2. *The eigenvalue with the largest modulus (dominating eigenvalue, also denoted as Frobenius eigenvalue) of a non-negative matrix is non-negative and real, that is, if we order the eigenvalues by magnitude*

$$\lambda_1 = |\lambda_1| \geqslant |\lambda_2| \geqslant \dots \geqslant |\lambda_n|.$$

Moreover, if λ_1 is strictly dominating and irreducible (see [13])

$$\lambda_1 = |\lambda_1| > |\lambda_2| \dots$$

then the corresponding eigenvector (Frobenius eigenvector) has positive components.

Another property is that the origin of the homogeneous linear time-invariant system (2.2) is globally asymptotically stable if and only if all eigenvalues of $\mathbf{A}$ have norm strictly less than one; i.e. the spectral radius $\rho(\mathbf{A})$ is less than one. In this, we call the matrix $\mathbf{A}$ stable (or *Schur stable*).

2.3.1 Monotonicity

A monotone dynamical system is just a dynamical system on an ordered metric space which has the property that ordered initial states lead to ordered subsequent states [133]. When dealing with linear time-invariant systems, this is equivalent to the non-negativity (Metzler) property of the state matrix [13], which is also equivalent to saying that the system is *monotone* [133].
Denoting by $\mathbf{x}(k) = f(k, \omega)$ the solution at time instant k of the initial value problem

$$\mathbf{x}(k+1) = f(x(k))$$

$$\mathbf{x}(0) = \omega$$

then the system is monotone if

$$\mathbf{x}(0) \geqslant \mathbf{y}(0) \ \Rightarrow\ f(k, \mathbf{x}(0)) \geqslant f(k, \mathbf{y}(0)), \quad \forall k \geqslant 0$$

and this is true if and only if the state matrix is non-negative (Metzler) [6, 12].

I

Biometrics

3

Background

This chapter stands as an introduction to the field of biometrics. It is important to present the concepts and primitives of performance metrics due to their impact on secure biometric systems. Thus, a brief overview is given to describe the main biometric traits along with their properties as well as the various biometric system operating modalities. Finally, the criteria for performance evaluation have been defined to determine the system accuracy and security which are related to the applicability in real-world deployments.

3.1 Taxonomy of biometric traits

With the rapid growth in demand for reliable and highly secure human authentication and identification systems, the importance of technological solutions and algorithms in the biometric field is growing along with security awareness [71]. In fact, traditional/conventional authentication methods, consisting in token-based systems that make use of something you have (e.g., ID card), and knowledge-based systems that make use of something you know (e.g., personal identification number (PIN) or password), are unable to meet the needed reliability and security requirements, while biometric systems make use of physiological (intrinsic) and/or behavioural (extrinsic) traits of individuals, overcoming the security issues affecting the conventional methods for personal authentication [122]. As a matter of fact, various biometric modalities have been developed over the years making the biometric technology landscape very vibrant. Prominent examples of physiological and behavioural biometric charracteristics, which have been the purpose of major real-world applications, are illustrated in Figure 3.2 and briefly described below.

Fingerprint

This trait appears as a series of dark lines and white spaces, when captured from device and it consists of set of ridges and valleys (minutiae) located on the surface tips of a human finger to uniquely distinguish individuals from each other [100].

Figure 3.1: Example of a fingerprint image.

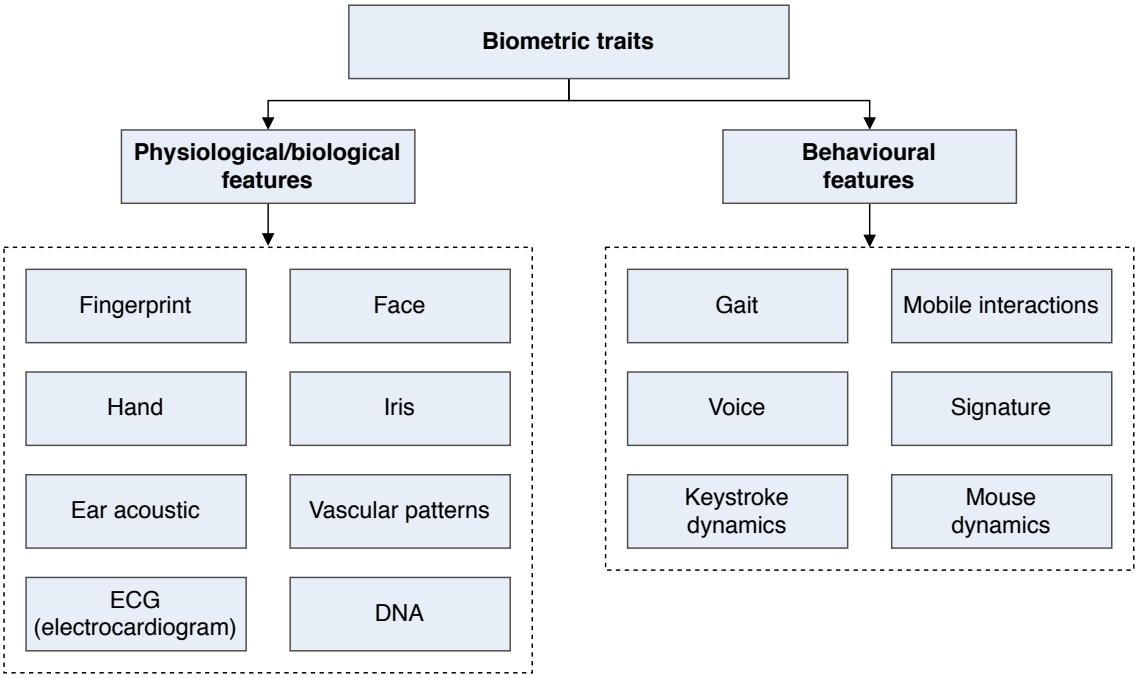


Figure 3.2: Examples of physiological/biological and behavioural traits applied in biometric recognition applications.

Face

Facial features use the location and shape (geometry) of the face, including the distance between the eyes, the distance from the chin to the forehead, or other measures that involve eyebrows, nose, lips, and jaw line (just to name a few) [157].

Figure 3.3: Example of facial recognition.

Hand geometry

Hand geometry is based on the geometric characteristics of the hand such as the length and width of fingers, their curvature, and their relative position to other features of the hand [128].

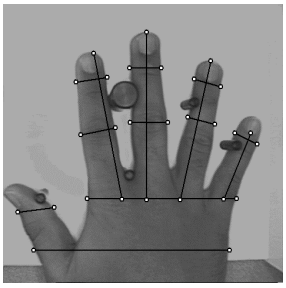


Figure 3.4: Example of a recognition based on hand geometry.

Iris

The iris is the colored part in the eye that consists of thick, thread-like muscles characterised by unique folds and patterns that can be used to identify and verify the identity of humans [33].

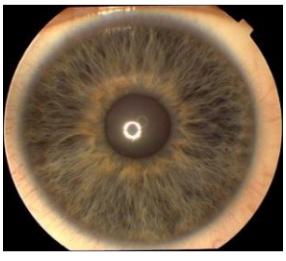


Figure 3.5: Example of an iris image.

Ear acoustic

An ear signature is generated by probing the ear with an inaudible sound waves which are reflected bouncing in different directions and picked up by a small microphone. The shape of the pinna and the ear canal determine the acoustic transfer function which forms the basis of the signature [3].

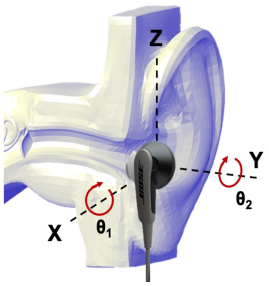


Figure 3.6: Example of ear acoustic.

Vascular patterns

Vascular biometrics is based on a type of biometric trait based on the network structure of blood vessels that can be used to identify individuals ensuring confidentiality and robustness to counterfeiting [119].

Figure 3.7: Example of a retina image.

ECG

The human heart and body anatomic features form the shape of the ECG signal, which reports the strength and timing of the electrical activity of the heart [151].

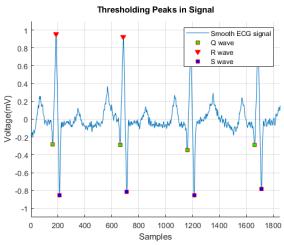


Figure 3.8: Example of an electrocardiogram (ECG) signal.

DNA

DNA matching is based on a common molecular biology method named short tandem repeat (STR) analysis, which is used to compare allele repeats at specific locations on a chromosome in DNA between two or more samples [73, 139].

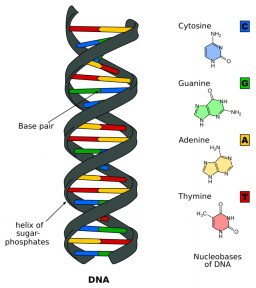


Figure 3.9: Example of deoxyribonucleic acid (DNA) structure.

Gait

This characteristic may be changeable over a large time span due to various reasons such as weight gain [113]. Thus, it can be used in low-security applications for massive crowd surveillance as it can quickly identify people from afar based on their walking style [103].

Figure 3.10: Example of gait recognition.

Mobile interactions

It is based on the unique ways in which users swipe, tap, pinch-zoom, type, or apply pressure on the touchscreen of mobile devices like tablets and phones, thus providing characteristic patterns that may be used to identify people [47].

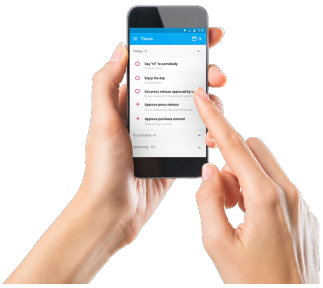


Figure 3.11: Example of phone in hand.

Voice

Voice biometric recognition allows to distinguish among humans' voice for personal authentication as voice features include physical characteristics such as vocal tracts, nasal cavities, mouth, and larynx [34].

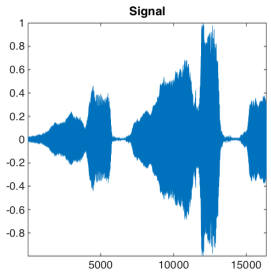


Figure 3.12: Example of voice signal and fundamental frequency estimation over time.

Signature

Signature biometric features are extracted by analysing curves, edges, spatial coordinates, pen pressure, and pen stroke of the signature samples in both off-line (static) and on-line (dynamic) applications [35].

Figure 3.13: Example of signature (David Bowie).

Keystroke dynamics

Keystroke dynamics include the tracking of the rhythm used to type on a keyboard or keypad analysing the variation of duration between consecutive keystrokes, placement of finger, pressure and durations of keystrokes [79].

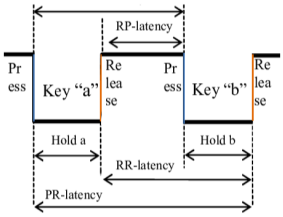


Figure 3.14: Example of keystroke dynamics features from a digraphs sequence.

Mouse dynamics

Mouse dynamics makes use of patterns in mouse or trackpad cursor movement including clicks, paths, direction changes, tracking speed, and the relationships between them to authenticate individuals [129].

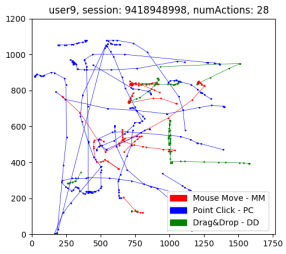


Figure 3.15: Example of mouse dynamics features.

3.2 Properties of biometric traits

The main requirements that should be satisfied before a trait can be characterised as suitable for its applicability in a biometric recognition system, are briefly discussed as follows [71].

Universality
Every individual, or at least the most of them, accessing the biometric application should possess the characteristic.

Distinctiveness
The given trait should be sufficiently different across individuals comprising the user population. Otherwise, the proportion of times the biometric system grants access to unauthorised individuals would be unacceptably high.

Permanence
The biometric trait of an individual should be sufficiently invariant (with respect to the matching criterion) over a period of time. This implies that the given trait should not change significantly over time otherwise the proportion of times the biometric system denies access to authorised individuals would be unacceptably high.

Collectability
The biometric trait can be measured quantitatively with particular regard to the easiness of obtaining the biometric data using suitable devices that do not cause undue inconvenience to the user.

Even though any human characteristic can be used as a biometric trait as long as the previous requirements are satisfied, in real-world biometric recognition applications there are a number of other issues that should be considered, such as:

Performance
This is a property aimed at assessing the verification or identification accuracy, the computational time required for a single recognition, as well as the operational and environmental factors that may affect or not the recognition accuracy and speed.

Acceptability
It indicates the extent to which people are willing to accept the use of a specific biometric application as well as their willingness to provide their biometric data.

Circumvention
This property reflects how easily the system can be deceived through potential spoofing attacks. It refers to the ways in which an attacker can endeavor to bypass a biometric system and finally attack the weakness spot of such a system in order to gain unauthorised access.

Real-life biometric recognition systems ought to meet the requirements of accuracy, speed, and resource constraints, be harmless to the users, be accepted by the intended population as well as sufficiently robust to various fraudulent methods and attacks to the system [72].

3.3 Biometric system operating modes

A biometric system can provide two kinds of operating modes (identity management functionalities), namely, *verification* and *identification*. Biometric systems can indeed automatically authenticate[1] or identify subjects in a reliable and fast way and are therefore suitable to be used in a wide range of applications to face the risks of unauthorised logical or physical access and identity theft, as well as new threats such as terrorism or cybercrime [67]. Figure 3.16 provides a high-level view of a generic biometric recognition system as well as all its basic building blocks.

[1]Throughout this thesis as in literature, the term authentication will be used as a synonym for verification.

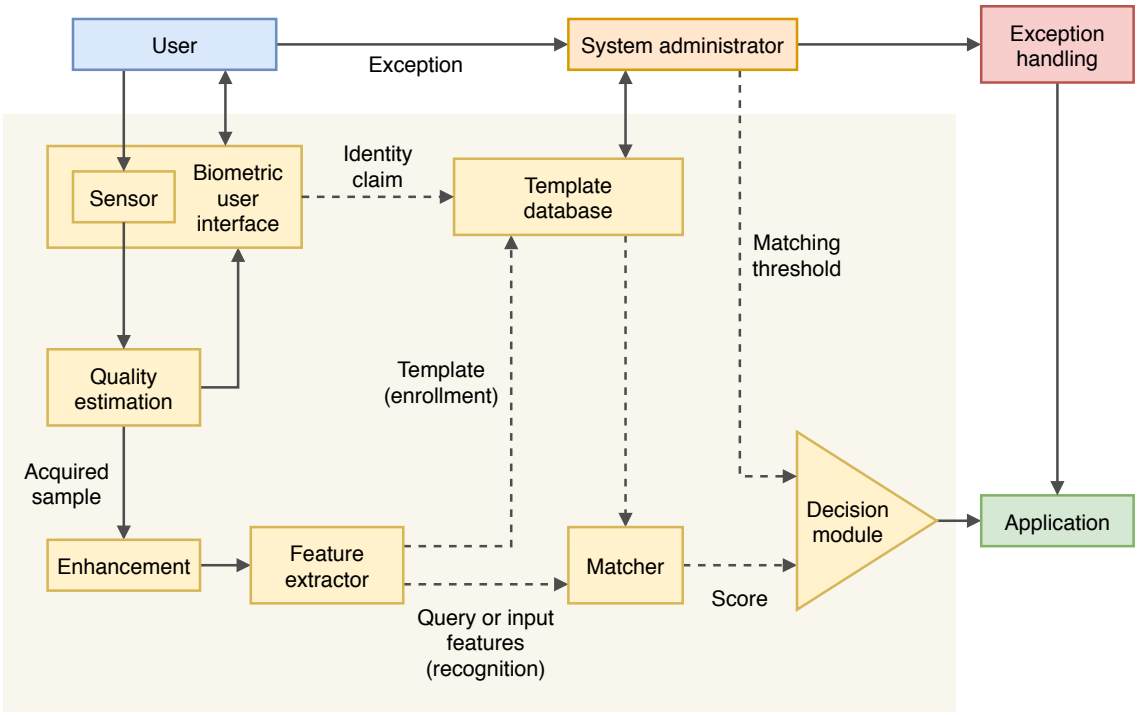


Figure 3.16: Basic building blocks of a generic biometric recognition system.

3.3.1 Authentication

In the authentication mode, the purpose of the biometric system is to verify whether a individual's claimed identity is genuine or not (binary classification). Thus, the captured biometric data (query) is compared only with the biometric template(s) stored in the system database and corresponding to the claimed identity (one-to-one or one-to-few comparison).

Definition 6. *Given a claimed identity* I *and a query feature set* $\mathbf{x}^{\mathrm{Q}}$*, the biometric system has to categorise* $(\mathrm{I}, \mathbf{x}^{\mathrm{Q}})$ *into "genuine" or "impostor" class. Let* $\mathbf{x}_{\mathrm{I}}^{\mathrm{E}}$ *be the stored biometric template corresponding to the identity* I *(i.e., the enrolled user with identity* I*). The similarity measure between* $\mathbf{x}^{\mathrm{Q}}$ *and* $\mathbf{x}_{\mathrm{I}}^{\mathrm{E}}$ *gives, as a result, a matching score. Hence, the biometric system applies the decision rule given by*

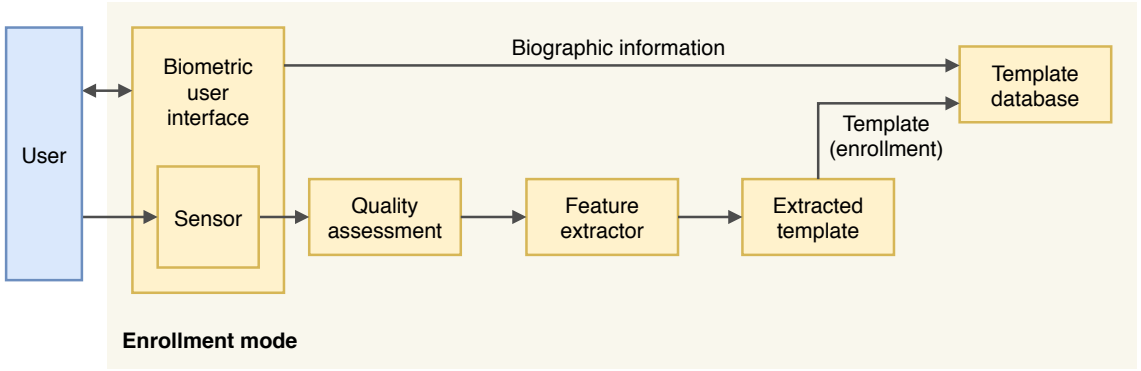
$$(\mathrm{I}, \mathbf{x}^{\mathrm{Q}}) \in \begin{cases} \textit{genuine}, & \textit{if } s\left(\mathbf{x}^{\mathrm{Q}}, \mathbf{x}_{\mathrm{I}}^{\mathrm{E}}\right) \geqslant \xi, \\ \textit{impostor}, & \textit{otherwise}, \end{cases} \tag{3.1}$$

where s *represents a similarity function and* ξ *represents a pre-defined threshold at which the system is intended to operate.*

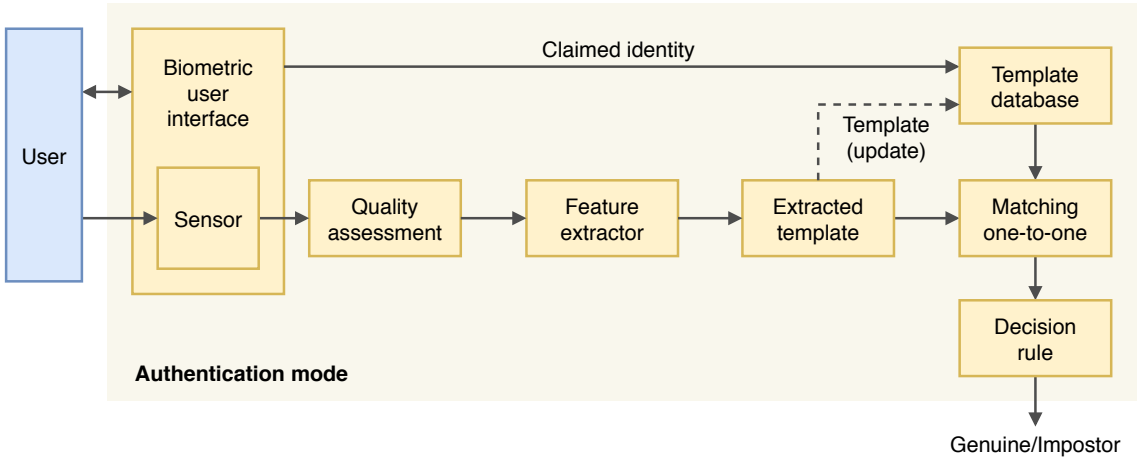
The authentication mode is typically employed for positive recognition, where the aim is to prevent multiple people from using the same identity [145].

3.3.2 Identification

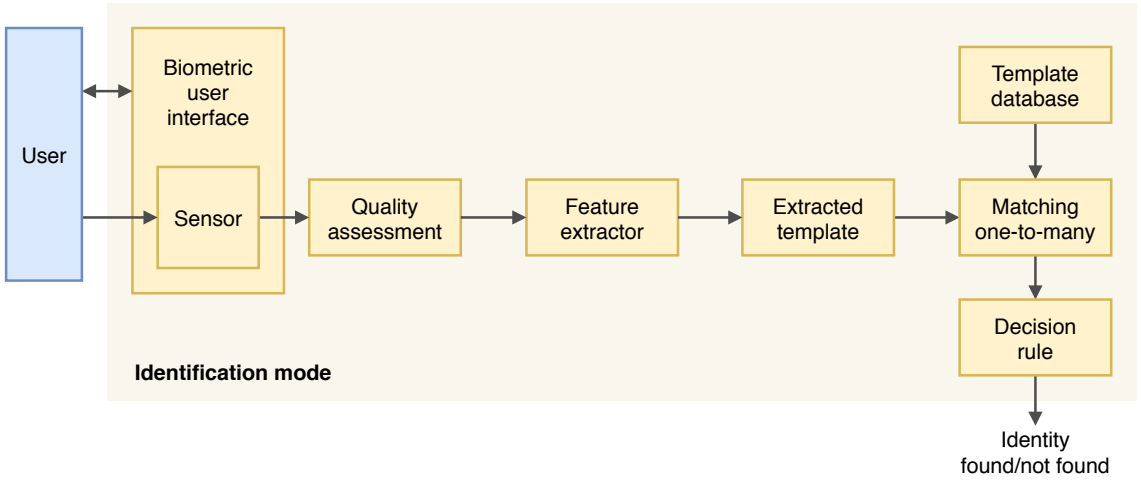
In the identification mode, the purpose of the biometric system is to recognise an individual's identity by searching the templates of all the enrolled individuals in the system database for a match (one-to-many comparison) without the subject having to claim an identity.



(a)



(b)



(c)

Figure 3.17: Different operating modes of a biometric system: (a) enrollment mode, (b) authentication mode (the dashed line is an optional operation aimed at updating a specific user's template), and (c) identification mode.

This operating mode can be further splitted into negative and positive identification: in the negative identification (also known as *screening*), the user is considered to be hiding her/his true identity from the biometric system, whilst in the positive identification, the user tries to positively identify herself/himself to the system without explicitly claiming an identity.

Definition 7. *Given a query feature set $\mathbf{x}^Q$, the biometric system has to determine the identity I_k $\forall k \in \{1, 2, \ldots, n, n+1\}$ where $\{I_1, I_2, \ldots, I_n\}$ are the identities of the enrolled users in the system, whilst I_{n+1} represents the failure case where no identity can be assigned for the given query (*open set identification*). Hence, assuming that $\mathbf{x}^E_{I_k}$ is the stored template corresponding to the identity I_k, the biometric system applies the decision rule given by*

$$\mathbf{x}^Q \in \begin{cases} I_k, & \textit{if } \max\limits_k \left\{ s\left(\mathbf{x}^Q, \mathbf{x}^E_{I_k}\right) \right\} \geqslant \xi, \\ I_{n+1}, & \textit{otherwise,} \end{cases} \tag{3.2}$$

where s represents a similarity function and ξ represents a pre-defined threshold at which the system is intended to operate.

The identification mode is typically employed for screening[2], where the aim is to prevent a single person from using multiple identities [145].

3.4 Criteria for performance evaluation

The reliability and validity of a biometric scheme as well as the selection of a certain biometric trait for an application are determined by specific measures that are used to evaluate the recognition accuracy and effectiveness as addressed in ISO/IEC Standards [69]. Accordingly, in order to evaluate the accuracy of the proposed method based on a single-sample approach for unimodal biometric systems, each sample in the database has undergone a one-to-one matching test against every single stored sample. Hence, a comparison between a subject with real identity I_r and a subject with claimed identity I_c is aimed at testing the hypothesis:

$$H_0 : \{I_r = I_c\} \qquad \text{versus} \qquad H_1 : \{I_r \neq I_c\} \tag{3.3}$$

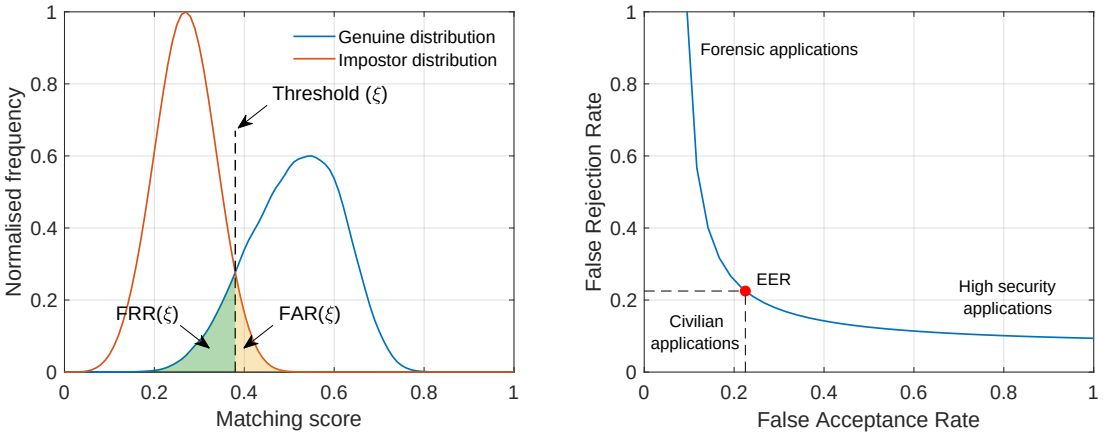
where H_0 is the null hypothesis that the user is who s/he claims to be (genuine or intra-class matching), whilst H_1 is the alternative hypothesis that the user is not who s/he claims to be (impostor or inter-class matching). To test the hypothesis in (3.3), it is required to compute a similarity measure, $s(Q, T)$ where large (respectively, small) values of s indicate that the template T of the claimed identity I_c in the database and the biometric query Q of a real user I_r are close to (far from) each other. Formally, the verification problem consists of determining if a claimed identity I with biometric data Q belongs to the class H_0 or not:

$$(I, Q) = \begin{cases} H_0, & \text{if } s(Q, T) \geqslant \xi, \\ H_1, & \text{otherwise.} \end{cases} \tag{3.4}$$

[2] In some real scenario, such as latent palmprint matching, it is preferable to use a semi-automated approach aimed at providing the top n identities that best match to the given template for further analysis by a human expert. Alternatively, it is possible to consider all the identities whose corresponding match scores exceed the threshold ξ in Equation (3.2), which leads to a challenging task in a quite large database (e.g., FBI's Next Generation Identification (NGI) system, which provides the world's largest repository of biometric and criminal history information [44]).

In particular, given a threshold value, ξ, all matching values s lower (respectively, greater) than ξ lead to the rejection (acceptance) of the null hypothesis H_0 [32]. Therefore, whether the hypothesis is accepted or rejected, the test is prone to two kinds of error:

1. false acceptance rate (FAR) that is the probability of accepting the null hypothesis H_0 when input is not valid (type-I error),
2. false rejection rate (FRR) that is the probability of rejecting the null hypothesis H_0 when input is valid (type-II error).



(a) (b)

Figure 3.18: Examples of biometric system error rates: (a) FAR and FRR for a given threshold ξ are displayed over the genuine and impostor score distributions and (b) typical operating points of different biometric applications are displayed on a ROC curve aimed at relating FAR and FRR at different threshold values.

Definition 8. *Let H_0 and H_1 be the labels that denote the genuine and impostor classes, respectively. Assume also that the $p(s|H_0)$ and $p(s|H_1)$ represent the probability density functions of the genuine and impostor scores, respectively. Then the FAR and FRR distributions are given by:*

$$FAR(\xi) = p(s \geqslant \xi|H_1) = \int_{\xi}^{+\infty} p(s|H_1)ds, \tag{3.5}$$

$$FRR(\xi) = p(s < \xi|H_0) = \int_{-\infty}^{\xi} p(s|H_0)ds. \tag{3.6}$$

The false acceptance rate and false rejection rate are functions of the system threshold ξ and are closely related because the increase of the one implies the decrease of the other. Hence, for a given biometric system, it is not possible to decrease both these errors at the same time by varying the threshold ξ [72]. The genuine acceptance rate (GAR) is instead the probability of accepting the null hypothesis H_0 when input is valid, hence it can be used as an alternative to FRR:

$$GAR(\xi) = p(s \geqslant \xi|H_0) = 1 - FRR(\xi). \tag{3.7}$$

Depending on the security level required by the final application (i.e., forensics, surveillance and homeland security, civilian, or high security applications), the same biometric system may operate at different threshold values (ξ), as illustrated in Figure 3.18(b). Hence, in order to evaluate the biometric system performance as a function of the threshold ξ, the following curves can be considered:

- the receiver operating characteristic (ROC) is a graphical plot that illustrates the trade-off between false acceptance rate and false rejection rate when the threshold varies, whilst the intersection point for which rejection and acceptance errors are equal is named equal error rate (EER). The curve is generated by plotting the genuine acceptance rate against the false acceptance rate at various threshold settings,
- the detection error trade-off (DET) is another graphical plot that illustrates the false rejection rate against the false acceptance rate at various threshold values. The two axes are scaled non-linearly by their standard normal deviates[3] or just by logarithmic transformation.

Furthermore, the above-mentioned receiver operating characteristic and detection error trade-off curves are threshold independent, allowing performance comparison of different biometric systems under similar conditions.

Definition 9. *Given a set of thresholds* $\{\xi_i\} \mid s_{min} \leqslant \xi_i \leqslant s_{max}\ \forall i \in \{1, 2, \ldots, n\}$ *where* s_{min} *and* s_{max} *are the minimum and maximum scores, respectively, in a given set of match scores* $\{s_i\} \mid 0 \leqslant s_i \leqslant 1\ \forall i \in \{1, 2, \ldots, n\}$. *Then, it is possible to generate a ROC curve computing the overall FAR and FRR for each threshold* ξ, *as follows:*

$$FAR = \frac{1}{N} \sum_{k=1}^{N} FAR(\xi), \tag{3.8}$$

$$FRR = \frac{1}{N} \sum_{k=1}^{N} FRR(\xi), \tag{3.9}$$

where N *represents all identities being evaluated by the system and*

$$FAR(\xi) = \frac{number\ of\ FARs}{number\ of\ impostor\ accesses}, \tag{3.10}$$

$$FRR(\xi) = \frac{number\ of\ FRRs}{number\ of\ genuine\ accesses}. \tag{3.11}$$

[3] In the normal deviate scale, the threshold values ξ correspond to linear multiples of standard deviation σ of a Gaussian distribution. Thus, if the false acceptance rate and false rejection rate distributions are Gaussian, the corresponding detection error trade-off curve would be linear [72].

4

A unified method for line-like feature matching

This chapter aimed at highlighting the main idea of the proposed unified method for line-like feature matching, which relies on a recursive algorithm based on a positive linear dynamical system whose evolution depends on the two matrices representing the templates to be compared. Given two image templates[1] (matrices of Boolean values) $\mathbf{X}, \mathbf{Y} \in \{0,1\}^{w \times h}$ of equal dimensions, for authentication purposes, it may be convenient adopting an operator of the form

$$\left(\tilde{\mathbf{X}}, \tilde{\mathbf{Y}}\right) = \Phi\left(\mathbf{X}, \mathbf{Y}\right), \tag{4.1}$$

which yields a pair of real matrices $\tilde{\mathbf{X}}, \tilde{\mathbf{Y}} \in \mathbb{R}^{w \times h}$, having the same size as $\mathbf{X}$ and $\mathbf{Y}$. For the authentication, the operator Φ must be chosen to assess the "matching level" between the two initial template images. To simply explain the idea, consider a simple (i.e., non dynamic) mismatch function.
A simple option to assess the matching level of two images is to consider the conjunction $\tilde{\mathbf{X}} = \tilde{\mathbf{Y}} = \mathbf{X} \wedge \mathbf{Y}$, associated with the truth-functional operator of logical conjunction:

$$[\mathbf{X} \wedge \mathbf{Y}]_{i,j} = \neg\left(\neg X_{i,j} \vee \neg Y_{i,j}\right) = X_{i,j} \wedge Y_{i,j} = \min\left(X_{i,j}, Y_{i,j}\right). \tag{4.2}$$

However, the approach based on this choice for the operator in (4.1) is clearly not noise-rejecting, since it evaluates the correspondence between single points (isolated from their context), which may be noisy. Conversely, to avoid a misleading recognition, noisy pixels that are accidentally matching in the two images should not provide a positive contribution to the matching score.
Thus, it has been pursued an approach that rejects noise and, to this aim, it has been taken into account the following assumption.

Assumption 1. *The matching of "isolated spots" is not as significant as the matching of cluster of points, whether they are wide "stripes" or "island", even if the number of matching isolated spots is very high.*

[1] The two image templates given to the dynamical system consist of Boolean matrices containing the extracted line-like features.

4.1 The dynamical system

The idea behind the algorithm is to recursively increase the value of a pixel i, j if in the complementary neighbourhood (namely, the neighbourhood of the corresponding pixel i, j in the comparison image) there are pixels with large values. Conversely, if the pixels in the complementary neighbourhood have low values, the value of pixel i, j converges to zero.

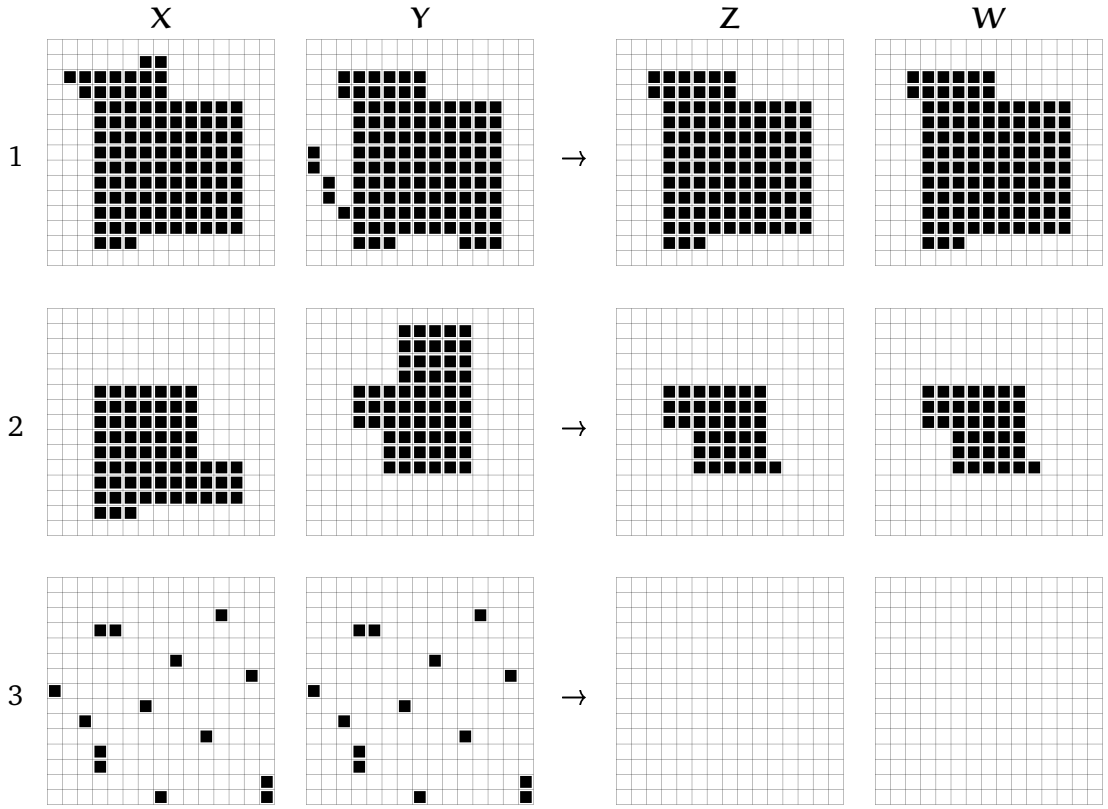


Figure 4.1: Desired matching behaviour for different patterns (active pixels in black) where X and Y are the initial matrices, whilst $Z = \tilde{X}$ and $W = \tilde{Y}$ are the desired transformed matrices. The first row illustrates an example of good matching, the second row presents an example of partial matching, and the last one depicts an example of no matching.

Hence, the operator Φ described in Equation (4.1) must follows the behaviour illustrated in Figure 4.1.

1. In the first example, two significant sets of active pixels[2] are surrounded by 0 pixels in both images X and Y, which show a good superposition or a *good matching*. Actually, almost all of the active pixels should survive in the transformed images $Z = \tilde{X}$ and $W = \tilde{Y}$.
2. In the second example, there is a just *partial superposition*. Hence, in both of the transformed images, the sets of active pixels should be partially eroded.
3. In the third example, there is a *perfect superposition* between *isolated points*. In this case, the transformed images should be almost empty.

[2]By convention an active point has been represented in black, whilst a non-active point has been represented in white.

As is experimentally shown later, the behaviour illustrated in Figure 4.1 can be achieved by means of the proposed unified method that involves a positive system, whose evolution depends on the matching level between the two input images $\mathbf{X}$ and $\mathbf{Y}$.

Remark 1. *In view of the iterative nature of the algorithm, it is only necessary to emphasise the presence of neighbourhoods in the comparison image, since in at most two steps this indirectly emphasises neighbourhoods in the image itself.*

Assumption 2. *It is further assumed that the images $\mathbf{X}$ and $\mathbf{Y}$, provided as inputs, have already undergone a preliminary processing (thoroughly described in Section 4.4.2) and that therefore contain the extracted line-like features.*

Given the square neighbourhood of the generic point i, j within a "radius" δ of integer amplitude grater than zero

$$\mathcal{N}_{i,j} = \{h, l : \|h - i\| \leqslant \delta,\ \|l - j\| \leqslant \delta,\ h, l \in \mathbb{Z}\}, \tag{4.3}$$

then the two images are processed according to the iterations

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k) \tag{4.4}$$

$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k) \tag{4.5}$$

where $k = 0, 1, \ldots, K-1$, λ represents the *fading factor*, and μ is the coefficient of the cross-matching part. At the K-th step, in order to achieve a Boolean image, pixels with value smaller than 1 are set to zero whilst pixels with value greater than 1 are saturated to 1. The saturation function is defined as follows.

Definition 10. *Let $\mathbf{X}$ and $\mathbf{Y}$ be the input real matrices of dimensions $n \times n$, then the piecewise-defined function $\Theta : \mathbb{R} \to \{0, 1\}$ given by*

$$\Theta(x) \doteq \begin{cases} 1 & \text{if } x \geqslant 1, \\ 0 & \text{if } x < 1, \end{cases} \tag{4.6}$$

generates the Boolean matrices $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$, having the same size as $\mathbf{X}$ and $\mathbf{Y}$, by means of the following computation for all i, j

$$\tilde{X}_{i,j} = \Theta\left(X_{i,j}\right), \qquad \tilde{Y}_{i,j} = \Theta\left(X_{i,j}\right). \tag{4.7}$$

The final score (or equivalently, the matching index) is performed on the number of pixels with value 1 (active), which is compared to the initial number as described in the following definition.

Definition 11. *Let $\mathbf{X}, \mathbf{Y} \in \{0, 1\}^{n \times n}$ be the input Boolean matrices and $\tilde{\mathbf{X}}, \tilde{\mathbf{Y}} \in \{0, 1\}^{n \times n}$ be the output Boolean matrices. Denoting by $\Sigma(\mathbf{X})$ and $\Sigma(\mathbf{Y})$ the number of active pixels in the initial matrices and by $\Sigma(\tilde{\mathbf{X}})$ and $\Sigma(\tilde{\mathbf{Y}})$ the number of active pixels in the output matrices, then the* matching index *between $\mathbf{X}$ and $\mathbf{Y}$ has been defined as*

$$\alpha(\mathbf{X}, \mathbf{Y}) = \frac{\gamma(\mathbf{X}, \mathbf{Y})}{2} \left[\frac{\Sigma(\tilde{\mathbf{X}})}{\Sigma(\mathbf{X})} + \frac{\Sigma(\tilde{\mathbf{Y}})}{\Sigma(\mathbf{Y})} \right]. \tag{4.8}$$

The coefficient γ is lower when the difference of active pixels in the input matrices is higher

$$\gamma(\mathbf{X},\mathbf{Y}) = \begin{cases} 1 - \log_n\left(\dfrac{\Sigma(\mathbf{X})}{\Sigma(\mathbf{Y})}\right) & \text{if } \Sigma(\mathbf{X}) \geqslant \Sigma(\mathbf{Y}), \\ 1 + \log_n\left(\dfrac{\Sigma(\mathbf{X})}{\Sigma(\mathbf{Y})}\right) & \text{otherwise.} \end{cases} \tag{4.9}$$

With this choice of γ, for large differences in the number of active pixels between the two initial matrices, the coefficient quickly tends to a low value, thus reducing $\alpha(\mathbf{X},\mathbf{Y})$.

Hence, the resulting algorithm works as follows.

Algorithm Matching index computation

Input: Boolean images $\mathbf{A}$ and $\mathbf{B}$.
Parameters: Number of steps K, positive constants $\lambda < 1$ and μ, integer neighbourhood amplitude $\delta > 0$ (which implies the size of the set $\mathcal{N}$).
Outputs: Matching index $\alpha(\mathbf{A},\mathbf{B})$.

1. Set the initial conditions converting the two input images from Boolean into real matrices $\mathbf{X}(0) := \mathbf{A}$ and $\mathbf{Y}(0) := \mathbf{B}$.
2. **for** $k = 0, k < K, k = k + 1$
   compute the updated values for each pixel in both images according to (4.4)–(4.5)

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)$$

$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)$$

   Set $\mathbf{X}(k) := \mathbf{X}(k) \odot \mathbf{A}$, $\mathbf{Y}(k) := \mathbf{Y}(k) \odot B$, where $\odot$ denotes the element-wise (Hadamard) product
   end for
3. Convert the matrices back to Boolean by means of the operator defined in (4.6):
   for each point (i,j) **do**

$$\tilde{A}_{i,j} = \Theta\left(X_{i,j,K}\right)$$

$$\tilde{B}_{i,j} = \Theta\left(Y_{i,j,K}\right)$$

   end for
4. Compute the matching index α as in (4.8)

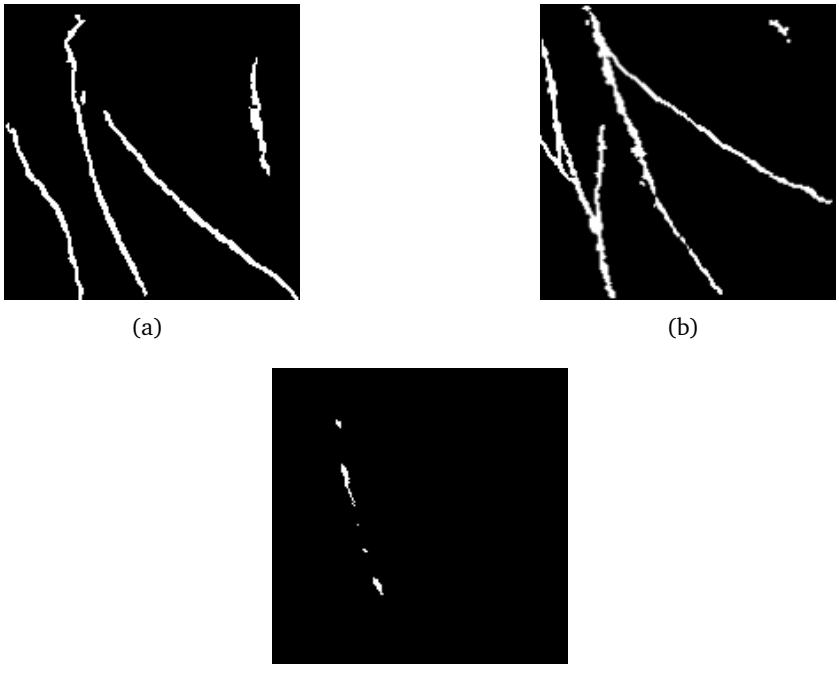
$$\alpha(\mathbf{A},\mathbf{B}) = \frac{\gamma(\mathbf{A},\mathbf{B})}{2}\left[\frac{\Sigma(\tilde{\mathbf{A}})}{\Sigma(\mathbf{A})} + \frac{\Sigma(\tilde{\mathbf{B}})}{\Sigma(\mathbf{B})}\right].$$

Note that the images are processed as real-valued matrices, in view of step 1. Due to the element-wise (Hadamard) product at step 2, all the pixels that are initially zero remain zero: the procedure can "inactivate" active pixels, but cannot "activate" inactive pixels. As a consequence, the index $\alpha \in \mathbb{R}$ is always a number $0 \leqslant \alpha \leqslant 1$.

The rationale of step 2 (the core of the procedure) is the following.
Let $\mathbf{X}, \mathbf{Y} \in \{0,1\}^{n \times n}$ be the matrices to be compared, and assume that a pixel in a matrix has no match on the other one (or equivalently, the cross-matching part is absent). Then, the corresponding equations become

$$X_{i,j}(k+1) = \lambda X_{i,j}(k)$$
$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k)$$

Hence, being $\lambda < 1$, the pixel values having no match on the other image, should asymptotically converge to zero: typically, in few iterations $X_{i,j}(k)$ and $Y_{i,j}(k)$ reach a value $\epsilon \ll 1$. Again, note that if a pixel has no match at the beginning, this status is preserved for all the steps of the procedure, due to the element-wise product at step 2. Conversely, the value of pixels having a large neighbourhood in the comparison image diverges. If the same condition is true in the complementary neighbourhood, then we are in the presence of two matching groups that "cooperate" by increasing each other. In this case, the pixel values represented by $X_{i,j}$ and $Y_{i,j}$ significantly increase in few iterations. The exponential convergence/divergence of the procedure assures that, in few iterations, some pixels become practically zero (eventually set to 0), and others achieve large values (eventually set to 1). To exemplify the algorithm behaviour, suppose that the two input images $\mathbf{A}, \mathbf{B} \in \{0,1\}^{n \times n}$ are those shown in Figures 4.2(a) and 4.2(b). Then, the result of the dynamic algorithm, $\tilde{\mathbf{A}} = \tilde{\mathbf{B}} \in \{0,1\}^{n \times n}$, is shown in Figure 4.2(c). In both images, the algorithm has iteratively eliminated points that do not have enough close neighbouring points in the comparison image. Hence, the number of survived pixels is very limited, even when starting from images having similar line-like features.



(a) (b)

(c)

Figure 4.2: Dynamic algorithm behaviour: (a), (b) similar line-like features provided as input matrices, and (c) the resulting image from the application of the dynamic algorithm.

4.1.1 Properties of the dynamical system

In this section has been analysed the algorithm based on the linear recursive Equations (4.4) and (4.5), to better understand its behaviour.
First of all, the positive parameters λ and μ in the procedure need to be tuned bearing in mind that:

- the *fading parameter* $\lambda < 1$ cancels "isolated pixels";
- the *emphasis parameter* μ increases "surrounded pixels".

Convergence conditions. *Let* $\mathbf{A}, \mathbf{B} \in \{0,1\}^{n \times n}$ *be two Boolean matrices to be compared. Consider a radius* $\delta \in \mathbb{N}$ *of a generic neighbourhood* $\mathcal{N}$ *whose cardinality is given by* $m = n(\mathcal{N}) = (2\delta+1)^2$*, and assume that all the points in* $\mathcal{N}$ *are initially active (set to 1) in both* $\mathbf{A}$ *and* $\mathbf{B}$*. Then, their values should "blow up". Given the updating equation for* $X_{i,j}$*,*

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)$$

and being all the pixels initially set to 1, a blow-up condition is

$$\lambda + \mu m > 1. \tag{4.10}$$

Let us now consider a partial superposition, in which only a percentage $b < 1$ *of the pixels in the neighbourhood are 1. It may be taken* b *as a threshold fraction, so that 1-pixels with less than a fraction* b *of 1-pixels in the complementary neighbourhood tend to fade. This leads to the criterion*

$$\lambda + b\mu m < 1. \tag{4.11}$$

From the considerations above, at step $k = 1$, all the surrounded pixels reach a value greater than 1; at step $k = 2$, the value becomes even bigger; and so on. The two constraints determined by Equations (4.10) and (4.11) represent the lower and upper bounds in Figure 4.3.

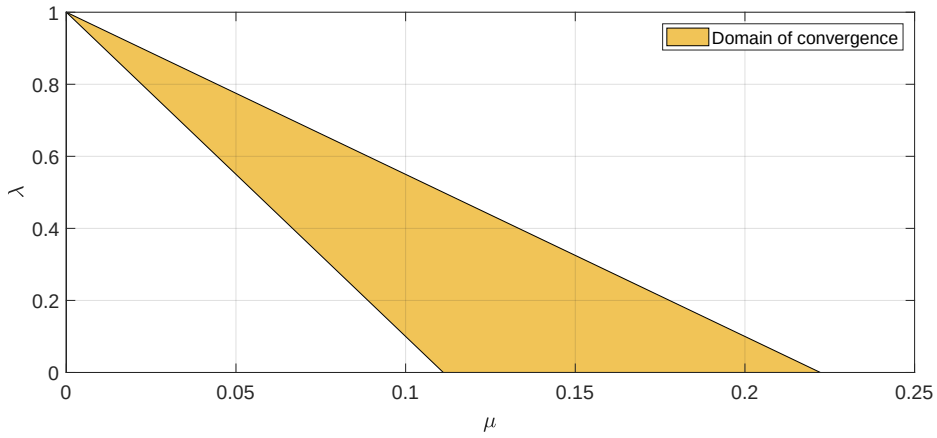


Figure 4.3: Example of convergence domain delimited by the Equations (4.10) and (4.11) with $\delta = 1$, which implies $m = 9$, and $b = \frac{1}{2}$.

Therefore, given the neighbourhood radius δ and the number of iterations K, a criterion to choose the parameters λ and μ is the following:

1. fix λ based on the desired fading speed;
2. compute m, the neighbourhood size, in terms of pixels;
3. fix b, the desired threshold fraction;
4. compute μ which satisfies the constraints (4.10) and (4.11).

Property 1 (Decoupling)

Let $\mathbf{A}, \mathbf{B} \in \{0,1\}^{n \times n}$ be two Boolean matrices to be compared. Then, different regions that are not connected in either $\mathbf{A}$ or $\mathbf{B}$ do not interact and have separate evolutions, as illustrated in Figure 4.4. Hence, isolated spots and blobs[3] can be analysed separately.

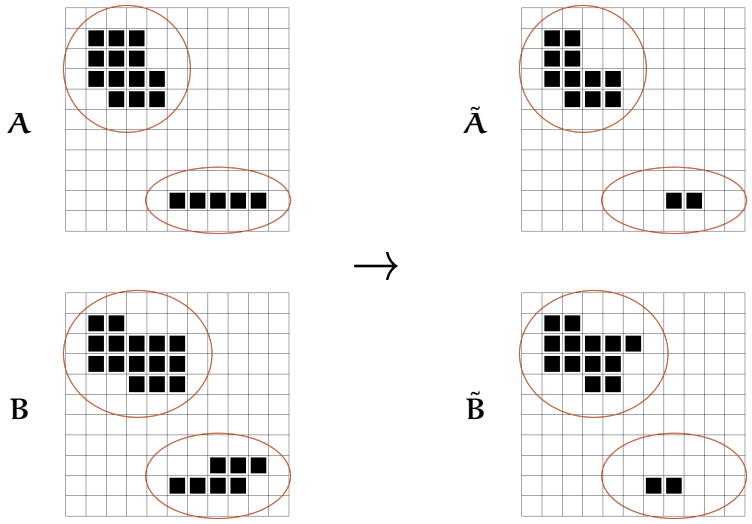


Figure 4.4: Example of two sets of pixels with separate evolutions.

Property 2 (Convergence/Divergence)

Let $\mathbf{A}, \mathbf{B} \in \{0,1\}^{n \times n}$ be two Boolean matrices to be compared. Then, two overlapping clusters in $\mathbf{A}$ (containing q pixels) and in $\mathbf{B}$ (containing s pixels) give rise to a positive linear system, as described next.
Let us group in a vector $\mathbf{x}(k) \in \mathbb{R}^q$ the variables $X_{i,j}$ of the cluster in $\mathbf{A}$ and in a vector $\mathbf{y}(k) \in \mathbb{R}^s$ the variables $Y_{i,j}$ of the cluster in $\mathbf{B}$. Then, the system evolves as follows:

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{y}(k+1) \end{bmatrix} = \begin{bmatrix} \lambda \mathbf{I} & \mu \mathbf{\Omega} \\ \mu \mathbf{\Omega}^\top & \lambda \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix} \tag{4.12}$$

where $\mathbf{I}$ is the identity matrix (of the proper dimension), λ and μ are the previously fixed constants and $\mathbf{\Omega}$ is a connection matrix defined as:

$$\Omega_{i,j} = \begin{cases} 1 & \text{if pixel } y_j \text{ is in the neighbourhood of pixel } x_i, \\ 0 & \text{otherwise.} \end{cases} \tag{4.13}$$

[3] A blob is a subset of an image in which some properties are constant or approximately constant.

Note that the non-negative matrix $\mathbf{M} \succeq \mathbf{O}$ appearing in Equation (4.12) is positive semidefinite and symmetric

$$\mathbf{M} = \begin{bmatrix} \lambda \mathbf{I} & \mu \boldsymbol{\Omega} \\ \mu \boldsymbol{\Omega}^{\top} & \lambda \mathbf{I} \end{bmatrix}.$$

In fact, if pixel y_j is in the neighbourhood of pixel x_i, also x_i is in the neighbourhood of y_j. Furthermore, since the matrix $\mathbf{M}$ has non-negative off-diagonal entries $M_{i,j} \geqslant 0$ $(\forall i \neq j)$, it is also a Metzler matrix (i.e., $\mathbf{M} \in \mathbb{M}$). Then, the fading or the permanence of the two connected spots is doomed by the convergence or divergence of the positive system (4.12). The linear system converges (i.e., is stable) if and only if the spectral radius ρ (i.e., the maximum among all the absolute values of each eigenvalue σ_i) of the Metzler matrix $\mathbf{M}$ is

$$\rho(\mathbf{M}) = \max\left\{|\sigma_i|\right\} = \sigma_{\max} < 1, \quad \forall i. \tag{4.14}$$

Conversely, the linear system diverges (i.e., is unstable) if and only if

$$\rho(\mathbf{M}) = \max\left\{|\sigma_i|\right\} = \sigma_{\max} > 1, \quad \forall i. \tag{4.15}$$

The significance of the expression is the following. The eigenvalues of matrix $\mathbf{M}$ are of the form $\sigma_k = \lambda + \mu\sigma_k^0$, where σ_k^0 denote the eigenvalues of the cross-term matrix:

$$\begin{bmatrix} 0 & \Omega \\ \Omega^{\top} & 0 \end{bmatrix}.$$

Then, the dominant (Frobenius) eigenvalue is $\sigma_{\max} = \lambda + \mu\sigma_{\max}^0$. The term $\mu\sigma_{\max}^0$ is associated with the cross-term element size which amplify the variables x and y, while λ is associated with the local decay.

4.2 An enhanced version of the dynamical system

The proposed enhancement consists of a slightly modified version of the previously described positive dynamical system. Given two initial Boolean images $\mathbf{X}, \mathbf{Y} \in \{0,1\}^{n \times n}$ of the same size, the recursive algorithm yields a pair of real matrices, $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$, with the same size as $\mathbf{X}$ and $\mathbf{Y}$. The algorithm is initialised as $\tilde{X}_{i,j}(0) = X_{i,j}$ and $\tilde{Y}_{i,j}(0) = Y_{i,j}$ where $X_{i,j}$ and $Y_{i,j}$ are the binary values of pixel i, j in the original images $\mathbf{X}$ and $\mathbf{Y}$, converted to real (floating point) values. The dynamical system also includes a term that *initially* increases the value of a pixel if the pixels in a proper neighbourhood *in the same image* have large values, and then vanishes with time; it has the effect of initially thickening the relevant patterns. The two images are processed according to the iterations

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \underbrace{\mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)}_{\text{cross-matching}} + \underbrace{\nu^k \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)}_{\text{initial expansion}} \tag{4.16}$$

$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k) + \underbrace{\mu \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)}_{\text{cross-matching}} + \underbrace{\nu^k \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)}_{\text{initial expansion}} \tag{4.17}$$

where $k = 0, 1, \ldots, K-1$, and $\mathcal{N}_{i,j}$ is a square neighbourhood of the pixel i, j of integer amplitude $\delta > 0$ as described in (4.3).

Convergence conditions. *Given the size* δ *of the neighbourhood* $\mathcal{N}_{i,j}$ *and denoting by* $m = n(\mathcal{N}) = (2\delta+1)^2$ *the corresponding number of pixels, the positive parameters* λ, μ, ν *are selected based on the following constraints, which limit the search region for the optimisation procedure within a bounded set:*

$$\begin{cases} 0 < \lambda, \mu, \nu < 1 & (4.18) \\ \lambda + \nu < 1 & (4.19) \\ b\mu m < 1 - \lambda < \mu m & (4.20) \end{cases}$$

Since we consider positive parameters, the requirement that $\lambda, \mu, \nu < 1$ *in* (4.18) *is implied by* (4.19) *and* (4.20).

From the considerations above, the convergence conditions determined by Equations (4.18)–(4.20) represent the bounded set illustrated in Figure 4.5.

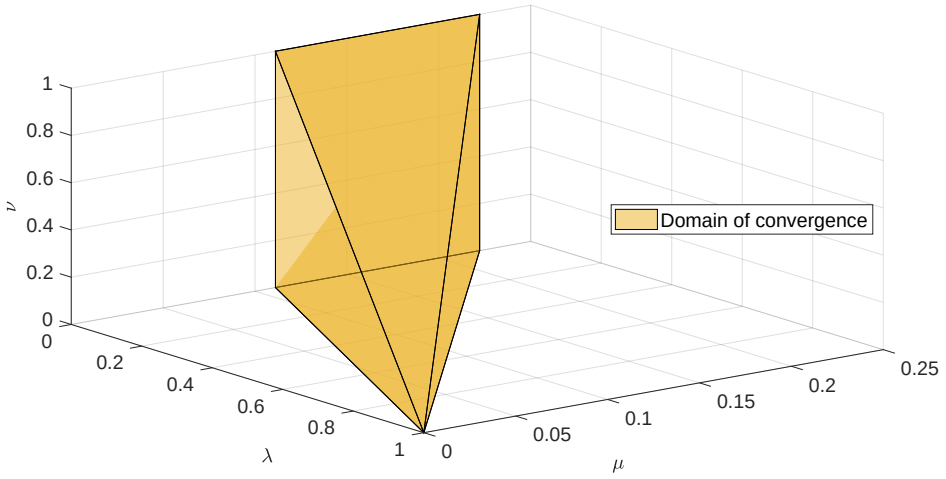


Figure 4.5: Example of domain of convergence of the system bounded by the constraints reported in (4.18)–(4.20) with $\delta = 1$, which implies $m = 9$, and $b = \frac{1}{2}$.

The reasoning behind the introduced constraints can be explained as follows.

- Given (4.16) and (4.17) in the absence of cross-matching ($\mu = 0$) and of initial expansion terms ($\nu = 0$), $X_{i,j}(k)$ and $Y_{i,j}(k)$ should asymptotically converge to zero.

  Proof. Given the following dynamical system

  $$X_{i,j}(k+1) = \lambda X_{i,j}(k)$$

  $$Y_{i,j}(k+1) = \lambda Y_{i,j}(k)$$

  then, the inequality the $\lambda < 1$ easily proves that $X_{i,j}(k)$ and $Y_{i,j}(k)$ asymptotically converge to zero if and only if the constraint is satisifed. □

- The expansion term initially augments $X_{i,j}$ (resp. $Y_{i,j}$) if the average value of the pixels in the neighbourhood of $X_{i,j}$ (resp. $Y_{i,j}$) is large. This is due to the factor ν^k, where the parameter ν has to satisfy the constraint (4.18), $0 < \nu < 1$, so that this initial effect quickly vanishes with time. This is needed because, in the long run, the persistence of this term would make all lines thicker and thicker, leading to false positives.

- The inequality in (4.19) needs to hold to ensure that, if there is an isolated pixel $X_{i,j}$ whose value is initially large, but there are no active pixels in both the neighbourhoods of the image itself and the complementary image, then the pixel value decreases from the very beginning of the iterations.

  Proof. Let $X_{i,j}$ be an isolated pixel with a high value which does not have active pixels in both the neighbourhoods of the image itself and the complementary image. Then, the updating equation for that point becomes

  $$X_{i,j}(k+1) = \left(\lambda + \nu^k\right) X_{i,j}(k).$$

  The inequality $\nu + \lambda < 1$ easily proves that such a pixel is therefore quickly cancelled. □

- The cross-matching term increases $X_{i,j}$ (resp. $Y_{i,j}$) if the average value of the pixels in the complementary neighbourhood is large.

  Proof. If $\nu^k \approx 0$, which is ensured for large k in view of (4.18), the updating equation becomes

  $$X_{i,j}(k+1) = \lambda X_{i,j}(k) \; + \; \mu n(\mathcal{N}) \, \frac{\overbrace{\sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)}^{\text{average value}}}{n(\mathcal{N})}. \tag{4.21}$$

  Hence, the first inequality in (4.20), $b\mu n(\mathcal{N}) + \lambda < 1$ implies that, if the average value in the complementary neighbourhood is around or below a fraction b of $X_{i,j}(k)$, then $X_{i,j}$ gets smaller:

  $$X_{i,j}(k+1) < X_{i,j}(k).$$

  Conversely, the second inequality in (4.20), $\mu n(\mathcal{N}) + \lambda > 1$, means that if the average value is about or greater than $X_{i,j}$, the complementary region should be consider as "populated" so $X_{i,j}$ should get larger:

  $$X_{i,j}(k+1) < X_{i,j}(k).$$

  The same holds for $Y_{i,j}(k)$. □

The resulting algorithm works as follows.

Algorithm Matching index computation

Input: Pair of Boolean images $\boldsymbol{A}, \boldsymbol{B} \in \{0,1\}^{n \times n}$.
Parameters: Number of steps K, positive constants $\lambda, \mu, \nu < 1$, integer radius $\delta > 0$.
Outputs: Matching index $\alpha(\boldsymbol{A}, \boldsymbol{B})$.

1. Set the initial conditions $\boldsymbol{X}(0) := \boldsymbol{A}$ and $\boldsymbol{Y}(0) := \boldsymbol{B}$, $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n \times n}$.
2. **for** $k = 0, k < K, k = k + 1$
   update pixel values in both images according to (4.16)–(4.17)
   Set $\boldsymbol{X}(k) := \boldsymbol{X}(k) \odot \boldsymbol{A}$, $\boldsymbol{Y}(k) := \boldsymbol{Y}(k) \odot B$ (Hadamard product)
   end for
3. Convert the matrices back to Boolean by means of the operator defined in (4.6)
4. Compute the matching index α as in (4.8)

At the end of the iterations, given $X_{i,j}(K)$ and $Y_{i,j}(K)$, the images $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ are produced, where some pixels are set to 0 and others to 1 according to the Boolean decision boundary in (4.6). After that, the matching index α is computed as in (4.8). Similar images will have a large number of surviving pixels, hence a large matching index α, whilst different images will be left with a very small number of non-zero pixels, with α considerably small. Since $\nu^k \to 0$ when $k \to \infty$, asymptotically the recursion becomes identical to that in the dynamical system (4.4)–(4.5) achieving the same steady-state condition.
Therefore, it can be formally guaranteed that asymptotically the performance is at least as good as with the early version of the system. Moreover, the enhanced dynamic algorithm has noteworthy advantages.

- The term ν^k provides an initial burst that considerably increases the speed of convergence. Its effect is that of initially enlarging the lines, which is beneficial and very rapidly leads to a situation where the discrimination is possible.
- The benefits of the joint cross-matching and the initial expansion term are seen in the first iterations, which allow to stop the algorithm at an early stage. Indeed, the best results in terms of discrimination are achieved after few iterations, so waiting any longer is useless (although it does not compromise performance in terms of the matching index). Stopping the algorithm after few iterations drops the computational time of an order of magnitude and still allows for very effective line-like feature matching.

4.3 A remarkable example of noise robustness

Noise can be systematically introduced into digital images during image acquisition and/or transmission [55]. An essential issue related to digital image processing is to effectively reduce noise from an image whilst keeping its features intact. The nature of the problem depends on the type of noise affecting the image. Mathematically, noise can be classified into three kinds of models which can adequately represent most image noises:

- *additive* (signal independent), described as

$$\mathbf{T}(i,j) = \mathbf{O}(i,j) + \mathbf{N}(i,j) \tag{4.22}$$

- *multiplicative* (signal dependent), described as

$$\mathbf{T}(i,j) = \mathbf{O}(i,j) \cdot \mathbf{N}(i,j) \tag{4.23}$$

- impulsive (signal independent and uncorrelated).

In particular, the most frequent image noises are: additive Gaussian noise (introduced during image acquisition) and impulse noise (introduced during image transmission) [51]. The second kind of noise raises more concerns for line-like feature matching purposes, as the noise arisen during acquisition should be removed in the preprocessing stage.

Assumption 3. *Throughout this chapter, it has been assumed that noise is independent of spatial coordinates, and that it is uncorrelated with respect to the image itself (that is, there is no correlation between the values of noise components and pixel values).*

Based on the assumptions in the previous section, the spatial noise descriptor with which we shall be concerned is the statistical behaviour of the intensity values in the noise component of the impulse noise model characterised by a specific probability density function (PDF) [120].

Definition 12. *The impulse noise model is always independent, randomly distributed, and uncorrelated with the images. This is also called data drop noise because statistically its drop the original data values.*

The probability density function is given by

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise.} \end{cases} \tag{4.24}$$

If $b > a$*, the intensity level* b *will appear as a bright spot in the image. Conversely, the intensity level* a *will appear as a dead spot. If either* P_a *or* P_b *is zero, the impulse noise is called unipolar. If neither probability is zero, and especially if they are approximately equal, impulse noise values will resemble salt-and-pepper grains randomly distributed over the image.*

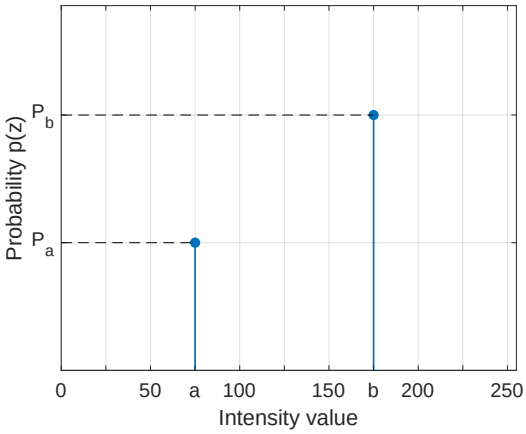
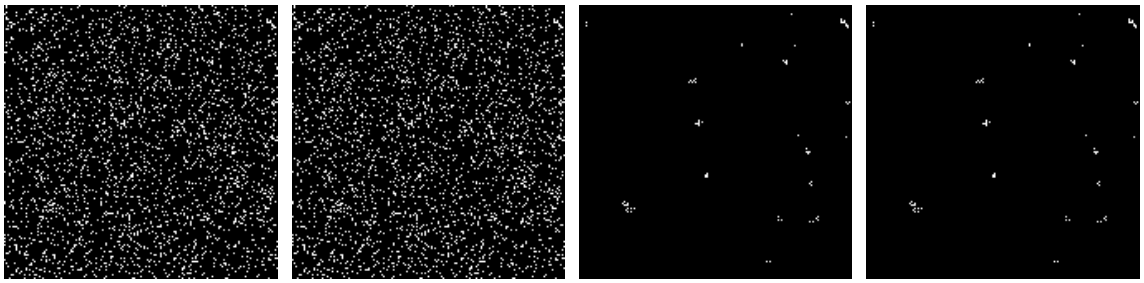


Figure 4.6: PDF of the noise.

A considerable advantage of the proposed unified method for line-like feature matching is its robustness with respect to such a noise. Indeed, images corrupted by noise (e.g., salt-and-pepper) are easily recognised, whilst an image randomly generated is rejected even if compared with itself.



(a) (b) (c) (d)

Figure 4.7: Example of salt-and-pepper noisy image comparison: (a)–(b) matrices $\mathbf{A}$ and $\mathbf{B}$ containing the same impulse noise ($\mathbf{A} = \mathbf{B}$) with probability $p = 20\%$, and (c)–(d) the resulting matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ ($\tilde{\mathbf{A}} = \tilde{\mathbf{B}}$) from the application of the dynamic algorithm.

Example 1. Salt-and-pepper noise. *To explain a peculiar feature of the system, it has been compared a salt-and-pepper noisy image with itself (see Figure 4.7). The algorithm eliminates almost all of the pixels and the resulting* matching index *is extremely low.*

Definition 13. *Let* U *and* V *be two disjoint sets, then a bipartite graph is defined as* $G = \langle U, V, E \rangle$ *where* E *connects every node in* U *with all nodes in* V.

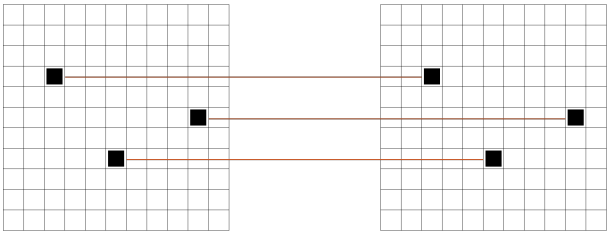
Let us now consider a family of bipartite graphs, whose nodes are the active pixels in the two matrices $\mathbf{A}$ and $\mathbf{B}$ to be compared. An arc connects each pair of active pixels that occupy neighbouring positions in the two images. In this way, clusters of active pixels are formed.

A cluster is preserved only if it includes enough pixels, otherwise it fades away. In the case of perfect matching of two images with isolated points (as in Figure 4.8(a)), there are just isolated pairs of nodes, which disappear when the dynamical system evolves (local stability and convergence to 0). Conversely, in the presence of partially matching blobs (as in Figure 4.8(b)), the clusters of nodes are numerous enough to survive (local instability with saturation to 1). Furthermore, as can be easily seen from the matrix form of the dynamical system

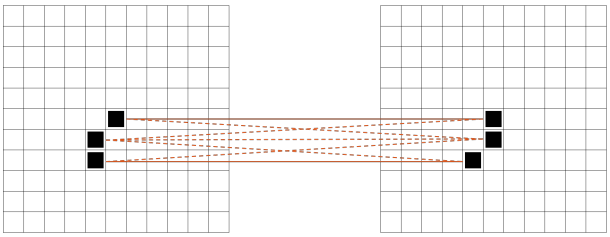
$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{y}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} (\lambda+\nu^k)\,\mathbf{I} & \mu\boldsymbol{\Omega} \\ \mu\boldsymbol{\Omega}^\top & (\lambda+\nu^k)\,\mathbf{I} \end{bmatrix}}_{\text{matrix } \mathbf{M}} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix} \tag{4.25}$$

it presents a non-negative matrix $\mathbf{M}$ with a dominant eigenvalue σ_{max} that is real and positive. Moreover, σ_{max} is an increasing function of parameters λ, ν, μ, and increases if the number of 1 entries in matrix $\boldsymbol{\Omega}$ increases. This explains why the pair of identical images containing only a reasonable amount of noise leads to a poor matching index: even if there is actually a good (or perfect) matching between the two images, there are not enough connections between the active pixels of one image and the active pixels of the other.
Finallly, to discriminate whether the complementary neighbourhood has to be considered "populated" or "unpopulated", it is reccommended to choose the empirical threshold fraction $b \approx 1/2$ since in the average, half of the pixels in the complementary neighbourhood are inactive. Then, in view of (4.10) and (4.11), this leads to the conditions:

$$\frac{1}{2} < \frac{1-\lambda}{\mu m} < 1. \tag{4.26}$$



(a)



(b)

Figure 4.8: Dynamic algorithm example: (a) pairs of isolated points and (b) clusters of points. Points in the same positions are connected through solid lines, whereas points in different positions but close enough (radius $\nu = 1$) are connected through dashed lines.

4.4 Example I: a robust palmprint-based biometric recognition system for personal identification

Palmprint recognition, a relatively novel but promising biometric technology, has recently received considerable interest, mostly for its importance in forensics[4] [24] and for several potential civil applications [80]. Compared with the other physical biometric characteristics, palmprint-based personal authentication has several advantages: low-resolution imaging, low-intrusiveness, stable line features, and low-cost capturing device. Indeed, since the principal lines and wrinkles can be observed under low-resolution images (e.g., 100 dpi or lower), palmprint systems do not require high resolution capturing devices [105]. However, palmprint recognition techniques based on both low- and high-resolution features are proposed in the literature; such methods use as local features, respectively, principal lines and ridges (see Figure 4.9).

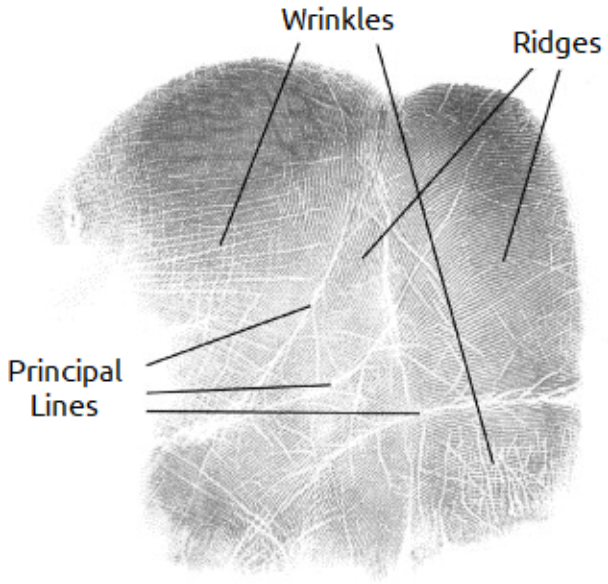


Figure 4.9: The distinctive features of a palmprint: principal lines, wrinkles, and ridges. The minutiae are the points where the ridges changed such as bifurcation and endpoint.

The approach discussed in this thesis is based on low-resolution features, since it uses as local features the principal lines of the palm, which are very important physiological characteristics to distinguish between different individuals because of their stability and uniqueness. Compared with ridges and wrinkles, principal lines are usually the consequence of genetic effects: therefore, they are the most significant features in palmprint images and have good permanence [20, 90]. However, principal lines may be similar in different individuals, which makes their distinctiveness relatively low; for this reason, palmprint recognition is a challenging problem [155].

[4]The evidential value of palmprints in forensics is clear as about 30% of the latents recovered from crime scenes are from palms [70].

To address this challenge, it has been proposed a method consisting of the following: (i) a ROI extraction phase, which follows the sequence of steps used in [140, 155], to face different issues mainly due to non-linear distortion, such as rotation and translation of the palm with respect to the image and non-uniform illumination; (ii) an unconventional feature extraction phase based on the principal lines of the palm; (iii) a novel approach to palmprint matching based on a dynamic algorithm as described in Section 4.1. The results obtained by means of this novel algorithm are promising, and comparable to existing state-of-the-art palmprint recognition algorithms.

4.4.1 Related work

Most authors focus on palm lines when analysing low-resolution palmprint images, in particular, in line-based approaches, the extracted palm lines are either matched directly or represented in other formats through some feature transformations for matching. Zhang and Shu [154] used the datum point invariant property and the line feature matching technique to conduct the verification process via the palmprint features. This approach, in which the authors inked the palmprints on to paper and then scanned them to obtain images, is not suitable for many on-line security systems. Wu *et al.* [149] used the Canny's operator [23] to detect palm lines and the orientations of the edge points are passed into four membership functions representing four directions. For each direction the energy is compute and normalised. Finally, Euclidean distance is used for matching. Another study by Wu *et al.* [150] also proposed an algorithm based on the hidden Markov model (HMM). This approach uses Sobel masks to compute the magnitude of palm lines, then the projections along both x and y directions are used to obtain the inputs of the HMMs. A third approach proposed by Wu *et al.* [148] is based on the design of two masks which compute the vertical first- and second-order derivatives obtained by rotating the two standard masks. They identify the edge points and corresponding directions by using the zero-crossing of the first-order derivative, then the modulus of the corresponding second-order derivative is considered as the magnitude of the lines. The feature vector, consisting of the weighted sum of the local directional magnitude, is then normalised. Finally, as in [149], Euclidean distance is used for matching. Boles *et al.* [16] used Sobel masks and thresholds to build binary edge images and then the Hough transform to detect the palm features as approximated straight lines for matching. Huang *et al.* [66] and Jia *et al.* [74] used the modified finite Radon transform (MFRAT) to extract the principal lines, then a matching algorithm based on pixel-to-area comparison measures the similarity between two palmprint images. Fei *et al.* [46] used a double half-orientation-based method for feature extraction and palmprint recognition, whilst in [45] they proposed a double-orientation code (DOC) algorithm to represent the orientation of palmprint features, evaluating the similarity between the DOC through a non-linear angular matching scheme. Gao and Leung [50] and Leung *et al.* [91] made use of Sobel masks to extract line-like features from the palmprints and then of line segment Hausdorff distance to compare two palmprints. Han *et al.* [58] employed Sobel and morphological operations to extract line-like features from palmprint images, and then a correlation function to measure the similarity between the two feature vectors. Similarly, for verification, Kumar *et al.* [86] used other directional masks to extract line-like features from the palmprints captured using a digital camera, then a normalised correlation for matching is used. Diaz *et al.* [37] used Sobel masks and morphologic operator as two separated feature extractors to obtain the gradient of the images, which are then classified by neural networks. Kung *et al.* [88] designed

a decision-based neural network (DBNN) classifier and applied it to face recognition and to palmprint verification. Other approaches based on low-resolution features are named subspace-based approaches, which usally adopt principal component analysis (PCA), linear discriminant analysis (LDA), and independent component analysis (ICA). Zhang *et al.* [155] used 2D Gabor filters to extract texture features from low-resolution palmprint images captured using a charge-coupled device (CCD) camera: two palmprint images are compared in terms of their Hamming distance. Connie *et al.* [29] made use of PCA and ICA to extract the palmprint texture features, then palmprints are compared based on three types of classifiers: Euclidean distance, cosine measure, and probabilistic neural network (PNN) methods. Ekinci and Aykut [38] proposed a wavelet-based kernel PCA method by integrating the Daubechies wavelet representation of palm images and the kernel PCA method for palmprint recognition; similarity measurement is accomplished by using a weighted Euclidean linear distance-based nearest neighbour (NN) classifier. Raghavendra *et al.* [124] proposed the use of kernel direct discriminant analysis (KDDA) as final classification on texture features obtained using log Gabor transform and concatenated to form a fused feature vector after the use of particle swarm optimisation (PSO) to reduce the dimension of this vector. Kumar and Zhang [85] made use of the product of sum rule on texture-, line-, and appearance-based features using Gabor filters, line detectors, and PCA respectively. The approach proposed by Wincy and Chandran [146] used phase-correlation function (PCF) and speeded up robust features (SURF) for palmprint identification. The PCF is a matching technique that uses the phase components in 2D discrete Fourier transforms (DFTs) of given palmprint images, whilst the SURF is applied to extract the key points by using a Hessian detector, then an orientation invariant descriptor is constructed for each key point. Krishneswari *et al.* [83] proposed instead a PSO-based fusion technique to combine the two Gabor and line feature vectors into a single feature vector without concatenation, then the resulting vector is reduced using PCA and finally the NN classifier is applied.

4.4.2 Hand palm image processing

The overall block diagram of palmprint image processing is shown in the figure below.

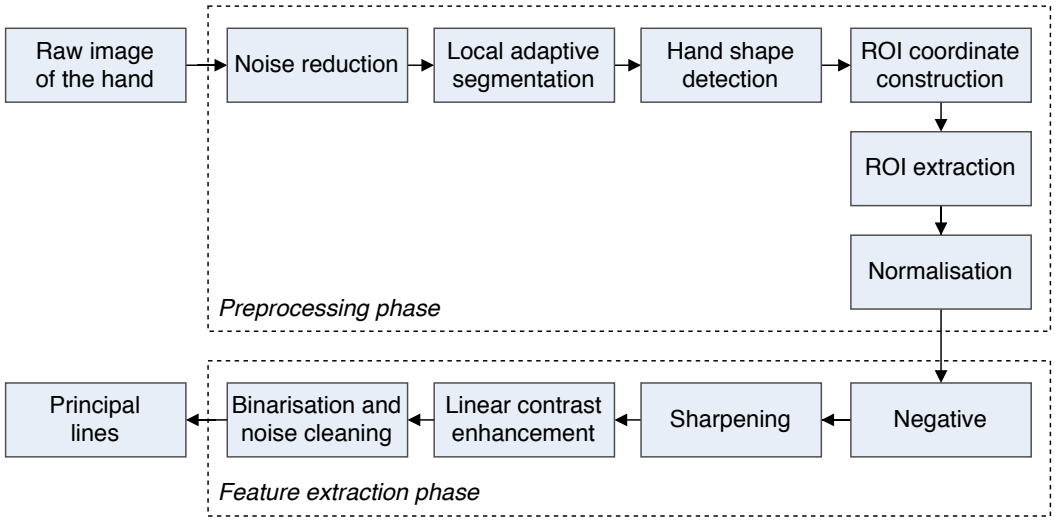


Figure 4.10: Block diagram of the hand palm image processing for palmprint recognition.

Preprocessing

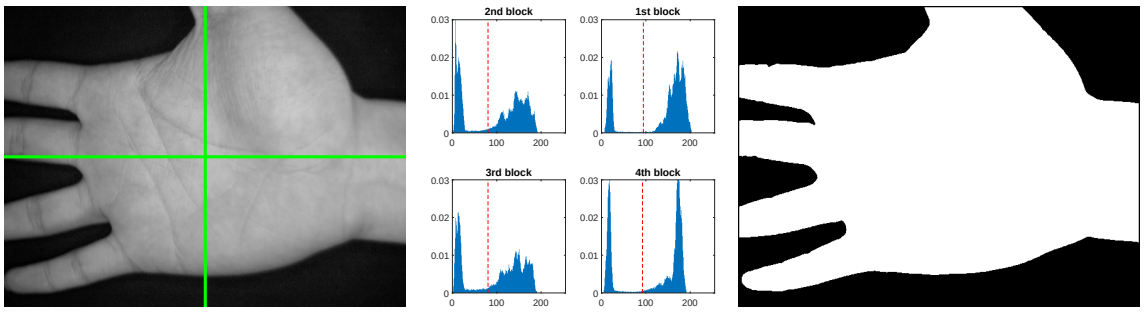
Before the feature extraction step, a preprocessing elaboration is required to extract the central ROI from the captured palmprint image. In fact, usually palmprint images can have different orientation and size, and are also subject to noise. Moreover, the region of not-interest (e.g., fingers, wrist, and image background) may affect the accuracy in processing and verification performance. As outlined in the Figure 4.10, the major steps involved in the preprocessing of raw images are: 1. noise reduction by means of proper filter, 2. local adaptive segmentation, 3. hand shape detection, 4. ROI coordinate construction and extraction, and 5. normalisation to have a specific mean and variance for all images [117].

Noise reduction A noise typically corrupting digital images is the impulse noise [51, 96]. This kind of noise is independent, randomly distributed, and uncorrelated with the image, since it can affect all pixels in the image with the same probability. Hence, a common non-linear spatial filter, i.e., median filter, can be used to remove unwanted information from noisy palmprint images preserving details. In our experiments, the kernel size has been set to 5×5 pixels.

Segmentation This step consists of a local adaptive clustering-based image thresholding used to reduce the input grey level palmprint image $I(x, y)$ to a binary image $B(x, y)$, where the hand is the foreground and the rest is the background. After partitioning the image $I(x, y)$ into non-overlapping sub-blocks of size $N \times N$, for each sub-block a local thresholding can be applied by using Otsu's method [116] (see Appendix A.1 for further details): the image pixels are split into two classes (background and object), and the best threshold value t is obtained based on the maximum value of the variance between the two classes [42]. Then, for each sub-block, a global thresholding has been applied as follows:

$$B_{x,y} = \begin{cases} 255 & \text{if} \quad I_{x,y} \geqslant t, \\ 0 & \text{otherwise}, \end{cases} \tag{4.27}$$

so that a pixel is shown in white when it has a value greater than the threshold, in black otherwise. Figure 4.11 illustrates the result of this operation.



(a) (b) (c)

Figure 4.11: Local adaptive binarisation: (a) original palmprint image divided in four blocks, (b) the statistical relationship between the grey levels of each sub-image and the frequency, and (c) output of the binarisation step.

Hand shape detection In order to detect the edge of the hand, the segmented binary image is filtered by using the Canny's operator [23]. This edge detector ensures good noise immunity and at the same time detect true edge points with minimum error [42]. A practical approximation of the Canny filter is the following (see Appendix A.2 for further details).

- Convolve the input image $I(x, y)$ with a two-dimensional Gaussian filter $G(x, y)$:

$$H(x, y) = (I * G)(x, y) \tag{4.28}$$

  where the Gaussian function is defined as

$$G(x, y) = c \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}\right) \tag{4.29}$$

  with $c = 1$, $x_0 = 0$, and $y_0 = 0$.
  In order to reduce the computational cost, a convolution with a two-dimensional Gaussian can be separated into two convolutions with one-dimensional Gaussians.
- Apply the gradient operator and calculate the gradient magnitude M and orientation Θ as follows:

$$H_x(x, y) = \frac{\partial G(x, y)}{\partial x} \qquad H_y(x, y) = \frac{\partial G(x, y)}{\partial y} \tag{4.30}$$

$$m(x, y) = \sqrt{H_x^2(x, y) + H_y^2(x, y)} \tag{4.31}$$

$$\Theta(x, y) = \arctan\left(\frac{M_y(x, y)}{M_x(x, y)}\right). \tag{4.32}$$

- Apply the non-maximum suppression algorithm that removes the non-maximum pixels, while preserving the connectivity of the contours, as follows: for each pixel (x, y), if one of the neighbouring pixels in the orthogonal directions to edge orientation is greater than the current pixel, then discard it.
- The last step consists of the hysteresis thresholding, which uses a high threshold t_H and a low threshold t_L such that: if any pixel (x, y) in the image has a value greater than t_H (less than t_L) is presumed (is not presumed) to be an edge pixel, and is marked as such immediately. Then, any pixel that is connected to this edge pixel and that has a value greater than t_L is also selected as edge pixel, and is marked too. The marking of neighbours can be done recursively, as it is in the function hysteresis, or by performing multiple passes through the image. The relationship between the two threshold values is $t_L = t_H/2$ where t_H is calculated with the Otsu's algorithm [42].

ROI extraction In a palmprint image, palm location and rotation angle will affect the ROI segmentation and the feature extraction of the palmprint may be compromised. To reduce the influence of non-linear factors (e.g., non-uniform illumination) as well as affine transformations, such as rotation, translation, and homothety[5] of the palm (see Appendix A.5), a standard reference system is used to align all the palm images in a standard pose, hence, it is possible to locate the peak and valley points of the palm tracking the distance between the barycentric coordinates of the segmented hand image and the contours of the hand shape.

[5]This is true for a special case that happens when the homotetic centre $\mathcal{S}$ coincides with the origin $\mathcal{O}$ of the vector space ($\mathcal{S} \equiv \mathcal{O}$), then every homothety is equivalent to an isotropic scaling (the homothety becomes a linear transformation).

Step 1 let the input image $I(x, y)$ be the output binary image of the previous step. Then, it is possible to compute the barycentric coordinates of the hand through the following equations:

$$x_0 = \frac{\sum_x \sum_y I(x,y)\, x}{S} \qquad y_0 = \frac{\sum_x \sum_y I(x,y)\, y}{S} \tag{4.33}$$

where $\mathbf{p}_0 = (x_0, y_0)$ is the centroid, and

$$S = \sum_x \sum_y I(x,y) \tag{4.34}$$

is the total amount of boundary points of the hand.

Step 2 in order to extract the central parts of the palmprint images (ROI), this step detects the three key points between fingers. The first and the last of them are used to construct a reference line segment for aligning the different palmprint images, and the midpoint between them is used to determine the central position of the ROI.

Definition 14. *For points* $p, q,$ *and* z*, with coordinates* (x, y)*,* (s, t)*, and* (v, w)*, respectively,* D *is a* distance function *if*

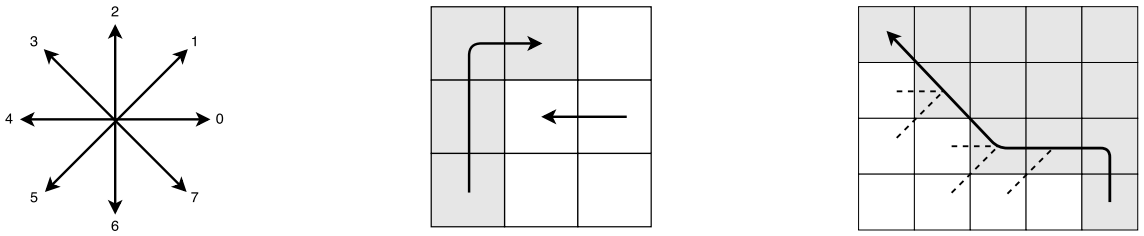
1. $D(p, q) \geqslant 0 \quad (D(p, q) = 0 \iff p \equiv q)$,
2. $D(p, q) = D(q, p)$,
3. $D(p, z) \leqslant D(p, q) + D(q, z)$.

The Euclidean distance between the points p *and* q *is defined as*

$$\begin{aligned} D(x,y) &= \left[(\mathbf{p}-\mathbf{q})^\top (\mathbf{p}-\mathbf{q})\right]^{\frac{1}{2}} \\ &= \sqrt{(x-s)^2 + (y-t)^2}. \end{aligned} \tag{4.35}$$

For this distance measure, the points having a distance less than or equal to some value v *from* (x, y) *are the points contained in a disk of radius* v *centred at* (x, y).

The key points are detected by analysing the first derivative of the distance vector $\mathbf{d}$, which contains all the distance measures between the barycentre and the contour of the hand, to find the three local minima corresponding to the key points (see Figure 4.13).



(a) (b) (c)

Figure 4.12: Direction notation: (a) 8-connectivity, (b) pixel neighbourhood search sequence in 8-connectivity, and (c) boundary tracing in 8-connectivity (dashed lines show pixels tested during the border tracing).

Definition 15. *Let the domain* X *be a metric space, then the function* d *is said to have a local minimum point in* $x_0 \in X$ *if* $\exists \epsilon > 0 \mid (\forall x \in X)\, D(x, x_0) < \epsilon \implies d(x_0) \leqslant d(x)$.

Algorithm Key points detection

Input: Image $I(x, y)$ (with centroid).
Output: Image $I(x, y)$.

1. Search from the centroid down until a pixel of the hand outline is found, then mark this point as starting pixel s_0 of the region border. Define a variable dir which stores the direction of the previous move along the border and assign for the first direction $dir = 2$ as shown in Figure 4.12(a).
2. Compute the distance between the border points $\mathbf{p}_i$ and the centroid $\mathbf{p}_0$ as in (4.35).
3. Search the 3×3 neighbourhood of the current pixel in a clockwise direction, beginning from the pixel positioned in the direction (see Figure 4.12(a)):

$$(dir + 1) \mod 8 \quad \text{if } dir \text{ is even,} \tag{4.36}$$
$$(dir + 2) \mod 8 \quad \text{if } dir \text{ is odd.} \tag{4.37}$$

   The first pixel found with the same value as the current pixel is a new bounday element. Update the dir variable.
4. **if** the current pixel (x, y) has the same coordinate x of the initial pixel **then**
   stop
   else
   goto step 2
   end if

Step 3 with this last step, we have to define a new coordinate system from the current one in order to align the palmprint image to a standard position and rotation. The line segment $\overline{AC}$, where $A(x_a, y_a)$ and $C(x_c, y_c)$ are the first and the last detected key points, defines the current coordinate system, then the square ROI can be extracted. The reference line equation can be calculated as follows:

$$(x_c - x_a) \cdot (y - y_a) - (y_c - y_a) \cdot (x - x_a) = 0 \tag{4.38}$$

with $x_a \neq x_c$ and $y_a \neq y_c$, finally we can explicit the reference line as:

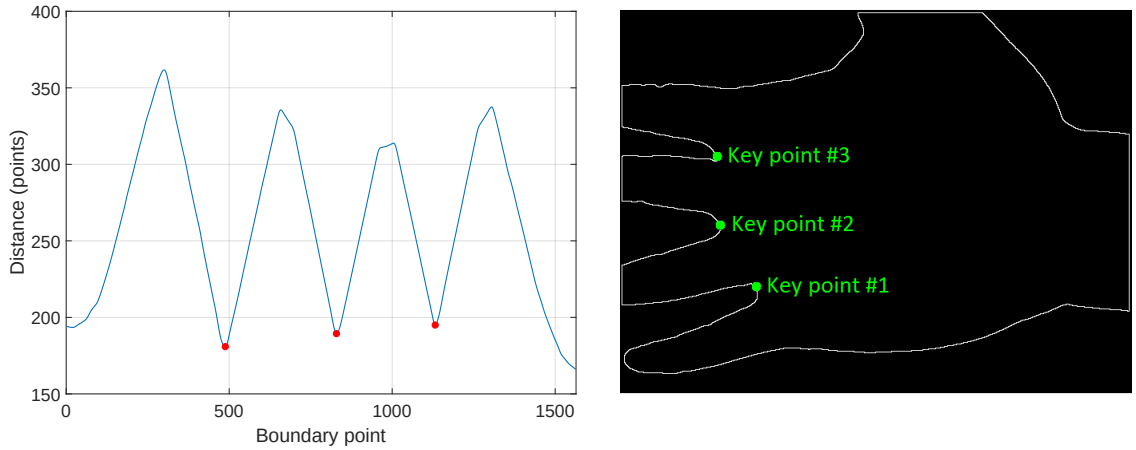
$$y = \underbrace{\left(\frac{y_c - y_a}{x_c - x_a}\right)}_{m} x + \underbrace{\left(y_a - x_a \frac{y_c - y_a}{x_c - x_a}\right)}_{q}. \tag{4.39}$$

The midpoint $m(x_m, y_m)$ is defined as:

$$x_m = \frac{1}{2}(x_a + x_c)\,, \quad y_m = \frac{1}{2}(y_a + y_c)\,. \tag{4.40}$$

Once the key points and reference line are defined, the proposed algorithm normalises the palmprint position by taking the middle point $m(x_m, y_m)$ as the centre of rotation to rotate the palmprint clockwise with an angle[6] φ.

[6]The angle between the reference line and the vertical axis is computed as $\varphi = \arctan(m) - \pi/2$.



(a) (b)

Figure 4.13: Key points detection: (a) the distance function $D(x, y)$ and (b) the key points detected using the algorithm.

In order to do this, let φ be the angle between the reference line and the vertical axis, then the rotation and scale of the palmprint image is done as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \frac{1}{s} \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x_t \\ y_t \end{bmatrix} + \begin{bmatrix} x_m \\ y_m \end{bmatrix} \tag{4.41}$$

where x_t and y_t are the translated points with respect to the midpoint $m(x_m, y_m)$, x' and y' are the points in the new coordinate system and s is a scaling factor defined as:

$$s = 1 + \frac{N - \sqrt{(x_c - x_a)^2 + (y_c - y_a)^2}}{N} \tag{4.42}$$

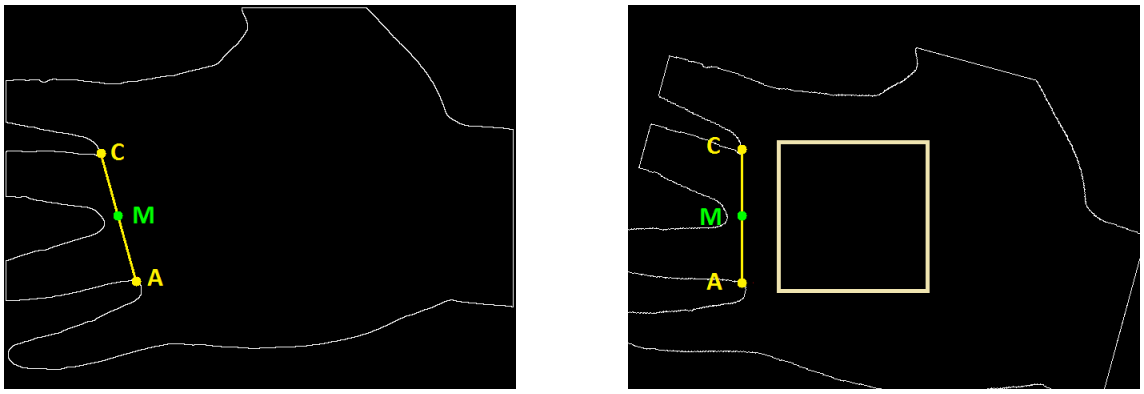
where (x_a, y_a) and (x_c, y_c) are the coordinates of the key points and N is the fixed size of the square ROI to be extracted and centred in (C_x, C_y) (see Figure 4.14).

Normalisation In order to have a pre-specified mean and variance for all palmprint images, the extracted ROI images are normalised to reduce the possible non-uniform illumination and noise.
Let the grey level value at pixel (x, y) of the ROI image be represented by $I(x, y)$, μ and σ^2 denote the mean and variance of the image, respectively, then the normalised image $I'(x, y)$ is defined as follows:

$$I'(x, y) = \begin{cases} \mu_n + k & \text{if} \quad I(x, y) > \sigma^2 \\ \mu_n - k & \text{if} \quad I(x, y) \leqslant \sigma^2 \end{cases} \tag{4.43}$$

where

$$k = \sqrt{\frac{\sigma_n^2 \, (I(x, y) - \mu)^2}{\sigma^2}}. \tag{4.44}$$



(a) (b)

Figure 4.14: ROI extraction: (a) detection of the positions of the three key points, including the midpoint, and (b) rotation and scaling of the image and ROI detection.

The values μ_n and σ_n^2 are the desired values for mean and variance, respectively, which are pre-tuned according to the image characteristics[7].
Figure 4.15 shows some examples of ROI extraction and normalisation.

Feature extraction

Once the central part (ROI) is segmented, features can be extracted for matching.
As described in Section 4.4, there are many features exhibited in a palm, such as the three principal lines caused by flexing hand and wrist in the palm, which are denominated as heart line, head line, and life line, respectively.

Principal lines are among the most stable features in palmprints, considering natural ageing processes and changes in acquisition conditions. For this reason, it is of great interest to detect them accurately.

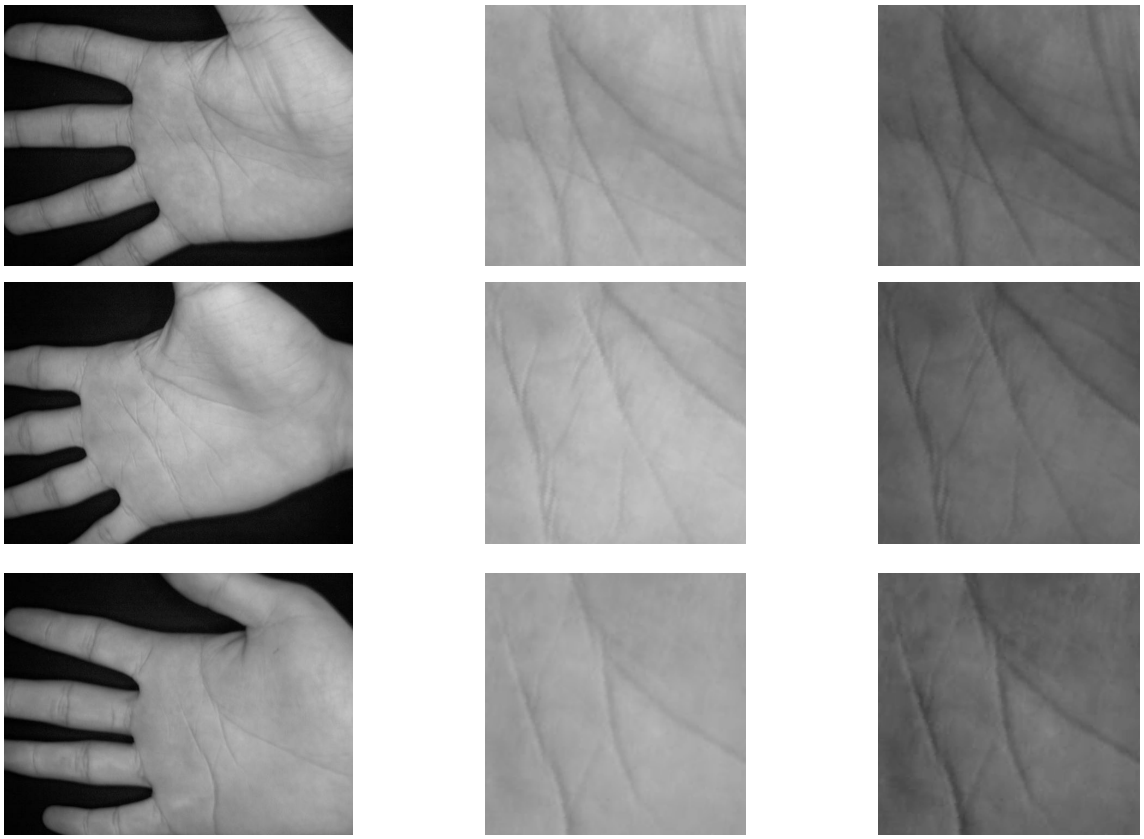
Actually, there were previous attempts to detect principal lines [75], as they can be used in several ways to improve palmprint recognition.

However, the extraction of principal palm lines is a challenging task, since palmprint images acquired by a camera (or a scanner) are low contrast images and palm lines are irregular and have different directions, depth, and shape even in the same palm.

The feature extraction stage illustrated in Figure 4.10, consists of the following steps:

1. conversion to a negative image;
2. sharpening of the image with Top-Hat filter and gradient magnitude computation;
3. linear contrast enhancement (LCE);
4. binarization with Otsu's method;
5. noise cleaning with median filter.

[7] In this work, the normalisation is the same suggested in [118], thus both the values of mean and variance have been set to 100.



(a) (b) (c)

Figure 4.15: Some results of ROI extraction and normalisation: (a) the input images, (b) the extracted ROI, and (c) the normalised images.

Negative After the last step of the preprocessing phase, the resulting enhanced ROI image $I(x, y)$ is converted to its negative[8]. The negative of an image with intensity levels in the range $[0, L-1]$ is obtained by using the negative transformation as follows:

$$I'(x, y) = sI(x, y) + (L - 1). \qquad (4.45)$$

Scaling by $s = -1$ flips the histogram, whilst the additive offset $L-1$ is required in order to have all values of the result falling in the allowable grey-scale range. This type of processing is particularly suited for enhancing white detail embedded in dark regions, in this way, the principal lines of the palm have high intensity.

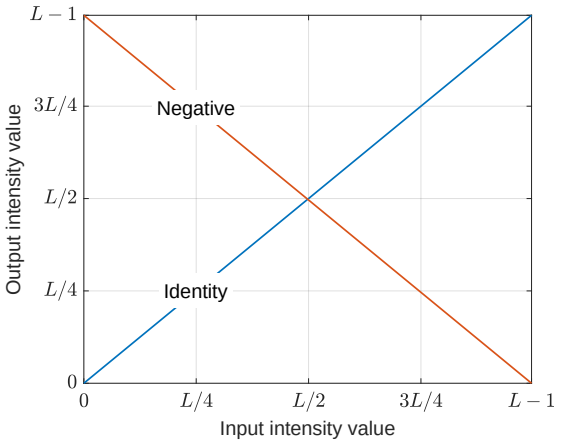


Figure 4.16: Negative transformation.

[8]Reversing the intensity levels of the image in this way produces the equivalent of a photographic negative.

Sharpening To correct uneven illumination it has been adopted a Top-Hat transform followed by a contrast enhancement to the output image. The Top-Hat transform enables the system to extract small bright objects from a varying background and is defined as the difference between the input image and its morphological opening by using a defined structuring element ρ (see Appendix A.4 for further details).
The opening of $I(x, y)$ by ρ is obtained by the erosion of $I(x, y)$ by ρ, followed by dilation of the resulting image by ρ:

$$I \circ \rho = (I \ominus \rho) \oplus \rho = \bigcup_{I(x,y) \subset \rho} I(x, y) \tag{4.46}$$

Hence, the opening of $I(x, y)$ by ρ is the union of all sets $I(x, y)$ contained in ρ. Then, the opening Top-Hat transformation of image I, denoted by TH, is given by

$$TH = I - (I \circ \rho) \tag{4.47}$$

which represents the difference set of the domain of $I(x, y)$ and the opening domain of $\mathbf{I}(x, y)$. After the morphological operations, the gradient magnitude is computed as:

$$\begin{aligned} J(x, y) = \nabla I\,(x, y) &= \begin{bmatrix} H_x(x, y) \\ H_y(x, y) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial I(x, y)}{\partial x} \\ \dfrac{\partial I(x, y)}{\partial y} \end{bmatrix} \\ M\,(x, y) &= \|\nabla I(x, y)\| \\ &= \sqrt{H_x^2(x, y) + H_y^2(x, y)} \approx \left(\left| \frac{\partial I(x, y)}{\partial x} \right| + \left| \frac{\partial I(x, y)}{\partial y} \right| \right) \end{aligned} \tag{4.48}$$

This sharpening operation returns the bright spots of the image that are smaller than the structuring element.

Contrast enhancement In order to improve the contrast of the image, a LCE consisting of a minimum-maximum contrast transformation[9] is applied. This transformation involves the identification of the lower and upper bounds from the histogram, which coincide with the minimum and maximum brightness values in the image. Then, it is possible to apply the transformation to stretch this range to fill the entire dynamic range $(0, 255)$.

Definition 16. *An image histogram is a grey-scale value distribution showing the frequency of occurrence of each grey level value. Hence, the histogram of a digital image* $I(x, y)$ *with intensity levels in the range* $[0, L - 1]$ *consists of a discrete function* $h\,(r_k) = n_k$ *where* r_k *is the* k-th *intensity value and* n_k *is the number of pixels in the image with intensity* r_k.

Let the variable r denote the intensities of a digital image to be processed assuming that $r \in [0, L - 1]$, with $r = L - 1$ representing white and $r = 0$ representing black. The transformation will be in the form $s = T(r)$, with $0 \leqslant r \leqslant L - 1$.
This transformation produces an output intensity level s for every pixel in the input image having intensity r. Assumed that $T(r)$ is a monotonically increasing function in the interval $[0, L - 1]$, and $0 \leqslant T(r) \leqslant L - 1 \quad \forall r \in [0, L - 1]$.

[9] LCE is best applied to images with near-Gaussian histograms, i.e., when all the brightness values fall within a single, relatively narrow range of the histogram and only one mode is apparent.

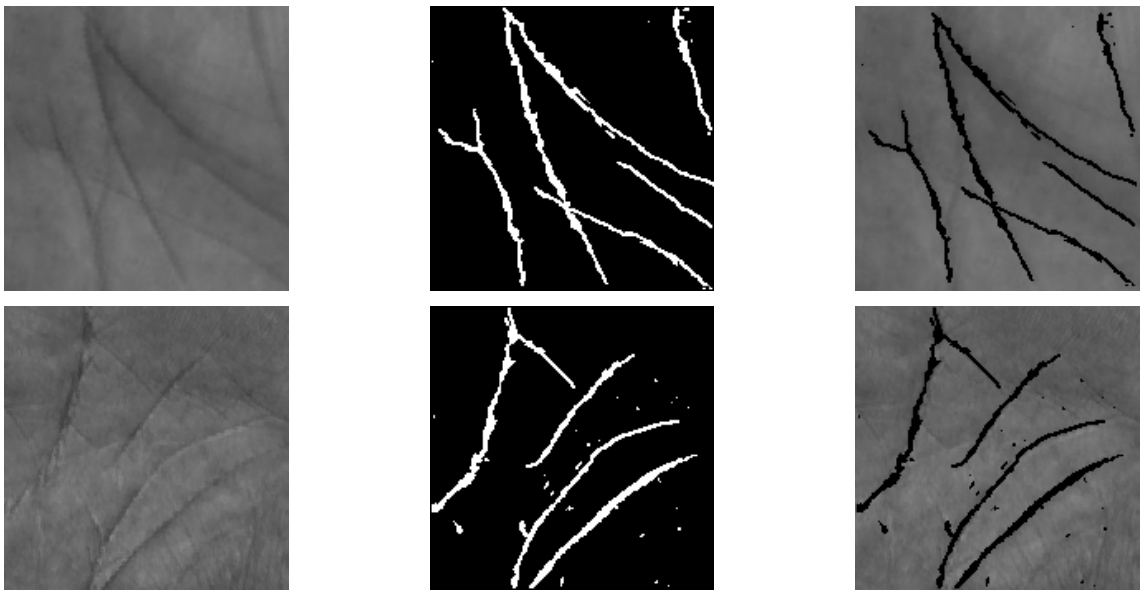
Let $\min(h(r))$ and $\max(h(r))$ be the lower and the upper limits of the histogram respectively, then each pixel of the image is scaled to the range $(0, q_k)$ according to the equation:

$$I'(x,y) = \left(\frac{I(x,y) - \min(h(r_k))}{\max(h(r_k)) - \min(h(x))} \right) q_k. \tag{4.49}$$

where $I(x,y)$ is the original input brightness value, $I'(x,y)$ is the output brightness value, and q_k is the maximum value of the output dynamic range.

Global binarisation A global thresholding is applied at the grey-scale image resulting from the previous sharpening filter and contrast enhancement. The threshold is calculated with Otsu's method [116], whose details are given in Appendix A.1.

Noise cleaning As a last step, a common order-statistic filter[10] (e.g., median filter) is used to remove noise and trivial lines from the image.



(a) (b) (c)

Figure 4.17: Some results of palm line extraction on CASIA (upper row) and PolyU (lower row) databases: (a) original ROI images, (b) boolean matrices representing the palm lines, and (c) original ROI images overlapped with the extracted palm lines.

4.4.3 Dynamic algorithm

The dynamic algorithm used for matching consists of the early version of the proposed system, which is deeply described in Section 4.1 and whose updating equations are

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)$$

$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k) + \mu \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)$$

For further details, please refer to the Section 4.1 and the references therein.

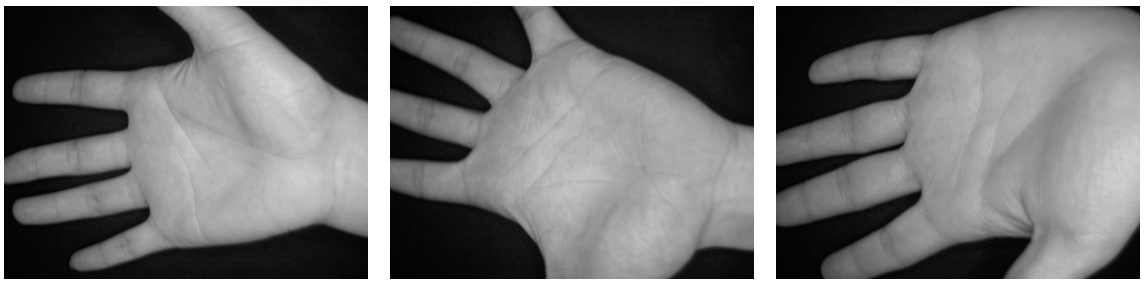
[10] Order-statistic filters are non-linear spatial filter whose response is based on ranking the pixels contained by the kernel, and then replacing the value of the center pixel with the value determined by the ranking result.

4.4.4 Experimental results

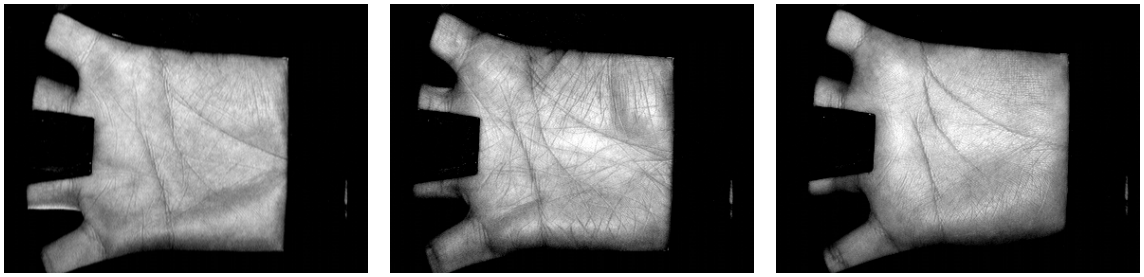
Extensive experiments carried out to assess the performance and the effectiveness of the proposed algorithm are described in this section, with particular regard to the databases used in the experiments and experimental setup, parameter optimisation, performance assessment, noise-rejection property, and computational efficiency of the algorithm.

Dataset and experimental setup

The performance of the proposed palmprint verification system has been tested upon two databases that have been worldwide shared for research purposes: *CASIA Palmprint Database* [28] and *PolyU Palmprint Database II* [63].
The first dataset contains 5 502 palmprint images with 8 bit grey level of size 640×480 pixels at 96 dpi resolution captured from 312 subjects by a low resolution CCD-based device operating in contactless mode. For each subject there are palmprint images from both left and right hand captured eight times at different times from people of different ages. The second dataset, which has a lower resolution, contains 7 752 palmprint images with 8 bit grey level of size 384×284 pixels at 75 dpi resolution captured from 386 palms of 193 subjects: these palmprint images have been collected in two sessions with the average interval over two months, where about 10 samples have been captured from each palm for each session. Figure 4.18, illustrates some examples of images from both the databases.



(a)



(b)

Figure 4.18: Some examples of raw left and right hand palm images: (a) from CASIA database and (b) from PolyU database (lower resolution).

As reported in Table 4.1, the experiments have been performed by taking

1. 16 sample palmprint images of the left and right hand (eight images for hand) of all 312 subjects, for a total of 4 992 images, from the *CASIA Palmprint Database*,
2. 10 sample palmprint images of 380 individuals related to the first and second sessions, for a total of 7 600 images, from the *PolyU Palmprint Database II*.

	CASIA	PolyU
Subjects	312 (624)	380
Samples for subject	16 (8)	20
Total samples	4 992	7 600
Hand holder	No	Yes
Dimensions (px)	640×480	384×284
dpi	96	75
Bit depth	8	8

Table 4.1: Specifications of the PolyU and CASIA palmprint databases used in the experiments.

Without loss of generality, it is possible to consider the left and right hand palm images of the same person as belonging to different individuals. This setup constitutes a total number of experiments equal to

1. $\binom{4992}{2} = 12\,457\,536$, including $624 \times \binom{8}{2} = 17\,472$ genuine experiments (the others are impostor experiments), for the *CASIA palmprint database*,
2. $\binom{7600}{2} = 29\,104\,200$, including $380 \times \binom{20}{2} = 72\,200$ genuine experiments, for the *PolyU palmprint database*.

	CASIA	PolyU
Genuine experiments	17 472	72 200
Impostor experiments	12 440 064	29 032 000
Total experiments	12 457 536	29 104 200

Table 4.2: Experimetns carried out on the CASIA and PolyU palmprint databases to assess the performance of the proposed system.

Parameter optimisation

Since the verification rate of the system varies depending on the values of λ and μ in the dynamic algorithm, these parameters need to be properly set, as discussed in Section 4.1. Thus, given the set $\mathcal{N}_{i,j}$ of neighbouring points of the generic point $p(i,j)$ we can choose the "fading factor" $0 < \lambda < 1$ and the coefficient $\mu > 0$. Figure 4.19 depicts the region of convergence of the system according to Eqq. (4.10) and (4.11), which represent the upper and lower bounds of the region, respectively. Moreover, since the thickness of the palm lines consists of a few pixels, it is reasonable to consider a square matrix with radius $\nu = 2$ (which amounts to considering a number of neighbouring pixels equal to $m = 25$) for a perfect coverage of a palm line avoiding excessive overlaps with other palm lines – or background – contained in the image to be compared. Therefore, to determine the value of the parameters that guarantees the best performance, a test over a subset of the CASIA palmprint database for many different values of the parameter pairs has been performed. In this phase, it has been verified every test pair for each of the palmprint images in the database and for each pair of the parameter values. The subset of the CASIA palmprint database consists of 360 palmprint images and the number of parameter pairs (λ, μ) inside the region of convergence is equal to 70, thus the amount of tests performed is 4 523 400.

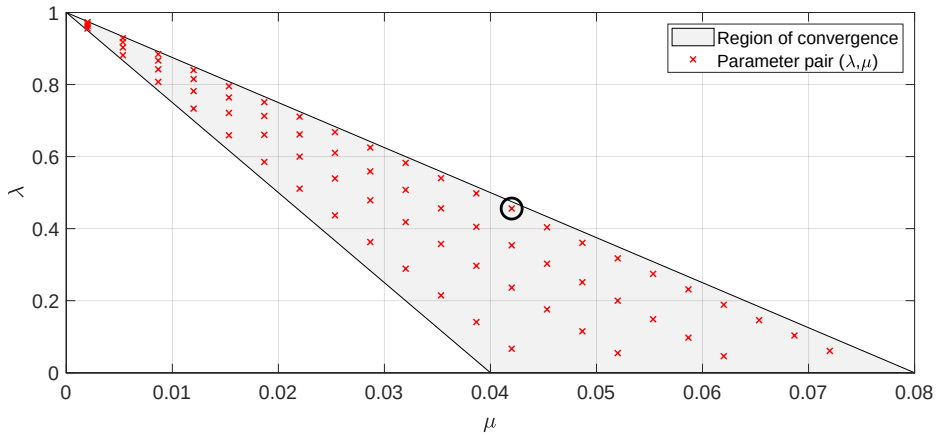


Figure 4.19: Region of convergence of the system delimited by (4.10) and (4.11) (with $m = 25$ and $b = \frac{1}{2}$), and values of the parameter pairs (λ, μ). Best results have been obtained by the highlighted pair of values.

As a result of these tests, the parameter pair that provides the best verficiation performance is given by:

$$\begin{cases} \lambda = 465 \cdot 10^{-3}, \\ \mu = 42 \cdot 10^{-3}. \end{cases} \tag{4.50}$$

These values have been used in the experiments for both the databases in order to test the performance of the proposed method, even without parameter re-tuning for the PolyU database. It is also important to choose a fixed number of iterations, after which we can rely on convergence of the dynamic system. From Figure 4.20 it is clear that, after few iterations, the reached matching value can be considered good enough with respect to the highest value achieved. Thus, to save computation time, the number of iterations for all the tests has been fixed to 50.

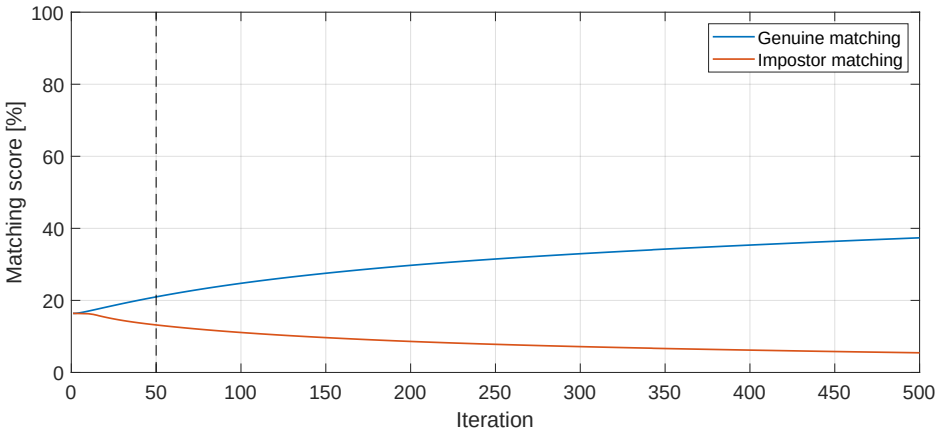


Figure 4.20: Average score against number of iterations in the case of true positive (genuine) recognition and true negative (impostor) recognition for palm line matching.

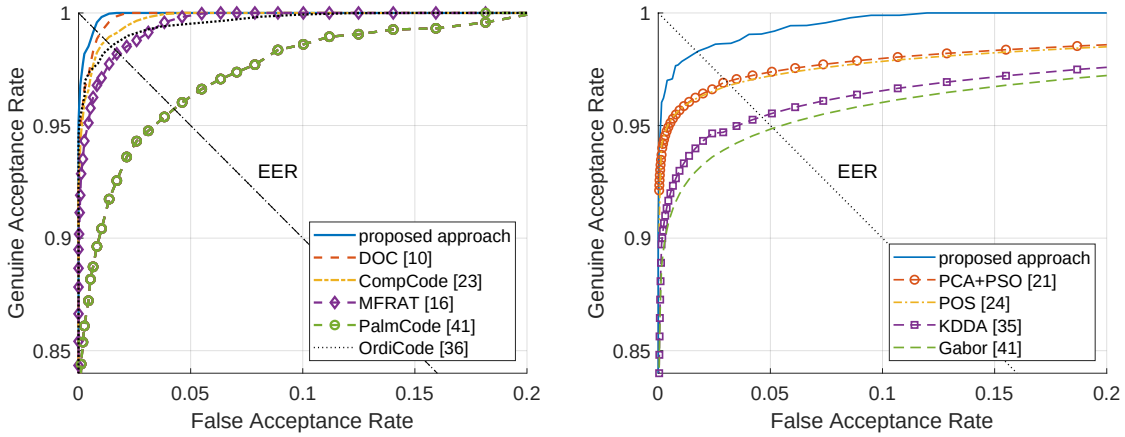
Performance assessment and comparison

The performance results of the developed palmprint recognition system are here compared with other algorithms by using the datasets as previously described. Hence, for comparison purposes, some of the representative techniques mentioned in Section 4.4.1 are used.

Method	CASIA	PolyU
CompCode [81]	0.0201	0.0122
PalmCode [155]	0.0367	0.0432
OrdiCode [137]	0.0175	0.0150
MFRAT [66]	–	0.0180
DOC [45]	–	0.0092
Proposed method [118]	0.0123	0.0071

Table 4.3: Comparison of verification results in terms of EER for several published methods [46, 114].

Table 4.3 lists the verification results, expressed in terms of EER, for different approaches, on databases that contain palmprints captured through touch-based (PolyU) and touch-less (CASIA) devices. From the results, it is clear that the CASIA database is more challenging than the PolyU database in terms of palmprint verification. However, the proposed method achieved good results outperforming the other approaches.
To better illustrate the verification performances, the ROC curves[11] of some of the above-mentioned methods and the proposed algorithm are illustrated in Figure 4.21. In particular, the dynamic algorithm has achieved: a GAR_2 that is almost 100% and a GAR_3 that is greater than 97% on the PolyU dataset, whilst for the CASIA dataset the system has obtained a GAR_2 equal to 98.5% and a GAR_3 that is similar to the result obtained on the other dataset.



(a) (b)

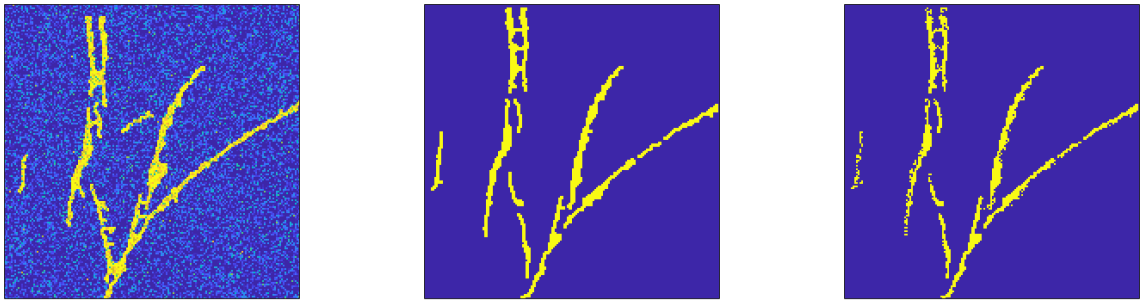
Figure 4.21: Comparative genuine acceptance rate against false acceptance rate graphs, for the proposed approach and for other state-of-the-art methods on (a) PolyU Palmprint Database and (b) CASIA palmprint database.

[11]The receiver operating characteristic curves are obtained by plotting GAR against FAR.

Noise immunity

To conduct experiments on noisy palmprints and demonstrate the robustness of the dynamic algorithm with respect to noise, the system has been tested by comparing palmprints corrupted by AWGN against normal palmprints.

Figure 4.22 depicts an example of the dynamic algorithm behaviour: it can be seen from the output image that the subject is easily recognised, validating the noise-rejection property described in Section 4.3.



(a) (b) (c)

Figure 4.22: Dynamic algorithm behaviour on different principal line images from the same subject: (a) image affected by statistical noise having zero-mean Gaussian distribution and variance equal to 0.05, (b) image to be compared, and (c) surviving points after the system evolution.

Table 4.4 reports the result of noise-rejection experiments[12] and shows that the performance of the system is not that much degraded: even in presence of noise, the GAR_1 is greater than 98.6% in both the databases. However, we have obtained better results on the PolyU database achieving a GAR of 97.8% with an EER equal to 0.022 against a GAR of 96.9% with an EER equal to 0.031 on the CASIA database.

Therefore, the experiments demonstrate that the proposed algorithm can recognise people by their palmprint even in presence of a reasonable amount of noise.

	Database			
	CASIA	$CASIA_{AWGN}$	PolyU	$PolyU_{AWGN}$
EER	0.0123	0.0311	0.0071	0.0219
GAR	0.9877	0.9689	0.9929	0.9781
GAR_1	1.0000	0.9863	1.0000	1.0000
GAR_2	0.9863	0.9475	0.9982	0.9625
GAR_3	0.9685	0.9107	0.9711	0.9113

Table 4.4: Numerical results comparison considering both the databases in normal conditions (e.g., normal palmprint images) and in presence of AWGN with $\mu = 0$ and $\sigma^2 = 0.05$.

[12]The experiments are carried out by corrupting the images with a statistical noise having zero-mean Gaussian distribution and variance σ^2 as $\sim \mathcal{N}(0, 0.05)$.

Computational efficiency

The experiments for the proposed approach have been conducted on a virtual machine equipped with two dedicated processors and 2048 MB RAM hosted on an Intel Core i7-4510U CPU (2.6 GHz), 8 GB RAM running a 32-bit Microsoft Windows 10. The code has been implemented using Matlab 8.0 and to get the following results each part of code has been performed for 200 times, then the average time has been considered. Thus, the computational times required for preprocessing, palm line extraction, and palmprint matching are 376 ms, 49 ms, and 295 ms, respectively, thus the average response time for verification is about 0.72 s.

4.5 Example II: a robust palm vascular-based biometric recognition system for personal identification

Vascular pattern features such as palm veins [25, 153, 156, 160], finger veins [30], hand veins [87], and hand dorsal veins [93], are an emerging biometric trait that has recently received considerable interest from both the research community and industries [108]. In recent years, the use of palm veins as a trait for automated secure personal authentication has been largely investigated due to their advantages over other biological features. In fact, the subcutaneous vascular pattern of the human body is unique to every individual, even between identical twins [87], does not vary during the course of a person's life, and lies underneath the human skin ensuring confidentiality and robustness to counterfeiting, as opposed to other intrinsic and extrinsic biometric traits that are more vulnerable to spoofing, thus leading to important security and privacy concerns [76]. In addition, since vascular patterns are typically acquired by touch-less devices, they allow for a secure authentication method ensuring high user acceptability without discomfort. Veins are part of the network structure of blood vessels underneath the human skin and are almost invisible in normal lighting conditions [158]. However, it is possible to identify the vascular pattern through NIR illumination with wavelength in the range from 750 nm up to 2000 nm[13].

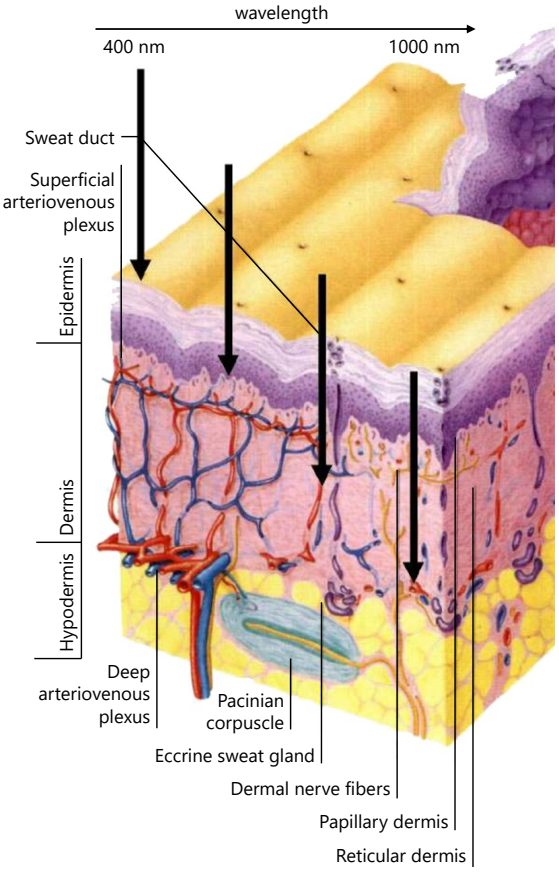


Figure 4.23: A cross-sectional anatomy of the human skin in which different light wavelengths will penetrate to different layers [52].

[13]In the course of this study, it has also been designed and developed a hand-based multispectral image acquisition system which will be used in a future work.

The incident light in the near-infrared spectrum penetrates into the human biological tissues up to 3-4 mm detecting the vascular pattern underneath the skin [143]. Veins can be distinguished from arteries because arteries carry oxygenated blood that contains oxyhemoglobin, whilst veins carry deoxygenated blood that contains deoxyhemoglobin, which has a different absorbency rate under near-infrared radiations: deoxyhemoglobin absorbs a higher level of near-infrared radiations, which allows to detect and isolate vein patterns. As a result of the acquisition in presence of near-infrared illumination, vascular patterns in raw images appear much darker than all other tissues, which facilitates the feature extraction step for matching.

4.5.1 Related work

Various methods for human authentication through palm vascular pattern matching have been proposed in literature. Among them, the work of Zhou and Kumar [160] presented a neighbourhood matching Radon transform (NMRT)-based method aimed at extracting line-like palm vascular features and a Hessian phase-based method to extract palm vein features analysing the eigenvalues of Hessian matrix of the input image. The matching score is computed making use of the Hamming distance. Khan *et al*. [78] used multidirectional representation derived from the non-subsampled contourlet transform, which is binarised into a hash table. Finally, a L_0-norm approach is used for matching. In another study, Sun and Abdulla [136] introduced an algorithm based on curvelet transform to obtain curve-like features, whilst Hamming distance is used for matching. The work of Al-juboori *et al*. [4] proposed the use of bank of Gabor filters to extract the vein features, followed by a dimensionality reduction using the Fisher discriminated analysis (FDA) method, and finally the use of the nearest neighbours technique for matching. The study of Kang and Wu [76], instead, utilised an improved local binary pattern (LBP) method based on mutual foreground for feature extraction and an improved χ^2 distance for matching, whilst the approach proposed in [142] by Wang *et al*., made use of the discriminative LBP algorithm for palm vein feature extraction and an improved χ^2 distance for verification. Another approach proposed by Kang *et al.* [77], used a local invariant feature extraction technique based on the square root of the scale invariant feature transform (RootSIFT). The work of Ma *et al*. [97] presented an adaptive two-dimensional Gabor filter for feature extraction, which are compared using the minimum normalised Hamming distance method, whilst Ahmad *et al*. [2] introduced the use of the wave atom transform (WAT) method for feature extraction and the normalised Hamming distance to compute the matching score. Hong *et al*. [60] made use of a hierarchical classifier based on the fusion of the block dominant orientation code (BDOC) and block-based histogram of oriented gradient (BHOG) features from different spectrum bands (red, green, blue and NIR). Finally, Elnasir *et al.* [39] suggested a combined approach based on wavelet scattering (WS) and spectral regression kernel discriminant analysis (SRKDA) for dimensionality reduction to enhance the discrimination, since the dimension of WS generated features is quite large.

All these techniques can be grouped in three main categories based on the nature of the features used for matching [80]: 1. holistic approaches based on multilinear subspace learning: dimensionality reduction techniques are used to project palm vascular images into subspaces aimed at capturing the main features of the palm; 2. line/curve matching using vessel extraction based on line-like feature extraction techniques that involve spatial domain filters for line/curve extraction; 3. texture based codes, which make use of the orientation of lines as features.

4.5.2 Hand palm image processing

Usually, palmprint images in the near-infrared band contain not only the blood vessels used to authenticate a person, but also a region of not-interest (e.g., shades, wrist, image background). Moreover, they have different size and orientation and could also be corrupted by noise. All these factors may affect the accuracy in processing and verification performance [159]. Thus, a preprocessing of all palmprint images is required to enable the feature extraction phase. Figure 4.24 outlines the preprocessing and feature extraction phases illustrating all the main steps involved in the vascular pattern extraction from a raw NIR-based hand palm image.

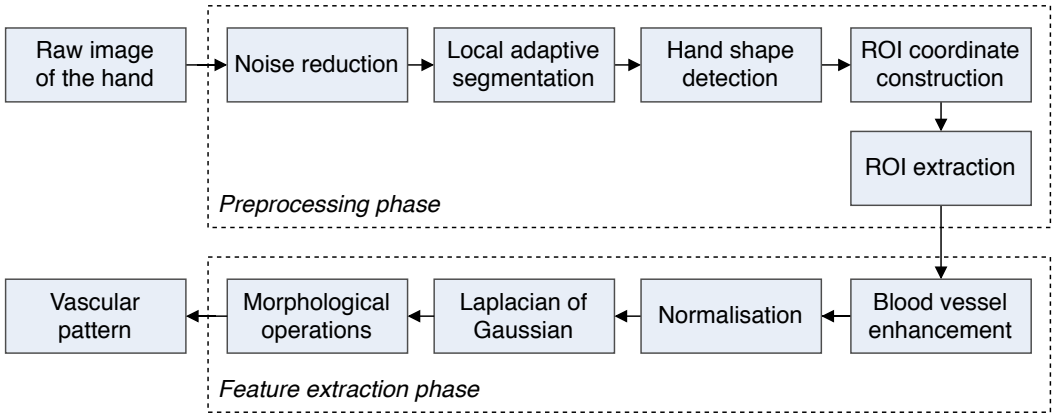


Figure 4.24: Block diagram of the hand palm image processing for palm vascular patterns recognition.

Preprocessing

The preprocessing elaboration is required to extract the central region of interest from the input image, as illustrated in Figure 4.25. As outlined in Figure 4.24, the major steps involved in the preprocessing of raw images are the same as those reported in the Section 4.4.2, except the last one:

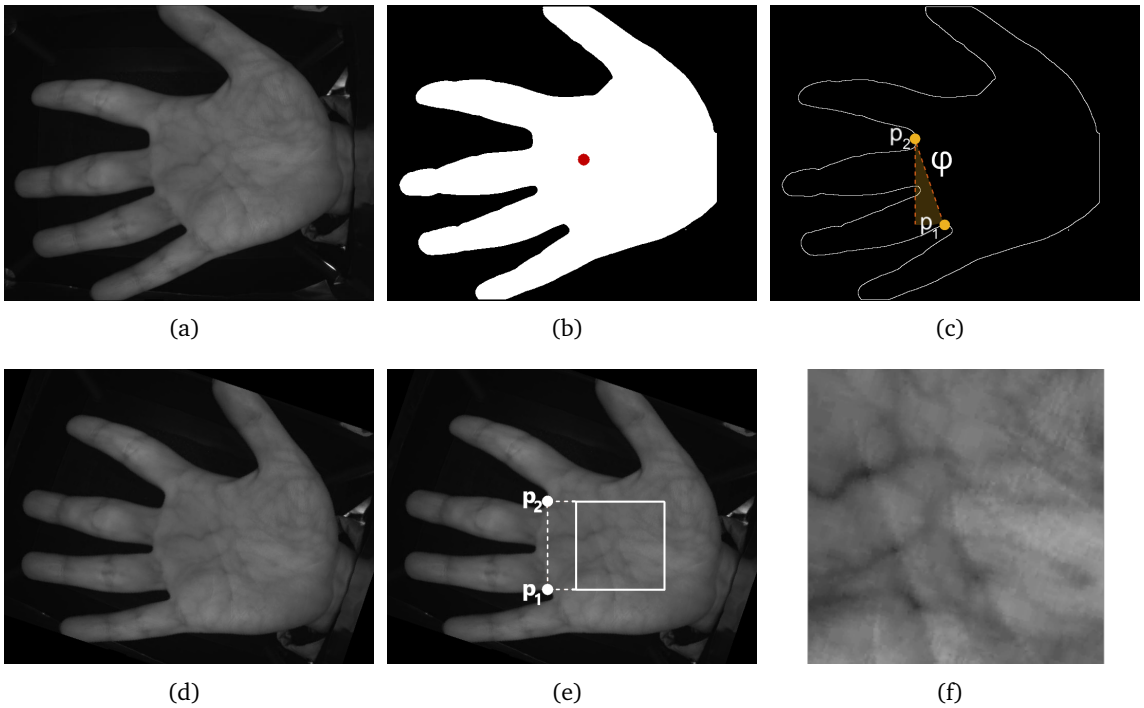
1. noise reduction by means of a non-linear spatial filter,
2. local adaptive clustering-based image segmentation,
3. hand shape detection,
4. standard hand palm coordinate construction,
5. ROI extraction.

Feature extraction

Since the features are used for matching, feature extraction plays a key role in biometric identification and authentication systems. The proposed feature extraction stage makes use of the following steps[14]:

1. highlighting blood vessels by enhancing contrast and sharpness,
2. Laplacian of Gaussian,
3. morphological operations aimed at cleaning the noise in the final image.

[14]Before diving into each step details, the reader is invited to take a look at the Appendices A.3 and A.4 in order to better understand the following sections.



(a) (b) (c)

(d) (e) (f)

Figure 4.25: Major steps involved in the preprocessing of raw palm images: (a) original NIR hand image, (b) noise reduction, local adaptive clustering-based image segmentation, and barycentric coordinates computation, (c) hand shape extraction, key points detection, and computation of angle ϕ used to build a new coordinate system, (d) image scaling and alignment in a standard pose, (e) detection and extraction of the ROI, and (f) final result.

Blood vessel enhancement This step is aimed at highlighting the blood vessels so that they are easily distinguishable from the background. To correct uneven illumination and to enhance the contrast it has been combined the Top-Hat and Bottom-Hat transforms, used to detect bright (dark) objects from a varying dark (bright) background.
The Top-Hat transform is defined as the difference between the input image $I(x, y)$ and its morphological opening by a cross shaped structuring element $\rho \subseteq \mathbb{Z}^2$:

$$\rho = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{4.51}$$

$$TH = I - (I \circ \rho) = I - ((I \ominus \rho) \oplus s) \tag{4.52}$$

whilst the Bottom-Hat transform is defined as the difference between the closing of the input image $I(x, y)$ by the structuring element ρ and the input image itself:

$$BH = (I \bullet \rho) - I = ((I \oplus \rho) \ominus \rho) - I \tag{4.53}$$

where the opening is obtained by the erosion of $I(x, y)$ by ρ followed by dilation of the resulting image by ρ, and the closing is obtained by the dilation of $I(x, y)$ by ρ followed by erosion of the resulting image by ρ.

Then, to remove the bright objects and enhance the black ones that represent the blood vessels, it has been adopted the Top-Hat and Bottom-Hat transforms as follows:

$$\begin{aligned} I_e &= I - TH - BH \\ &= I - (I - (I \circ \rho)) - ((I \bullet \rho) - I) \end{aligned} \tag{4.54}$$

After this operation, a normalisation [61] is applied to preset the values of mean and variance for all palm images:

$$I_n(x,y) = \begin{cases} \mu_n + k & \text{if} \quad I_e(x,y) > \sigma^2 \\ \mu_n - k & \text{if} \quad I_e(x,y) \leqslant \sigma^2 \end{cases} \tag{4.55}$$

where

$$k = \sqrt{\frac{\sigma_n^2 \left(I_e(x,y) - \mu\right)^2}{\sigma^2}} \tag{4.56}$$

with $\mu_n = 128$ and $\sigma_n = 40$, determined experimentally [119].

Laplacian of Gaussian The Laplacian is a two-dimensional isotropic operator used to estimate the second spatial derivative of an image and is commonly used to extract line-like features, since can preserve the pattern suppressing the noise at the same time [136]. In fact, to decrease its sensitivity to noise the operator is applied to an image already smoothed by a two-dimensional Gaussian operator, whose expression is given by

$$G(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right). \tag{4.57}$$

Since convolution and differentiation are the only linear operators involved, it is possible to interchange them:

$$\nabla^2 \underbrace{\big(G(x,y) * I_n(x,y)\big)}_{\text{Gaussian smoothing}} = I_n(x,y) * \nabla^2\big(G(x,y)\big). \tag{4.58}$$

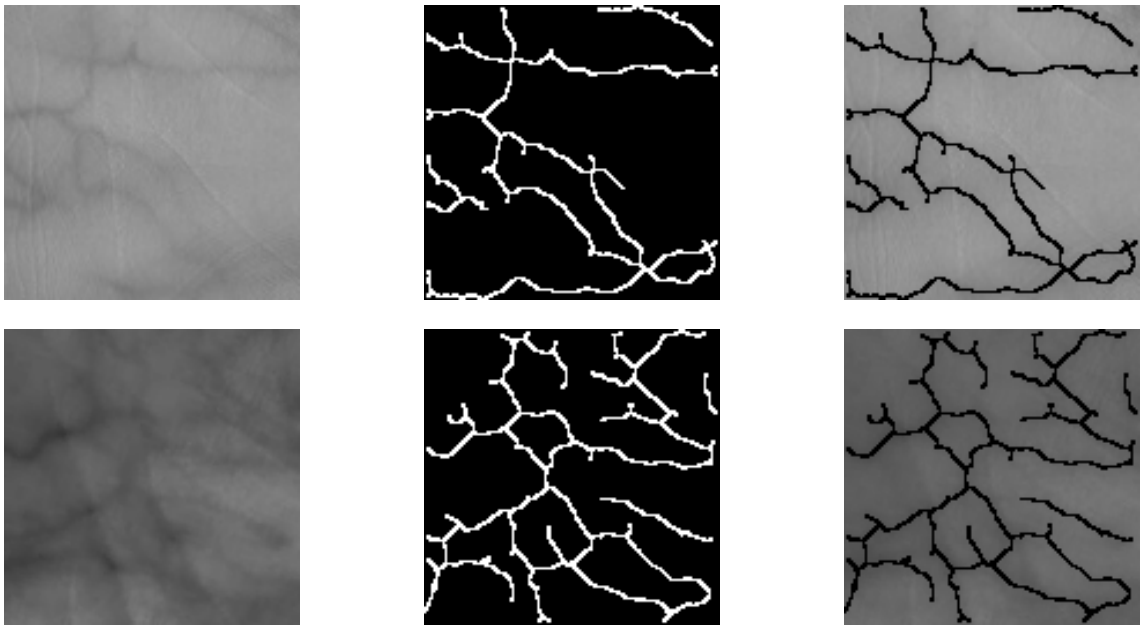
Hence, the Laplacian of Gaussian can be precomputed as:

$$\begin{aligned} \nabla^2 G(x,y) &= \nabla^2 \left\{ \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \right\} \\ &= -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2+y^2}{2\sigma^2}\right) \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &= \frac{x^2+y^2-2\sigma^2}{\sigma^4} G(x,y). \end{aligned} \tag{4.59}$$

Morphological operations They are aimed at cleaning the vascular pattern image from small objects and noise such as random bright spots on black background and black holes on bright components. To accomplish this goal we use a morphological filter composed of an opening followed by a dilation through a structuring element ρ as follows:

$$F = (I \circ \rho) \bullet \rho. \tag{4.60}$$

Finally, an iterative thinning transformation is applied to reduce a foreground object to a minimal connected stroke preserving the topology [89], since the final vascular pattern image is homotopically equivalent to the input image. Figure 4.26 exemplifies the vascular pattern extraction method.



(a) (b) (c)

Figure 4.26: Results of the proposed vascular pattern extraction method on PolyU (upper row) and CASIA (lower row) databases: (a) original ROI images, (b) boolean vascular pattern matrices, and (c) original ROI images overlapped with the extracted vascular patterns.

4.5.3 Dynamic algorithm

The dynamic algorithm used for matching consists of the enhanced version of the proposed system, which is deeply described in Section 4.2 and whose updating equations are

$$X_{i,j}(k+1) = \lambda X_{i,j}(k) + \underbrace{\mu \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)}_{\text{cross-matching}} + \underbrace{\nu^k \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)}_{\text{initial expansion}}$$

$$Y_{i,j}(k+1) = \lambda Y_{i,j}(k) + \underbrace{\mu \sum_{h,l \in \mathcal{N}_{i,j}} X_{h,l}(k)}_{\text{cross-matching}} + \underbrace{\nu^k \sum_{h,l \in \mathcal{N}_{i,j}} Y_{h,l}(k)}_{\text{initial expansion}}$$

Remark 2. *We recall that*

$$\lim_{k \to \infty} \nu^k = 0$$

since $0 \leqslant \nu \leqslant 1 - \lambda$*, in view of the constraints* (4.18) *and* (4.19)*. Hence, asymptotically the recursion becomes identical to that in the dynamical system* (4.12) *achieving the same steady-state condition.*

For further details, please refer to the Section 4.2 and the references therein.

4.5.4 Experimental results

The performance of the proposed palm vascular pattern authentication system has been tested upon the *PolyU multispectral palmprint database* [62] and the *CASIA multispectral palmprint database* [27], which are worldwide shared for research purposes and whose details are given in Table 4.5.

Dataset and experimental setup

The first database consists of 6 000 palmprint images for each electromagnetic spectrum, captured from 250 subjects, including 195 males and 55 females, by a CCD-based device. Samples have been collected in two separate sessions, in each of which every individual has provided 6 images for each palm. Therefore, 24 images of each illumination from 2 palms have been collected from each subject. All images are 8 bit grey-scale of size 352×288 pixels at 96 dots per inch (dpi) resolution. The second database consists of 1 200 palmprint images for each electromagnetic spectrum, captured from 100 subjects by a CCD-based device. Samples have been collected in two separate sessions, in each of which every individual has provided 3 images for each palm. Between two samples, it has been allowed a certain degree of variations of hand postures in order to increase diversity of intra-class samples and simulate practical use. All images are 8 bit grey-scale of size 768×576 pixels at 96 dpi resolution.
Even though in both databases, for each subject there are palmprint images from both left and right hands captured from people of different ages at different times, it is possible to consider the left and right hand palm images of the same person as belonging to different individuals, without loss of generality.

	PolyU	CASIA
Subjects	250	100
Samples for subject	24	12
Total samples for each spectrum	6000	1200
Hand holder	yes	no
Dimensions (px)	352×288	768×576
dpi	96	75
Bit depth	8	8
Spectrum bands	4	6
Wavelengths [nm]	470, 525, 660, 880	460, 630, 700, 850, 940, WHT

Table 4.5: Specifications of the PolyU and CASIA multispectral palmprint databases used in the experiments.

In the following experiments, tests have been performed by taking 12 samples in the NIR spectrum at 880 nm of the left and right hands of all the subjects from the *PolyU multispectral palmprint database* and 6 samples in the NIR spectrum at 850 nm of the left and right hands of all the subjects from the *CASIA multispectral palmprint database*, for a total of 6 000 and 1 200 samples, respectively. Furthermore, in order to increase the amount of intra-class tests and to comparatively assess the performance from various approaches, both hands have been considered to belong to different subjects [2, 4, 77, 97, 136, 142].

As a matter of fact this setup constitutes a total number of experiments equal to:

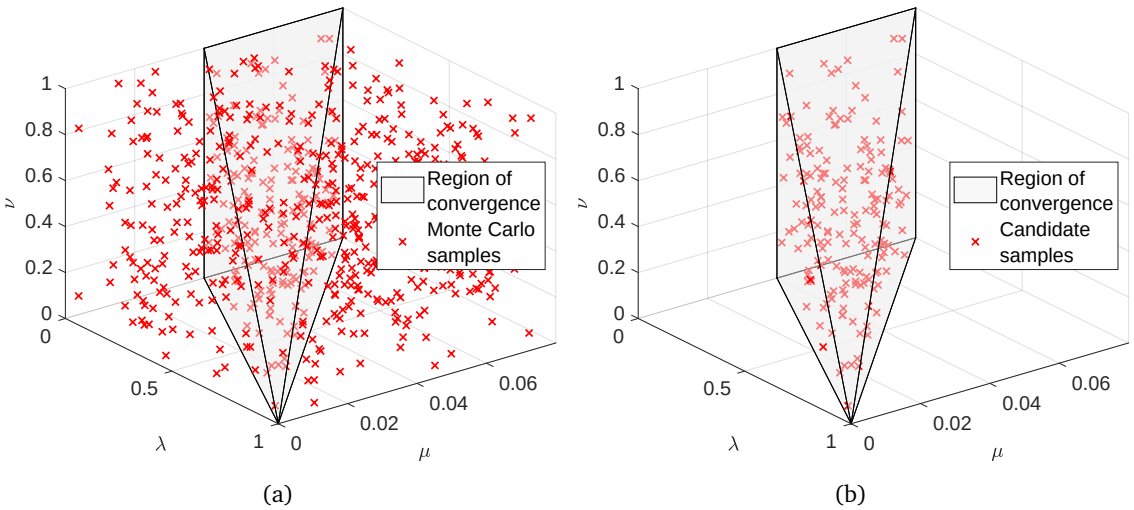
1. $\binom{6\,000}{2} = 17\,997\,000$, including $2 \times 250 \times \binom{12}{2} = 33\,000$ intra-class experiments for the PolyU database,
2. $\binom{1\,200}{2} = 719\,400$, including $2 \times 100 \times \binom{6}{2} = 3\,000$ intra-class experiments for the CASIA database.

	CASIA	PolyU
Genuine experiments	3 000	33 000
Impostor experiments	716 400	17 964 000
Total experiments	719 400	17 997 000

Table 4.6: Experimetns carried out on the CASIA and PolyU multispectral palmprint databases to assess the performance of the proposed system.

Parameter optimisation

Since the proposed approach for matching is based on a linear parameter-dependent system, it is very important to set its internal parameters in order to maximise the system performance. Hence, in this phase we have carried out a one-time parameter tuning procedure which consists of a massive experiment to estimate the values of the parameters λ, μ, and ν that maximise the accuracy of the system. Thus, given the set $\mathcal{N}$ of nearby points of a generic point $p(x, y)$, it is convenient to set the parameters in accordance with the criteria (4.18)–(4.20). To define the set $\mathcal{N}$, it is reasonable to consider a small radius as $\delta = 2$, since the thickness of blood vessels typically amounts at most to a couple of pixels[15]. This choice allows a perfect coverage of a blood vessel and avoids excessive unwanted overlaps with other blood vessels in the comparison image.



(a) (b)

Figure 4.27: Convergence domain of the system bounded by the hard constraints (4.18)–(4.20) (with $n(\mathcal{N}) = 25$ and $b = 1/2$), and valid Monte Carlo samples (λ, μ, ν).

[15] A radius equal to 2 means that the cardinality of $\mathcal{N}$ is equal to $n(\mathcal{N}) = 25$.

	Parameter set						
	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}
EER	0.002	0.019	0.014	0.032	0.008	0.029	0.021
GAR_1	1.000	0.995	0.995	0.983	0.998	0.989	0.994
GAR_2	0.995	0.974	0.985	0.957	0.994	0.958	0.975
GAR_3	0.993	0.944	0.958	0.895	0.983	0.906	0.952
GAR_4	0.988	0.882	0.922	0.814	0.969	0.844	0.900

Table 4.7: Comparative analysis of the performance in terms of equal error rate and genuine acceptance rate using the subset of the CASIA database considering the near-infrared spectrum band at 940 nm (detailed values have been highlighted in Appendix C).

The suitable parameter values, in terms of accuracy and convergence speed, have been found by means of a one-time experiment, conducted over a subset of the *CASIA multispectral palmprint database*. Since it is not possible to thoroughly invesitgate in the convergence domain to find the optimal parameter values, the candidate parameters have been chosen using a Monte Carlo sampling-based approach, generating a large number of pseudo-random points in the space, selecting the only points within the convergence domain, and using the candidate parameters to test the behaviour of the system. Figure 4.27 depicts the convergence domain of the system according to the hard constraints argued in Chapter 4. The subset of the database considers half of the right hand samples of all the subjects acquired in the spectrum band at 940 nm, whilst the number of parameter sets (λ, μ, ν) generated by the Monte Carlo sampling and belonging to the convergence domain is equal to 176. Hence, the amount of the tests performed is $176 \times \binom{300}{2} = 7\,893\,600$.
Figure 4.28 illustrates a comparative analysis of the performance by plotting the genuine acceptance rate against the false acceptance rate for several different parameter sets (λ, μ, ν), whilst Table 4.7 presents detailed results in terms of equal error rate and genuine acceptance rate achieved by each parameter set (see Appendix C for a more comprehensive overview about the parameter optimisation experiment results).

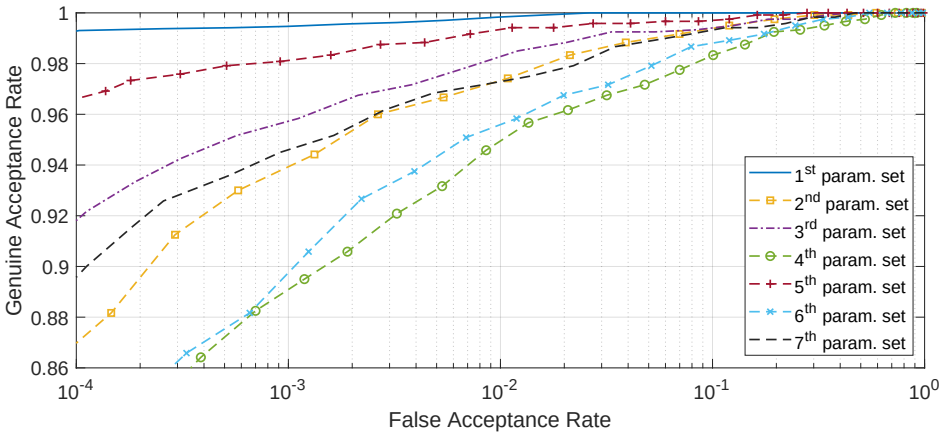


Figure 4.28: Comparative graph of several receiver operating characteristic curves generated by plotting the genuine acceptance rate against false acceptance rate obtained using different parameter configurations.

It is worth of note that to verify the effectiveness and robustness of this system, the parameters obtained from the test conducted using images acquired under 940 nm wavelength illumination have been used for the verification experiments on both the testing databases using different wavelength illumination images without parameter re-tuning. Thus, the best parameter values resulting from the simulation are:

$$\begin{cases} \lambda = 631 \times 10^{-3} \\ \mu = 295 \times 10^{-4} \\ \nu = 292 \times 10^{-3}. \end{cases} \tag{4.61}$$

To limit the computational cost of the matching process, it is important to set a priori the number of iterations after which we can consider the response of the dynamical system close enough to its steady-state condition.

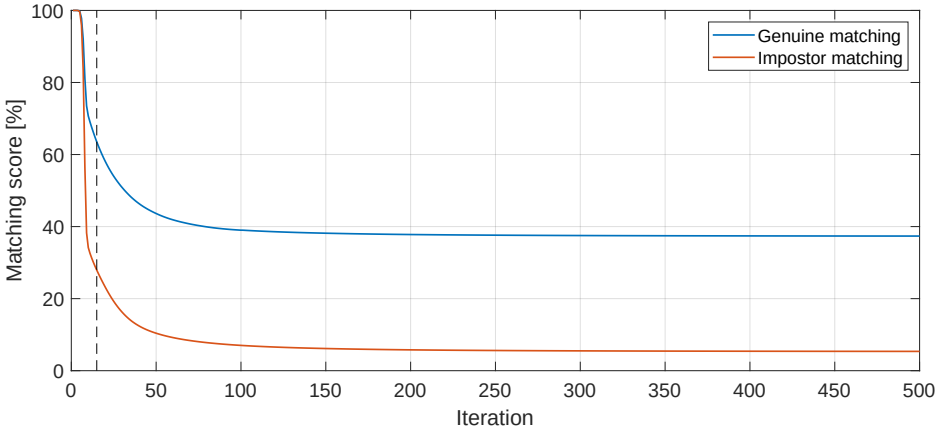
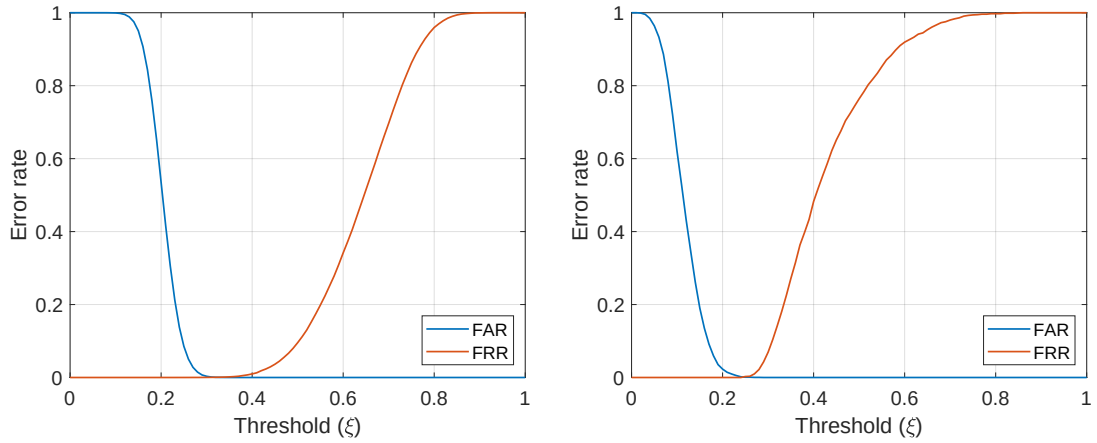


Figure 4.29: Average score against number of iterations in the case of true positive (genuine) recognition and true negative (impostor) recognition for palm vascular matching.

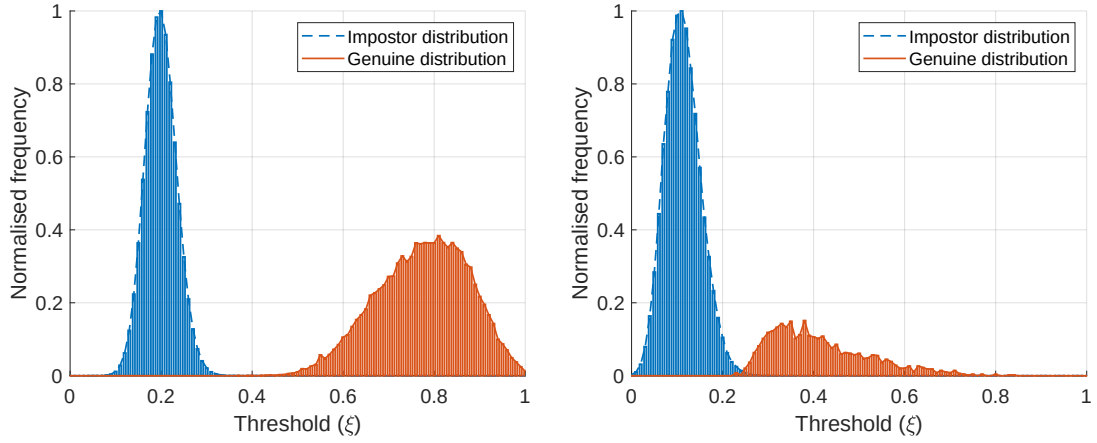
Figure 4.29 clearly shows that, within a few iterations, the achieved matching score can be considered close enough to the convergence value of the dynamical system in the steady-state condition. These results consistently suggests that this unified approach achieves significantly improved performance over the one used in the palmprint recognition system [118], ensuring also greater reliability thanks to its higher discriminating power. Hence, to save computation time, the number of iterations for the tests has been set to 15, as highlighted by the vertical dashed line in the above-mentioned figure.

Performance assessment and comparison

In order to evaluate the accuracy of the proposed authentication method based on a single-sample approach for single biometric systems, each sample in the database has undergone a one-to-one matching test against every single stored sample. Figure 4.30(a) outline the trade-off between the FRR and the FAR curves when the threshold varies, whilst the two EERs identified by the intersection point between the curves are $2.341 \cdot 10^{-5}$ for the PolyU database and $1.081 \cdot 10^{-3}$ for the CASIA database. Figure 4.30(b) instead, illustrate the genuine (intra-class) and impostor (inter-class) distributions for both the databases.
The two distributions (or classes) are clearly separated in both the databases, indicating the ability of the system to distinguish the genuine user samples from those of the impostors.



(a)



(b)

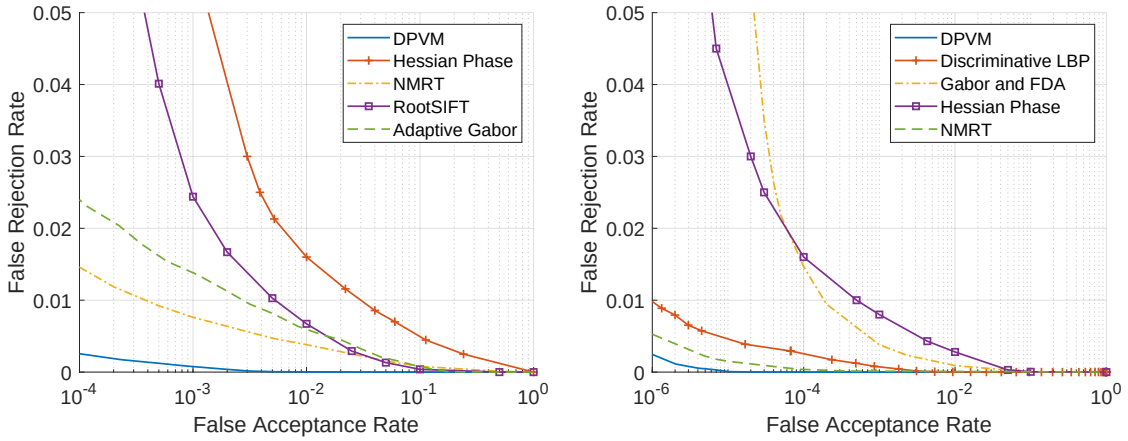
Figure 4.30: Performance assessment from PolyU (first coloumn) and CASIA (second coloumn) databases: (a) false acceptance rate and false rejection rate curves and (b) the genuine (intra-class) and impostor (inter-class) distributions.

Indeed, the separation also provides a hint on the threshold point that maximises the variance between the two classes in order to correctly mark a user sample image as authentic or impostor. To assess the performance of the proposed dynamic palm vein matching (DPVM) system with respect to several other approaches present in literature, comparison of the detection error trade-off curves has been presented in Figure 4.31, where each curve has been drawn by plotting FRR against FAR. As the FRR indicates the number of match errors, the closer the curve is to the bottom of the graph, the better the biometric performance of the system. Hence, from Figure 4.31 it is clear that the dynamic palm vein matching algorithm has achieved better performance with regard to all the other methods, obtaining a GAR_6 equal to 9.99×10^{-1} and 9.78×10^{-1} for the PolyU and CASIA databases, respectively. Table 4.8 presents a summary of the performance in terms of EER of different approaches in literature.

Table 4.8: Summary of the performance in terms of EER derived from several published methods.

		Database (DB)			Methodology		
Reference	**Year**	**Name**	**Users**	**Samples**	**Features**	**Matching**	**EER (%)**
Zhou *et al.* [160]	2011	CASIA	200	6	Neighbourhood matching Radon transform	Hamming distance	0.51
					Hessian phase	Hamming distance	1.44
		PolyU	500	12	Neighbourhood matching Radon transform	Hamming distance	0.004
					Hessian phase	Hamming distance	0.43
Sun *et al.* [136]	2012	PolyU	500	12	Curvelet transform	Hamming distance	0.66
Al-juboori *et al.* [4]	2013	PolyU	500	12	Gabor filters and Fisher discriminated analysis	Single nearest neighbours	0.233 5
Kang *et al.* [76]	2014	CASIA	100	6	Mutual foreground local binary pattern	χ^2 distance	2.53
					Mutual foreground local binary pattern	χ^2 distance and support vector machine score fusion	0.267
Hong *et al.* [77]	2015	CASIA	200	6	Root of the scale invariant feature transform	Local binary pattern-based mismatching removal	0.996
Wang *et al.* [142]	2017	PolyU	500	12	Discriminative local binary pattern	χ^2 distance	0.079
Ma *et al.* [97]	2017	CASIA	200	6	Adaptive Gabor filter	Normalised Hamming distance	0.12
Ahmad *et al.* [2]	2020	PolyU	500	12	Wave atom transform	Normalised Hamming distance	1.98
Palma *et al.* [119]	2020	CASIA	200	6	Morphological operations and Laplacian of Gaussian filter	Dynamical system	0.108
		PolyU	500	12	Morphological operations and Laplacian of Gaussian filter	Dynamical system	0.002

In particular, the proposed system has achieved a zero false acceptance rate (ZeroFAR) and a zero false rejection rate (ZeroFRR), which represent the FRR (resp. FAR) value when FAR (resp. FRR) is zero, equal to 5.57×10^{-5} and 3.03×10^{-5} for the PolyU database, and 1.96×10^{-3} and 4.27×10^{-2} for the CASIA database, respectively. Thus, these results show that the unified method outperforms all the other approaches with an EER reduced at least by 50% with respect to the listed techniques, demonstrating the effectiveness of the proposed system.



(a) (b)

Figure 4.31: Comparison of detection error trade-off curves between the proposed system and other algorithms using (a) CASIA database and (b) PolyU database.

Noise immunity

To demonstrate the robustness of the dynamic algorithm against noise, further experiments have been carried out to compare normal images and images highly corrupted by random impulse noise. The model of this noise is always independent, randomly distributed, and uncorrelated with the images and can be described as follows:

$$I_n(x,y) = \begin{cases} 2^8 - 1 & \text{with probability} \quad p\,, \\ I(x,y) & \text{with probability} \quad 1-p\,. \end{cases} \tag{4.62}$$

Furthermore, the system robustness has also been tested performing impostor matching experiments by adding the same random impulse noise to the user sample images. These experiments lead to low matching scores because there are not enough connections between the active points of both the images to be compared (i.e., half of the points in the complementary neighbourhood are not active), though the pixels affected by noise are the same. As a result, the amount of survived points after the algorithm evolution is very limited. Figure 4.32 illustrates the dynamic algorithm behaviour in presence of random impulse noise with probability p equal to 20%. In particular the first column shows the user sample images to compare, the second column shows the user sample images of the claimed identity, and the last column shows the remaining points after the algorithm evolution.

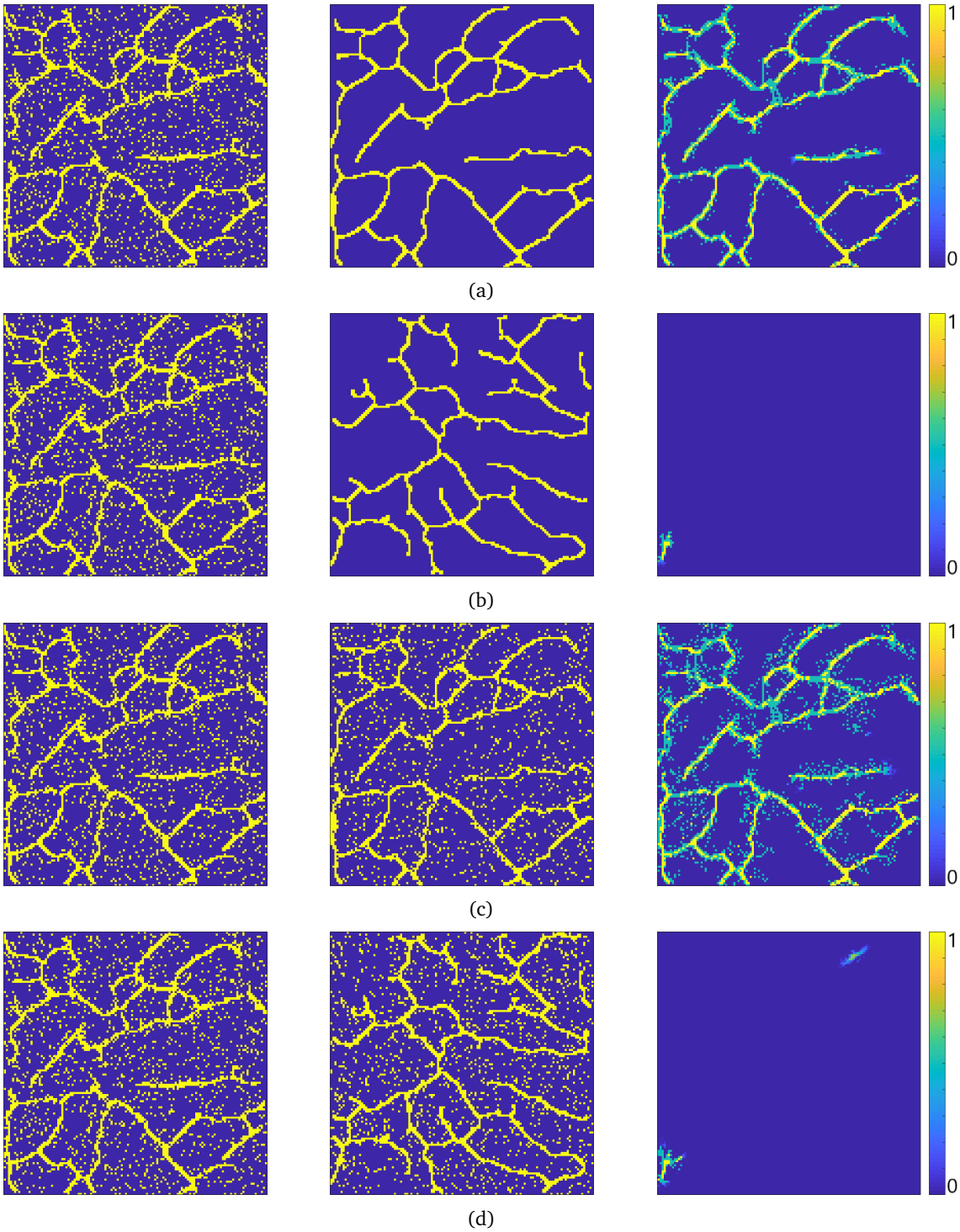


Figure 4.32: Dynamic algorithm behaviour in presence of random impulse noise with probability $p = 20\%$: a) true positive match with one image corrupted by noise, b) true negative match with one image corrupted by noise, c) true positive match with both images corrupted by noise, and d) true negative match with both images corrupted by identical noise (i.e., the same corrupted pixels).

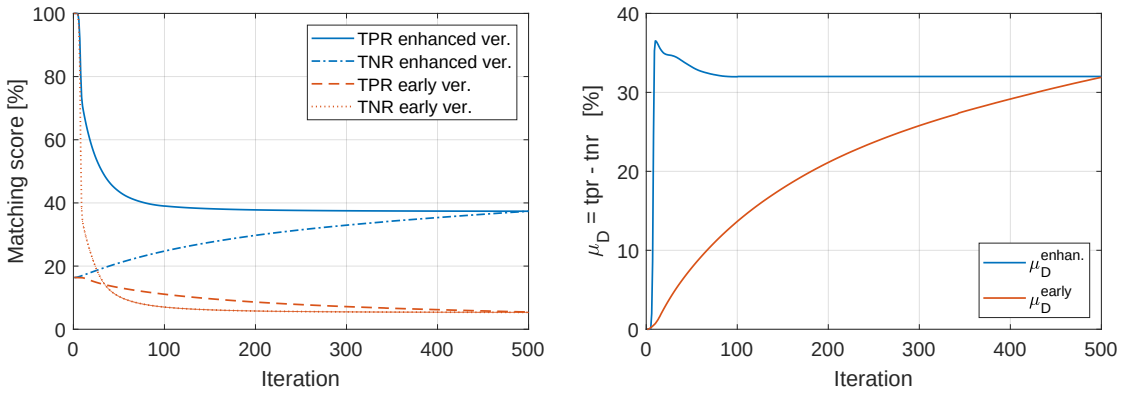
Four most significant examples have been reported, testing the following conditions: a) TP match with the user sample image corrupted by noise, b) TN match with the user sample image corrupted by noise, c) TP match with both the images corrupted by noise, and d) TN match with both the images corrupted by identical noise (i.e., the same corrupted pixels). These tests demonstrate that the system is able to recognise a subject with ease even if the samples are highly affected by noise.

Computational efficiency

The experiments have been performed making use of a virtual machine configured with two dedicated processors and 4096 MB RAM hosted on an Intel Core i5-7200U CPU (2.5 GHz) with 8192 MB RAM running a 64-bit Microsoft Windows 10 operating system. The code has been implemented using Matlab R2016b; to estimate the computation time, each part of the code has been performed 500 times, then it has been considered the mean time. As a result, the average computation times required for preprocessing, feature extraction, and matching of the proposed algorithm are 81 ms, 28 ms, 126 ms respectively. Hence, the mean response time for verification is about 0.235 s, making this approach suitable to be used in a real-time biometric authentication or identification system. The template consists of a square Boolean matrix of dimensions 128×128 pixels, hence the total size for each template is 2048 bytes. In terms of algorithm particularities and user friendliness the system is computationally simple as it only requires one image as an enrollment template, making the algorithm well suited even for systems with limited resources.

4.6 Remarks

The unified method described in Chapter 4 has been proposed in two versions, the first of which has been applied to palm line features acquired in the visible electromagnetic spectrum, whilst the enhanced version has been applied to subcutaneous palm vascular pattern in the near-infrared spectrum.



(a) (b)

Figure 4.33: Comparison between the early and the enhanced versions of the proposed approach in terms of discriminating power: (a) mean true positive and true negative rates and (b) difference between the mean values of the true positive and true negative rates.

To better illustrate the behaviour of the system in terms of accuracy with respect to the early version of the algorithm, same tests have been performed executing the algorithms with a large number of iterations, thus allowing the systems to reach the steady-state condition. Thus, the results from the experiments obtained by means of the unified approach for line-like features and its early version, have been used to graph their behaviours in terms of discriminating power.
Figure 4.33 illustrates the difference between the mean values of the true positive (TP) and true negative (TN) rates against the number of iterations, demonstrating the effectiveness of the enhanced system which has achieved significantly improved performance over the earlier version of the system ensuring also greater reliability thanks to its higher discriminating power within a very short span of time. Asymptotically, the performance of the enhanced system is at least as good as its early version, as formally proved in Section 4.2 (cf. Figure 4.33(b)).
Furthermore, the proposed unified method for line-like feature matching, which relies on a recursive algorithm based on a monotone dynamical system, can admit several extensions and applications.

II

Phytopathology

5

Background

This chapter stands as an introduction to the field of phytopathology oriented to image-based diagnosis of plant disease symptoms. It is important to present the concepts and primitives of performance metrics due to their impact on such systems. Thus, a brief overview is given to describe the main grapevine pathogenic diseases along with their properties as well as the analysis of visual symptoms used for the assessment of disease severity. Finally, the criteria for performance evaluation have been defined to determine the system accuracy and reliability which are related to the applicability in real-world deployments.

5.1 Fundamental concepts of disease in plants

Phytopathology (or plant pathology) is a science that studies plant diseases and attempts to improve the chances for survival of plants when they are faced with unfavorable environmental conditions and parasitic microorganisms that cause disease [1].
A plant is said to be healthy, or normal, when it is able to carry out its physiological functions to the best of its genetic potential. When this ability of the plant is continuously disturbed by either a pathogenic organism or an adverse environmental factor results in an abnormal physiological process that inhibits the normal activities of the plant. This interference with an essential physiological or biochemical system of the plant induces characteristic symptoms or pathological conditions. Initially, the infection is specifically confined to a few plant cells and is not visible. Soon, however, the reaction becomes widespread and affected parts of the plant develop visible or otherwise measurable adverse changes (symptoms), which reflect the amount of disease in the plant [94].
Plant diseases can be broadly classified according to the nature of their primary causal agent, either biotic (infectious) or abiotic (non-infectious). As illustrated in Figure 5.1, the range of phytopathogenic (infectious or parasitic) organisms that attack plants is diverse and includes viruses, mycoplasma, bacteria, fungi, nematodes, protozoa, and parasites, each of which has a unique mode of pathogenicity, whilst non-infectious (non-parasitic) organisms include unfavorable environmental conditions, nutrient deficiencies, disadvantageous relationships between moisture and oxygen, and the presence of toxic chemicals in air or soil [135]. Most known human and animal diseases are caused by viruses and bacteria. Conversely, plants are more commonly affected by fungi and viruses. This difference in the importance of particular groups of microorganisms can be explained partly as follows.

Bacteria generally prefer warm, alkaline conditions with high nitrogen levels. Because bacteria are unicellular their spread within the host is enhanced by a circulatory system. Filamentous fungi are more effective parasites of higher plants as their requirements are generally in direct contrast to those of bacteria [94]. A plant that has to contend with a nutritional deficiency or an imbalance between oxygen and soil moisture is often prone to infection by phytopathogens, and a plant infected by one of them is often more susceptible to invasion by secondary phytopathogens that make the disease complex, thus a plant may be affected by more than one disease-causing agent at a time. Moreover, infectious agents are capable of reproducing within or on its host and spreading from one susceptible host to another whilst, conversely, non-infectious causal agents are not organisms capable of reproducing within a host, and thus they are not transmissible, less dangerous, and are mostly avoidable. For all those reasons, the main consideration has been focused on infectious (biotic) diseases in plants that are caused by pathogenic microbes as described above.

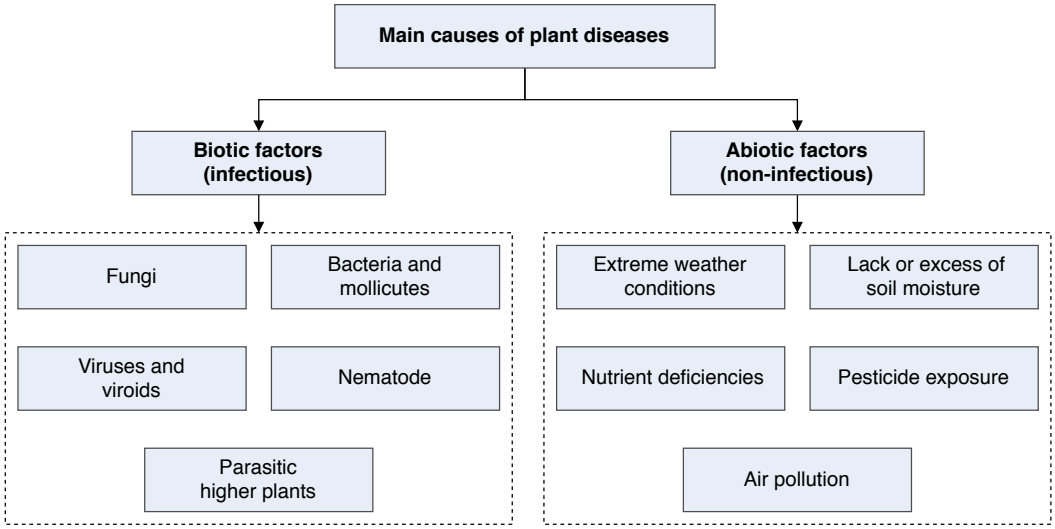


Figure 5.1: Classification of the major plant disease-causing agents, either due to biotic factors (i.e., those caused by living components such as pathogens and parasitic plants) or abiotic factors (i.e., those caused by non-living components such as nutritional deficiencies, extreme weather conditions, etc.).

Throughout this thesis, the term *pathogen* has been used to describe microbial disease agents, with particular regard to fungi, bacteria and viruses. It refers to the ability of an organism to cause disease. The allied term *pathogenesis* describes the complete process of disease development in the host, from initial infection to production of symptoms [94]. Plant phenotyping, instead, is the identification of effects on the phenotype (i.e., the plant appearance and performance) as a result of genotype differences (i.e., differences in the genetic code) and the environmental conditions to which a plant has been exposed [65, 107]. The aboveground plant phenotypes can be grouped into three categories [31]:

1. *physiological phenotypes*, which refer to traits that affect plant processes regulating growth and metabolism,
2. *structural phenotypes*, which are related to the morphological attributes of the plants,
3. *temporal phenotypes*, which consider the analysis of a sequence of images of a group of plants (belonging to different genotypes).

5.2 Diagnosis of grapevine pathogenic diseases

The occurrence of plant diseases cause severe threats to global food security and significant economic losses in yeld and quality as well as affecting agricultural industry all around the world [92]. As a consequence, an effective diagnosis of plant diseases before proper control measures can be suggested, it can give rise to a substantial improvement in product quality [14]. Common strategies to disease control are mainly oriented on spraying pesticides uniformly over cropping areas at different times during the growth cycle. Even though these methodologies can be effective, they present a negative impact in ecological and economic terms, introducing new pests and elevating resistance of the plant pathogens [106]. Traditionally, diseases detection and diagnosis has been mostly performed by human, indeed visual inspection in vineyards is still the main approach to determine if plants have already been infected presenting various symptoms, which can often be divided in: (i) underdevelopment of tissues or organs (e.g., lack of chlorophyll, leaf malformation), (ii) overdevelopment of tissues or organs, (iii) necrosis of plant parts (leaf spots, leaf blights, wilts), (iv) alternations like mosaic patterns and altered coloration in leaves. The most common way to determine if disease symptoms are present is to seek their presence on leaves, stems, or other plant parts. However, this method relies on experienced professionals performing continuous monitoring of plants, which might be time-consuming, prohibitively expensive in large vineyards, as well as prone to considerable risk of error. Plant pathogen detection conventionally relies on molecular assays, including nucleic acid-based and immunological technologies. Various approaches such as fluorescence imaging [84], immunofluorescence techniques [144], thermography [99], chain reactions [22], DNA- or RNA-based affinity biosensor [40], have been often used for quality evaluation of leaves. However, the problems with these techniques lie in the fact that are complicated, time-consuming, and constrained to centralised laboratories [92].
Alongside the remarkable development of new technologies and practices in the agriculture, the world of wine is heading towards a transformation enabling *precision agriculture (PA)* applied to vineyard performance optimisation, also known as *precision viticulture*, with the aim of maximising grape yield and overall quality whilst reducing the environmental footprint and risk. Precision viticulture thus seeks to exploit the available observations to provide recommendations to improve management efficiency in terms of quality, production, and sustainability. Therefore, consideration for new automatic and accurate along with inexpensive and efficient techniques to detect and estimate pathogenic disease symptoms is of great realistic significance [36]. As a result, digital image processing and image analysis have a vital role in biology and agricultural sectors with a great perspective especially in the plant protection field, which ultimately leads to crops management [5, 126].

5.2.1 Common grapevine pathogenic diseases

Grapevine (*Vitis vinifera*[1]) plantings are vulnerable to several different types of diseases caused by agents which are the same or very similar to those causing diseases in humans and animals. These diseases, significantly affect the growth of the grape crops and thus effective detection at an early stage may increase productivity. In the vineyard, symptoms are visually most noticeable on leaves, which may include a detectable change in color, shape, or function of the plant as it responds to the pathogen.

[1] *Vitis vinifera* belongs to order *Vitales* and its sole family, *Vitaceae* (grape family). It is a species of *Vitis*, native to many regions, including the Mediterranean, central Europe, and southwestern Asia.

Among the potential pathogens, phytoplasmas, viruses, bacteria, fungi, and nematodes can attack grapevine with different infection mechanisms and evasion strategies [7].
Common signs and symptoms that appear on the leaves of the grapevine plant, caused by a few relevant pathogenic diseases, are illustrated in Figure 5.2 and described below.

Black rot (*Guignardia bidwellii*)

Black rot of grape is probably the most serious disease of grapes where it occurs. In favorable weather, the crop may be destroyed completely, either through direct rotting of the berries or through blasting of the blossom clusters [1].
Symptoms
The symptoms of this disease begin as numerous small circular red necrotic spots on the leaf surface, and then, as the spots enlarge, they appear brown, bordered by a narrow band of dark tissue. On the upper side of the spot, within the dark border, black dot-like pycnidia[2] are rapidly formed and their dispersal through wind and rainfalls favours secondary infections.
Pathogen
The fungus *Guignardia bidwellii* (anamorph *Phyllostica ampelicida*) survives in infected vines, tendrils, fallen leaves, and mummified berries on vines or in the soil. Warm and moist climate with extended periods of rain and cloudy weather favours the development of the disease.

Leaf blight (*Pseudocercospora vitis*)

Pseudocercospora leaf spot, also called isariopsis leaf spot or leaf blight, occurs primarily after harvest, when spraying is discontinued.
Symptoms
Symptoms first appear on the lower, more shaded leaves as brown angular spots that eventually darken and become brittle. The spots are often surrounded by a clearly defined border on upper leaf surfaces and a more diffuse margin on lower leaf surfaces. Dark bristle-like fruiting bodies can appear on the leaf undersides. Premature defoliation may occur in wet seasons.
Pathogen
Pseudocercospora vitis (anamorph *Mycosphaerella personata*) is a fungal plant pathogen whose infection is favoured by high humidity and therefore the disease constitutes a potential problem for grapevine in the specific cropping areas where it is grown.

Esca (*Phaeomoniella spp.*)

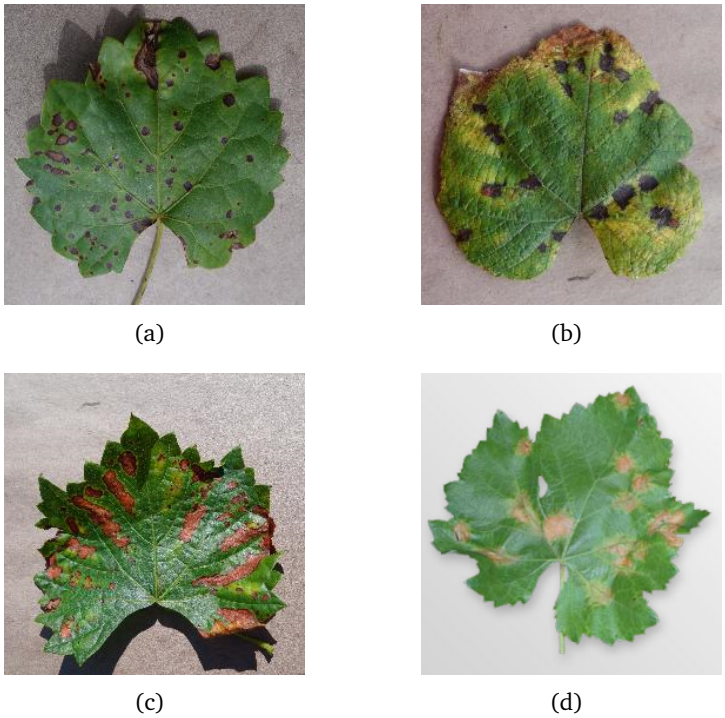
The disease develops slowly in the host until the plant exhibits a sudden decline, finally killing the grapevine within a few days [54].
Symptoms
More recent research has created a clearer understanding of the relationship between symptoms and causative pathogens of Esca. Foliar symptoms are shown to be commonly caused by the vascular fungi *Phaeomoniella chlamydosporum* and *Phaeoacremonium aleophilum*. The symptoms first appear as chlorotic spots that subsequently coalesce, turning dark red in some cultivars, and finally becoming necrotic. Often they extend to the interveinal areas of the foliar blade, leaving a narrow strip of unaffected tissue along the main veins.

[2] A pycnidium is defined as a small flask-shaped structure containing spores that occurs in certain fungi.

Pathogen
Esca, also called apoplexy or black measles, is a complex trunk disease associated with a number of different fungal species, specifically *Phaeomoniella chlamydosporum* and *Phaeoacremonium aleophilum* but also with the presence of *Botryosphaeria spp.*, *Eutypa lata*, *Phomopsis viticola*, *Cylindrocarpon* and others in infected vines [112], which obstruct the vascular system. The disease is most prevalent in areas with high summer temperatures.

Downy mildew (*Plasmopara viticola*)

Downy mildew is a serious foliage and fruit disease. Severe infections can cause losses to current season crops by reducing the sugar accumulation in fruit and the fruit yield [111].
Symptoms
The causal fungus attacks all green parts of the vine as well as fruit. Leaf lesions generally appear as irregular bright yellow spots on the upper leaf surface which gradually turn reddish brown and angular, often limited by the leaf veins, and the surface directly beneath these lesions appears almost fluffy with white mycelial growth. Severely infected leaves often drop prematurely.
Pathogen
Plasmopara viticola, an obligate biotrophic oomycete (parasite), is the causal organism of downy mildew on grape. The fungus overwinters mainly in the fallen leaves which are the source of primary infection. Secondary infection occurs by motile zoospores by splashing rain. The most serious outbreaks have been found to occur when a wet winter is followed by a wet spring and a warm summer with intermittent rains.



(a) (b) (c) (d)

Figure 5.2: Examples of lesions on grape leaf caused by various infectionus diseases: (a) Black rot (*Guignardia bidwellii*), (b) Leaf blight (*Pseudocercospora vitis*), (c) Esca (*Phaeomoniella spp.*), and (d) Downy mildew (*Plasmopara viticola*).

5.3 Criteria for performance evaluation

Plant disease symptoms can be estimated or measured in various ways that quantify the intensity, prevalence, incidence, or severity of disease. Even though disease severity is the only measure considered in this thesis, the following relate to the way in which disease symptoms are quantified [115]:

- disease intensity is a general term used to describe the amount of disease present in a population;
- disease prevalence is the proportion of fields, counties, states, etc. where the disease is detected, and reveals disease at a grander scale than incidence;
- disease incidence is the proportion of plants (or plant units, leaves, branches, etc.) diseased out of a total number assessed;
- disease severity is the area (relative or absolute) expressed as a percentage of the sampling unit (leaf, fruit, etc.) showing symptoms of disease.

The reliability and validity of a proposed scheme are determined by common measures that are used to evaluate the classification accuracy and effectiveness.
In the experimental results presented in Section 6.3.5, each disease has been treated separately leading thus to a dichotomous binary classification problem, where the labels are P (healthy) and N (diseased) and the predictions of a classifier are summarised in a 2×2 contingency table known as *confusion matrix* [123] (expanded in Table 5.1):

$$\mathbf{M} = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix} \tag{5.1}$$

which completely describes the outcome of the classification task. This contingency table may be expressed using raw counts of the number of records from class times each predicted label is associated with each actual class.
Hence, given an input leaf image $\mathfrak{X}$, the classifier is aimed at testing the null hypothesis

$$H_0 : \{\mathfrak{X} = P\} \tag{5.2}$$

that is, the test consists of the classification of the leaf as healthy or diseased.

		Predicted class		
		P	N	**Total**
Actual class	P	TP	FP (Type-I error)	TP+FP
	N	FN (Type-II error)	TN	FN+TN
	Total	TP+FN	FP+TN	

Table 5.1: Example of confusion matrix for a dichotomous binary classification problem.

As illustrated in Table 5.1, the confusion matrix reports the number of:

1. true positive (TP), the probability of correctly accepting the null hypothesis;
2. true negative (TN), the probability of correctly rejecting the null hypothesis;
3. false positive (FP), the probability of falsely rejecting the null hypothesis;
4. false negative (FN), the probability of falsely accepting the null hypothesis.

Based on the entries in the confusion matrix, the total number of correct prediction carried out by the model is $TP + TN$, whilst the number of incorrect prediction is $FP + FN$ [49]. Therefore, if

$$\mathbf{M} = \begin{bmatrix} n^+ & 0 \\ 0 & n^- \end{bmatrix} \tag{5.3}$$

where obviously $n^+ = TP + FN$ and $n^- = FP + TN$, then the classification has been perfectly done. Conversely, if the confusion matrix is as follows

$$\mathbf{M} = \begin{bmatrix} 0 & n^+ \\ n^- & 0 \end{bmatrix} \tag{5.4}$$

it represents the worst case (perfect misclassification).
Several measures have been defined to assess the quality of a prediction [26], aimed at conveying into a single figure the structure of $\mathbf{M}$. The most used functions are briefly described as follows.

Definition 17. *Precision (also known as positive predictive value (PPV)) counts the true positives, how many samples are properly classified within the same cluster (closeness of the measurements to each other)*

$$PPV = \frac{TP}{TP + FP}. \tag{5.5}$$

Definition 18. *Recall (also known as sensitivity or true positive rate (TPR)) evaluates the proportion of elements that are properly included in the same cluster*

$$TPR = \frac{TP}{TP + FN} = 1 - FNR. \tag{5.6}$$

Definition 19. *F-measure combines precision and recall in a single metric, indeed, it is the harmonic mean of precision and sensitivity and as a function of $\mathbf{M}$, has the following form:*

$$F_1 = 2\frac{PPV \cdot TPR}{PPV + TPR} = \frac{TP}{TP + \frac{1}{2}(FN + FP)} \tag{5.7}$$

where the worst case ($F_1 = 0$) is achieved for $TP = 0$, whilst the best case($F_1 = 1$) is reached for $FN = FP = 0$.

Definition 20. *Accuracy represents the ratio between the correctly predicted instances and all the instances in the dataset, whose range is between 0 (worst case) and 1 (best case):*

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}. \tag{5.8}$$

Definition 21. *Matthews correlation coefficient is the measure of the quality of binary (two-class) classifications:*

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{5.9}$$

it is a correlation coefficient between the actual and predicted binary classifications and it returns a value in the range -1 (worst case) and 1 (best case).

6

A unified method for disease severity estimation

This chapter aimed at highlighting the main idea of the proposed unified method for automatic disease severity estimation, which relies on a recursive algorithm based on a positive non-linear dynamical system whose evolution depends on the input tensor representing the leaf image to be analysed. Given an input tensor[1] template $\mathcal{X} \in \mathbb{R}_+^{n\times n\times 3}$, it may be convenient adopting an operator of the form

$$\tilde{\mathbf{X}} = \Phi\left(\mathcal{X}\right), \tag{6.1}$$

which yields a Boolean matrix $\tilde{\mathbf{X}} \in \{0,1\}^{n\times n}$ representing the affected area of the leaf image. Hence, the operator Φ must be chosen to assess the presence of a specific disease in the initial template image. To simply explain the idea, consider a simple (i.e., non dynamic) thresholding function.
A simple option to seek the presence of visible signs and symptoms of the infectious disease is to consider a thresholding-based segmentation method in order to approximate the area of the diseased leaf on the basis of the different intensities or colours in the image:

$$\tilde{\mathbf{X}} = \begin{cases} 1 & \text{if } \|\mathbf{t}_L\| \leqslant \left\|\mathbf{X}_{i,j,:}\right\| \leqslant \|\mathbf{t}_H\|, \quad \mathbf{t} \in \mathbb{R}_+^3 \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

where the threshold value is calculated according to a specific function. However, the approach based on this choice for the operator in (6.1) is clearly not noise-rejecting, since noise inhibits the localisation of the threshold value. Indeed, to avoid a misleading diagnosis, noisy pixels that are present on the leaf region of the image should not provide a positive contribution to the severity estimation.
Thus, it has been pursued an approach that rejects noise and, to this aim, it has been taken into account the following assumption.

Assumption 4. *The disease severity due to "isolated spots" is not as significant as that of cluster of points, whether they are wide "stripes" or "island", even if the number of isolated spots is very high.*

[1] A tensor is a multi-dimensional array that for the specific case it represents a three-dimensional matrix which constitutes the inpu RGB image.

6.1 The dynamical system

The idea behind the algorithm is to recursively spread the disease to fill the infected regions of the leaf only if there are symptoms of the condition itself, otherwise the leaf will not be affected by any changes.

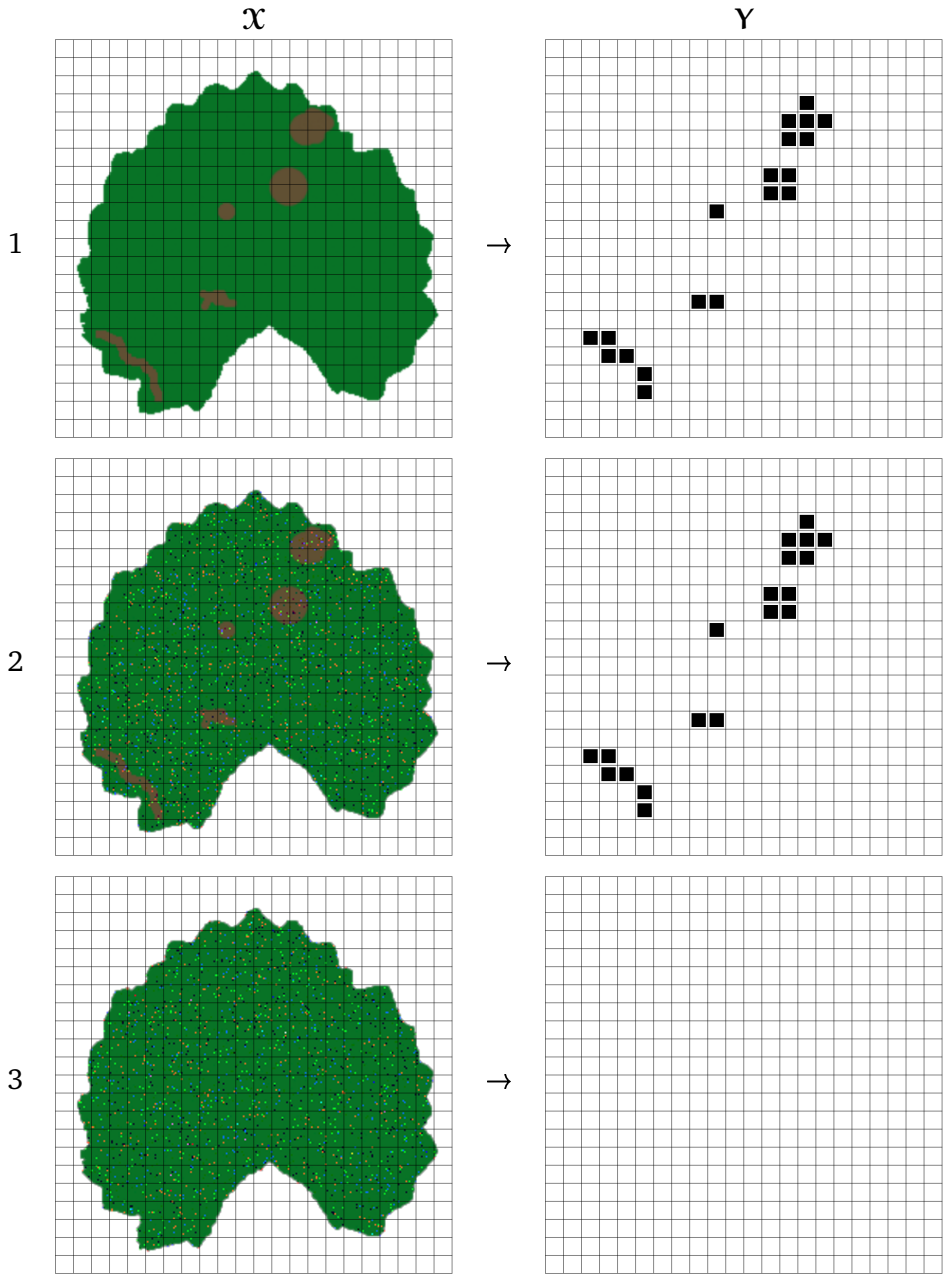


Figure 6.1: Desired behaviour in different situations (active pixels in black) where $\mathcal{X}$ is the initial image and $\mathbf{Y}$ is the desired output matrix. The first two rows examplify the correct detection when the input is a leaf image with disease symptoms upon pathogen infection (in the second row the input image has been corrupted by impulse noise), whilst the last one depicts an example of no detection when the input is a leaf image in healthy condition.

Hence, the operator Φ described in Equation (6.1) must follows the behaviour illustrated in Figure 6.1.

1. In the first example, a leaf image with disease symptoms upon pathogen infection has been provided as input. The output consists of a matrix with some sets of active pixels[2] representing the diseased regions of the leaf. Actually, almost all of the active points in the output matrix $\mathbf{Y} = \tilde{\mathbf{X}}$ should be superimposable to the visible symptoms presented in $\mathcal{X}$.
2. The second example follows the same behaviour presented in the previous example with the only exception of the input image, which has been corrupted by adding a random impulse noise with probability $p = 10\%$. The rsulting output matrix $\mathbf{Y} = \tilde{\mathbf{X}}$ should be similar to that of the previous example (i.e., the noise does not affect the accuracy of the disease severity estimation).
3. In the third example, a leaf image in healty condition has been provided as input. Inthis case, the output matrix $\mathbf{Y} = \tilde{\mathbf{X}}$ should be almost empty.

As is experimentally shown later, the behaviour illustrated in Figure 6.1 can be achieved by means of the proposed unified method that involves a positive system, whose evolution depends on the presence of disease symptoms in the input RGB image $\mathcal{X}$.

Remark 3. *In view of the iterative nature of the algorithm, it is only necessary to emphasise the presence of cluster points in diseased regions that differ in colour from those present in healthy regions.*

Assumption 5. *The image $\mathcal{X}$, provided as input, has already undergone a preliminary processing (thoroughly described in Section 6.3.2) and that therefore contains the extracted region of interest.*

Assumption 6. *It is further assumed the knowledge of the two vectors $\overline{\mathbf{x}}_d$ and $\overline{\mathbf{x}}_h$ (reported in Section 6.3.2) that represent the average colour components of the diseased and healthy conditions, respectively.*

Definition 22. *Let $\mathcal{N}_{i,j}$ be the square neighbourhood of the generic point i, j within a "radius" δ of integer amplitude grater than zero*

$$\mathcal{N}_{i,j} = \{h, l : \|h - i\| \leqslant \delta,\ \|l - j\| \leqslant \delta,\ h, l \in \mathbb{Z}\}, \tag{6.3}$$

then, given a varying (i.e., tunable) tolerance $\xi \in \mathbb{R}_{++}$ such that $\xi < 1$, the criterion used to determine whether a point $\mathbf{x} \in \mathbb{R}^3_+$ is subject to the diseased condition *is defined as follows*

$$\left\| \left[\frac{\overline{\mathbf{x}}}{\|\overline{\mathbf{x}}\|} - \frac{\overline{\mathbf{x}}_d}{\|\overline{\mathbf{x}}_d\|} \right] \right\| \leqslant \xi \tag{6.4}$$

where $\overline{\mathbf{x}}$ represents the average of the RGB component vectors in the neighbourhood $\mathcal{N}_{i,j}$ of the observed point (i, j) at time instant k

$$\overline{\mathbf{x}} = \frac{1}{n(\mathcal{N})} \sum_{h,l \in \mathcal{N}_{i,j}} \mathbf{X}_{h,l,:}(k), \tag{6.5}$$

whilst $\overline{\mathbf{x}}_d \in \mathbb{R}^3_+$ identifies the average colour components of the diseased condition.

[2] By convention an active point has been represented in black, whilst a non-active point has been represented in white.

Once the diseased condition is met, the image is processed according to the updating equation

$$\mathbf{X}_{i,j,:}(k+1) = \mathbf{X}_{i,j,:}(k) + \tau\, \rho_{i,j} \left(\overline{\mathbf{x}}_d - \mathbf{X}_{i,j,:}(k)\right) \tag{6.6}$$

where $\mathbf{X}_{i,j,:}(k) \in \mathbb{R}^3_+ \ \forall k \in \{0, 1, \ldots, K-1\}$ represents the RGB vector of the observed point at the specific time instant k, the sampling time is defined by τ, and ρ governs the speed of convergence as described in the following definition.

Definition 23. *Let $(\alpha, p) \in \mathbb{R}_{++}$ be two scalar coefficients, then the function ρ is defined as follows:*

$$\rho \doteq \frac{1}{1 + \alpha \nu^p} \tag{6.7}$$

where

$$\nu \doteq \left\| \left[\frac{\mathbf{x}}{\|\mathbf{x}\|} - \frac{\overline{\mathbf{x}}_d}{\|\overline{\mathbf{x}}_d\|} \right] \right\|. \tag{6.8}$$

Remark 4. *The function ρ represents a measure of how quickly the infectious disease can spread through the leaf and depends on the colour discrepancy between the vectors $\mathbf{x}$ and $\overline{\mathbf{x}}_d$ (i.e., how far the observed point is from the* disease condition*) measured by the Equation* (6.8)*. Furthermore, the two coefficients $(\alpha, p) \in \mathbb{R}_{++}$ need to be set appropriately in order to meet the desired rate of convergence in* (6.7)*.*

At the K-th step, to achieve a Boolean image as illustrated in Figure 6.1, all pixels that have been affected by changes are set to one (i.e., all the labeled pixels) whilst pixels that have not been modified in any way are set to zero.
The saturation function is defined as follows.

Definition 24. *Let $\tilde{\mathcal{X}}$ be the resulting real-valued tensor having the same size as $\mathcal{X}$, then the piecewise-defined function $\Theta : \mathbb{R}^3 \to \{0, 1\}$ is called saturation function and is defined as*

$$\Theta(x) \doteq \begin{cases} 1 & \text{if } x \text{ has been modified,} \\ 0 & \text{otherwise.} \end{cases} \tag{6.9}$$

Thus, through the function defined above, it is possible to generate the Boolean matrix $\tilde{\mathbf{X}}$ of dimensions $n \times n$, by means of the following computation for all i, j

$$\tilde{X}_{i,j} = \Theta\left(\mathbf{X}_{i,j,:}\right). \tag{6.10}$$

Disease severity (DS) is the area (relative or absolute) of the sampling unit (leaf) showing symptoms of disease and is most often expressed as a percentage or proportion [14, 115] of affected leaf area. Hence, the final score is performed on the number of pixels with value 1 (active), which is compared to the total amount of points within the leaf area, as described in the following definition.

Remark 5. *Given the Assumption* (5)*, it is very easy to compute the leaf area from the segmented RGB image $\mathcal{X} \in \mathbb{R}^{n \times n \times 3}_+$. Indeed, using the Boolean matrix $\mathbf{S} \in \{0, 1\}^{n \times n}$ representing the mask used to remove the background part from each channel of the initial RGB image, the leaf area can be calculated as*

$$A = \sum_{i,j} S_{i,j}. \tag{6.11}$$

Definition 25. *Let* $\mathbf{S} \in \{0,1\}^{n\times n}$ *be the Boolean matrix described above and* $\tilde{\mathbf{X}} \in \{0,1\}^{n\times n}$ *be the output Boolean matrix. Denoting by* $\Sigma(\mathbf{S})$ *and* $\Sigma(\tilde{\mathbf{X}})$ *the number of active points in the respectively matrices, then the* disease severity *is defined as*

$$DS\,(\mathcal{X}) = \frac{\Sigma(\tilde{\mathbf{X}})}{\Sigma(\mathbf{S})}. \tag{6.12}$$

Hence, the resulting algorithm works as follows.

Algorithm Disease severity estimation

Input: leaf image $\mathcal{X}$ in RGB color space.
Parameters: Number of steps K, positive real constants $\xi < 1$, τ, α, and p, vectors $\overline{x}_d$ and $\overline{x}_h$, integer neighbourhood amplitude $\delta > 0$ (which implies the size of the set $\mathcal{N}$).
Outputs: Disease severity estimation DS $(\mathcal{X})$.

1. Set the initial condition $\mathcal{X}(0) := \mathcal{X}$
2. **for** $k = 0, k < K, k = k + 1$
   for each point (i, j) belonging to the leaf area **do**
   compute the average of the RGB component vectors in the neighbourhood $\mathcal{N}_{i,j}$ according to (6.5)

$$\overline{x} = \frac{1}{n(\mathcal{N})} \sum_{h,l\in\mathcal{N}_{i,j}} \mathbf{X}_{h,l,:}(k)$$

   if the Criterion (6.4) $\left\| \left[\frac{\overline{x}}{\|\overline{x}\|} - \frac{\overline{x}_d}{\|\overline{x}_d\|} \right] \right\| \leqslant \xi$ is satisfied **then**

   compute the speed of spreading disease according to (6.8)–(6.7)

$$\nu = \left\| \left[\frac{x}{\|x\|} - \frac{\overline{x}_d}{\|\overline{x}_d\|} \right] \right\|$$

$$\rho = \frac{1}{1 + \alpha\nu^p}$$

   compute the updated value according to (6.6)

$$\mathbf{X}_{i,j,:}(k+1) = \mathbf{X}_{i,j,:}(k) + \tau\, \rho_{i,j} \left(\overline{x}_d - \mathbf{X}_{i,j,:}(k) \right)$$

   end if
   end for
   end for
3. Convert the real-valued matrix to Boolean by means of the operator defined in (6.9):
   for each point (i, j) **do**

$$\tilde{X}_{i,j} = \Theta\left(\mathbf{X}_{i,j,:}\right)$$

   end for
4. Compute the disease severity as in (6.12) DS $(\mathcal{X}) = \Sigma(\tilde{\mathbf{X}})/\Sigma(\mathbf{S})$, where $\mathbf{S}$ represents the mask used to remove the background part from each channel of the initial RGB image $\mathcal{X}$.

The rationale of step 2 (the core of the procedure) is the following. Let $\mathcal{X}$ be the input RGB image and assume that it represents a diseased leaf, hence in correspondence of a visible symptom, the Criterion (6.4) would be satisfied if the discrepancy between the normalised vectors $\mathbf{x}$ and $\overline{\mathbf{x}}_{\mathrm{d}}$ is less than or equal to the tolerance ξ. Figure 6.2, illustrates the maximal closed cone containing all accepted vectors in the normalised RGB colour space. Actually, this is valid not only in correspondence of a visible symptom, but also in the nearest proximity of a visible symptom due to the mean filtering described in Equation (6.5), which reduces the amount of intensity variation between neighbouring points depending on the size of the square neighbourhood $\mathcal{N}$ defined through an integer radius $\delta > 0$. Note that, even though the average of the RGB component vectors in the neighbourhood has been considered to test the diseased condition criterion, no point in the image has been replaced by this value, thus preserving image details.

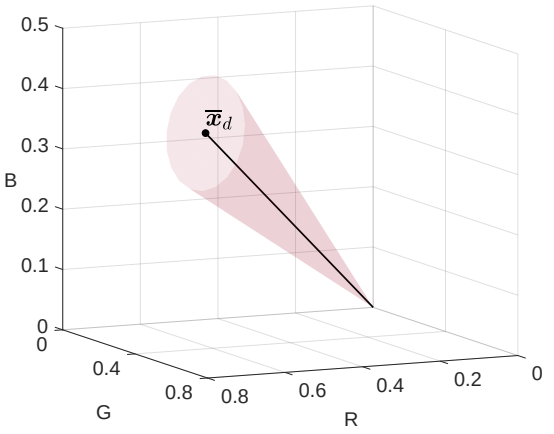


Figure 6.2: Maximal closed cone defined by the vector $\overline{\mathbf{x}}_{\mathrm{d}}$ and a tolerance $\xi = 0.1$.

Then, assuming that the Criterion (6.4) is satisfied, the speed of spreding disease can be computed by means of the Equations (6.8)–(6.7). In particular the Equation (6.8) measures the colour discrepancy between the observed point $\mathbf{x}$ (not to be confused with $\overline{\mathbf{x}}$, which is the average of the RGB component vectors in the specific neighbourhood of the observed point) and $\overline{\mathbf{x}}_{\mathrm{d}}$, which represents the colour of the *disease condition*. Hence, through the Equation (6.7) it is possible to calculate the function ρ that affect the extent to which the observed three-dimensional vector $\mathbf{x}$ converges to that representing the *disease condition* $\overline{\mathbf{x}}_{\mathrm{d}}$.

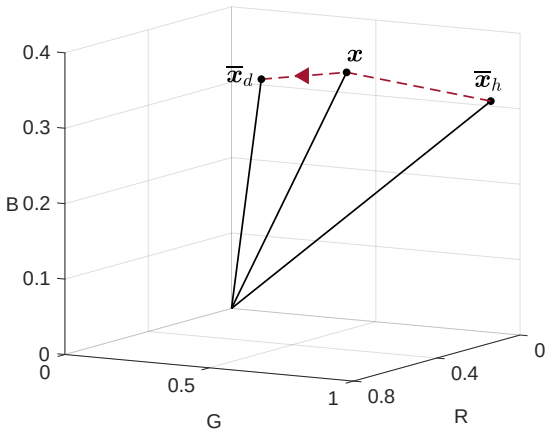
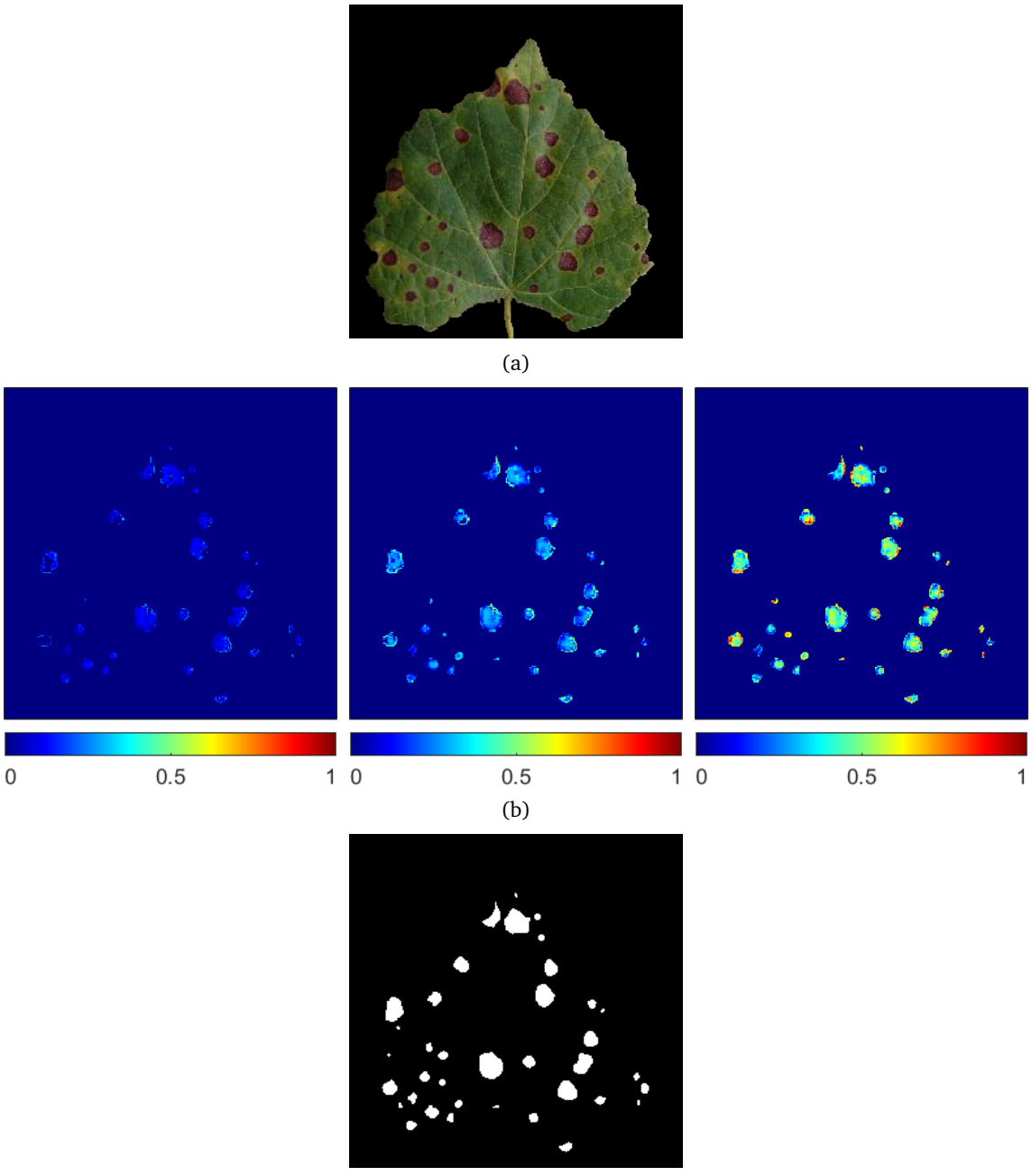


Figure 6.3: Representation of the vectors $\overline{\mathbf{x}}_{\mathrm{d}}$, $\mathbf{x}$, and $\overline{\mathbf{x}}_{\mathrm{h}}$ in the normalised RGB colour space.

Note that, in view of the two aforementioned equations, the greater the difference between the RGB vectors of the observed point and the *healthy condition*, the faster the convergence of the observed point to the *disease condition*. Indeed, this behaviour is ensured by the dynamic Equation (6.6), which enables the vector $\mathbf{x}$ to asymptotically converge towards $\overline{\mathbf{x}}_{\mathrm{d}}$, usually in few iterations. The depiction in Figure 6.3 illustrates this important aspect showing the three-dimensional vectors $\overline{\mathbf{x}}_{\mathrm{d}}$, $\mathbf{x}$, and $\overline{\mathbf{x}}_{\mathrm{h}}$ in the normalised RGB colour space as well as the direction towards which the vector $\mathbf{x}$ converges. Once the system has reached the steady-state condition (i.e., the condition in which the state variable is constant in spite of ongoing procedures that strive to change it) it is possible to estimate the disease severity through the Equations (6.10)–(6.12) as reported in steps 3–4, respectively.

To exemplify the algorithm behaviour, suppose that the input image $\mathcal{X} \in \mathbb{R}_{+}^{n \times n \times 3}$ is that shown in Figure 6.4(a). Then, the result of the dynamic algorithm after the transformation to Boolean by means of the operator defined in (6.1), which leads to the matrix $\tilde{\mathbf{X}} \in \{0,1\}^{n \times n}$, is shown in Figure 6.4(c). In this example, the system has estimated a disease severity equal to $\mathrm{DS}(\mathcal{X}) = \Sigma(\tilde{\mathbf{X}})/\Sigma(\mathbf{S}) = 0.0768$.



(a)

(b)

(c)

Figure 6.4: Dynamic algorithm behaviour: (a) input image representing a real leaf affected by *Black Rot* disease, (b) the resulting image from the application of the dynamic algorithm (representation of the pixels modified by the iterative procedure during the transient state until the steady-state condition has been reached), and (c) final result after binarisation.

6.2 Properties of the system

In this section has been analysed the algorithm based on the recursive Equation (6.6), to better understand its behaviour. First of all, the positive parameters α and p in the procedure need to be tuned to define appropriately how the function ρ should behave. Let $\boldsymbol{v}_1, \boldsymbol{v}_2 \in \mathbb{R}^3_+$ be two vectors to be compared, then it is possible to determine a metric defined on the normalised RGB colour space in order to find out the maximum differences along any coordinate dimension between two vectors (i.e., the maximum distance).
Figure 6.5 illustrates a plot of two vectors in the normalised RGB colour space

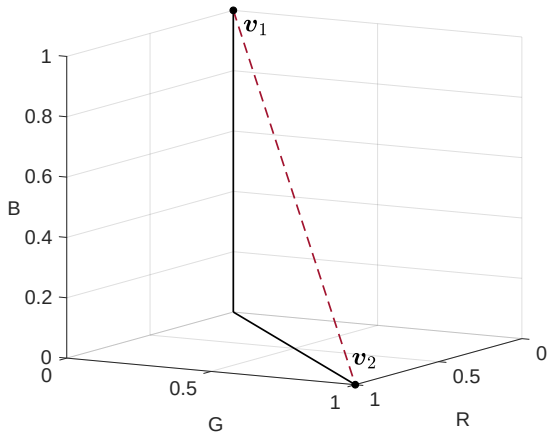


Figure 6.5: Representation of the vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ in the normalised RGB colour space.

$$\boldsymbol{v}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \boldsymbol{v}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

where the Euclidean distance between those points is the length of the dashed, straight line between the two points.

Definition 26. *Given a vector space $\mathbb{X}$, for $\mathbf{x} \in \mathbb{X}$ a non-negative function $\|\mathbf{x}\|$, which represents the norm of the vector $\mathbf{x}$, is defined such that*

- $\|\mathbf{x}\| \geqslant 0$ *and* $\|\mathbf{x}\| = 0 \iff \mathbf{x} = \mathbf{0}$ *(being positive definite),*
- $\|\lambda\mathbf{x}\| = |\lambda|\|\mathbf{x}\|$ *for any scalar* λ *(being absolutely scalable),*
- $\|\mathbf{x} + \mathbf{y}\| \leqslant \|\mathbf{x}\| + \|\mathbf{y}\|$ *(satisfying the triangle inequality).*

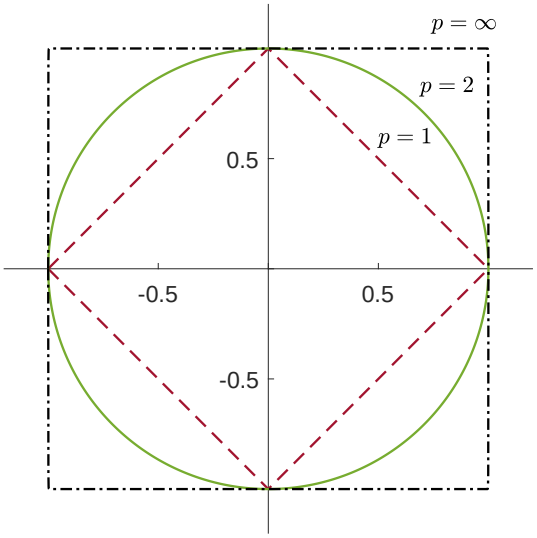


Figure 6.6: Illustrations of unit circles in different L^p norms (every vector from the origin to the unit circle has a length of one).

Let $p \geqslant 1$ be a real number, then the L^p norm of a vector $\mathbf{x} = (x_1, x_2, \ldots x_n)$ is defined as

$$\|\boldsymbol{x}\|_p \doteq \sqrt[p]{\left(\sum_{i=1}^{n} |x|^p\right)}. \tag{6.13}$$

As illustrated in the figure on the left, for $p = 1$ we get the L^1 norm or Manhattan distance*, where the distance between two points is the sum of the absolute differences of their Cartesian coordinates, whilst for $p = 2$ we get the L^2 norm or* Euclidean distance*, where the distance between two points is the length of a line segment between the two points, and as p approaches ∞ we get the L^∞ norm or* maximum norm *which is equivalent to*

$$\|\boldsymbol{x}\|_\infty \doteq \max_i \{|x_i|\}.$$

Since the normalised RGB colour space is described by treating the component values as ordinary Cartesian coordinates in a Euclidean space that represents a cube of non-negative values such that $x_i \in \mathbb{R}_+ : x_i \leqslant 1$ for $i = \{1, 2, 3\}$, it is convenient to consider a L^2 norm. Hence, the real-valued function ν (6.8) is positive and bounded above $\sqrt{3}$.

Remark 6. *Let $\mathbf{u}, \boldsymbol{\nu} \in \mathbb{R}^3_+$ be two unit vectors to be compared in the normalised RGB colour space. Then, ν is non-negative being a norm and equal to 0 if the two vectors to be compared are identical, that is, $u_i = \nu_i$ for $i = \{1, 2, 3\}$. Conversely, if $\mathbf{u} \neq \boldsymbol{\nu}$ the distance (L^2 norm) between $\mathbf{u}$ and $\boldsymbol{\nu}$ is*

$$\|\mathbf{u} - \boldsymbol{\nu}\| = \sqrt{(u_1 - \nu_1)^2 + (u_2 - \nu_2)^2 + (u_3 - \nu_3)^2} \leqslant \sqrt{3}.$$

The upper bound $\sqrt{3}$ is a maximum: it can be reached if we take two complementary vectors, for instance: $\mathbf{u} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^\top$ and $\boldsymbol{\nu} = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}^\top$.

Property of convergence/divergence. *Let us now consider just a point of a segmented RGB image $\mathcal{X} \in \mathbb{R}^{n \times n \times 3}_+$ representing a symptomatic leaf. Then the presence of diseased regions in the image give rise to a monotone system, as described next. Let us group in a vector $\mathbf{x}(k) \in \mathbb{R}^3$ the RGB component values $X_{i,j,:}$ of the (diseased) point (i, j) in the image $\mathcal{X}$. Then, the system evolves as follows:*

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \tau\rho\left(\overline{\mathbf{x}}_d - \mathbf{x}(k)\right) \tag{6.14}$$

where

$$\rho = \frac{1}{1 + \alpha\nu^p}$$

and $\overline{\mathbf{x}}_d$ identifies the average colour components of the diseased condition.

As an example, consider the situation illustrated in Figure 6.7. Then, the vector $\mathbf{x}$ might be attracted towards the vector $\overline{\mathbf{x}}_d$, and as such the dynamics of the system might lead to the filling of all diseased regions by acting on all the vectors that satisfy the criterion expressed in the Equation (6.4).

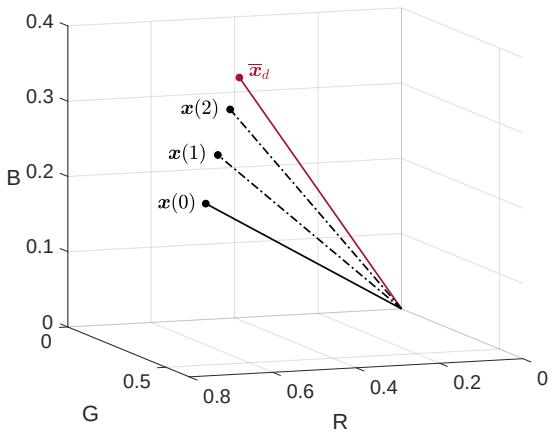


Figure 6.7: Dynamic algorithm example: a few steps of the dynamical system evolution.

Assumption 7. *The parameter τ representing the sampling time is chosen so as to guarantee that $1-\tau\rho \geqslant 0$, $\forall\, \rho \in \mathbb{R}_+ : \rho \leqslant 1$.*

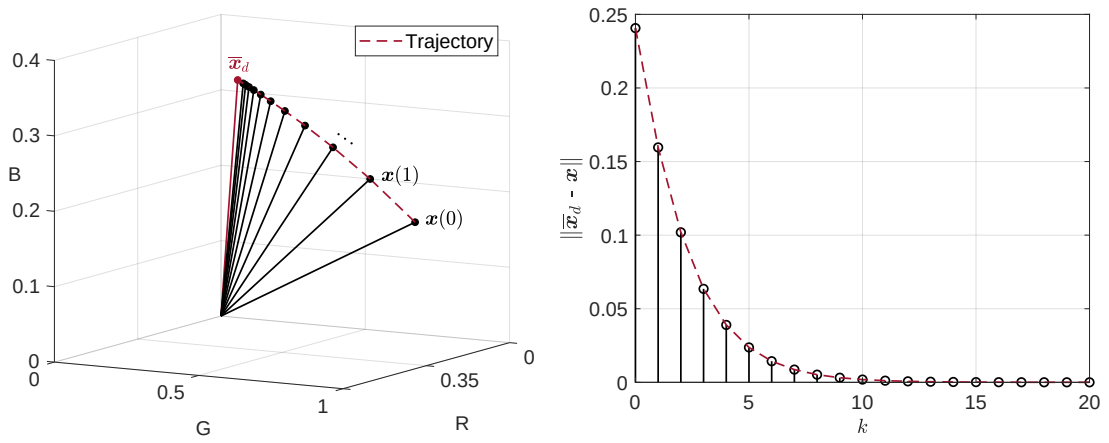
Definition 27. *If $\mathbf{A} \in \mathbb{S}^n$ with all the entries outside the main diagonal are all zero, i.e., $A_{i,j} = 0$ whenever $i \neq j$, then $\mathbf{A}$ is called diagonal.*

Let us consider the previous updating Equation (6.14) in the following form

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1-\tau\rho & 0 & 0 \\ 0 & 1-\tau\rho & 0 \\ 0 & 0 & 1-\tau\rho \end{bmatrix}}_{\text{matrix } \mathbf{P}} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \tau\rho \begin{bmatrix} \overline{x}_{d1} \\ \overline{x}_{d2} \\ \overline{x}_{d3} \end{bmatrix}. \tag{6.15}$$

Hence, it can be easily seen that the non-negative matrix $\mathbf{P} \succeq \mathbf{O}$ appearing in the equation above is positive semidefinite and diagonal, which implies that the matrix is also symmetric (i.e., $\mathbf{P} = \mathbf{P}^\top$). Therefore, $\mathbf{P}$ is obviously a scalar matrix[3] which can be viewed as a scalar multiple of an identity matrix, that is, $(1-\tau\rho)\,\mathbf{I}_3$. Note that multiplication by the identity matrix is equivalent to (scalar) multiplication by 1, and that multiplication by a scalar matrix $(1-\tau\rho)\,\mathbf{I}_3$ is equivalent to multiplication by the scalar $(1-\tau\rho)$ [57].
Moreover, since the matrix $\mathbf{P}$ has non-negative off-diagonal entries $P_{i,j} \geqslant 0$ $(\forall i \neq j)$, it is also a Metzler matrix (i.e., $\mathbf{P} \in \mathbb{M}$). If the Assumption (7) holds, then the matrix $\mathbf{P} \in \mathbb{S}^3_{++}$ is called Schur stable[4], since all its eigenvalues lie inside the unit circle, or equivalently its spectral radius ρ (i.e., the eigenvalue with maximum modulus) is non-negative, real, and equal to $\rho(\mathbf{P}) = 1-\tau\rho$.

Remark 7. *Given the Assumption* (7)*, the term $1-\tau\rho$ serves to inhibit potential instability of the system because as $\mathbf{x}$ approaches $\overline{\mathbf{x}}_d$, $1-\tau\rho$ approaches* 0*, ensuring thus a unique steady state that is globally asymptotically stable (monostability).*



(a) (b)

Figure 6.8: Transient behaviour: (a) a vector $\mathbf{x}$ (healthy) converges in norm to $\overline{\mathbf{x}}_d$ (diseased) and (b) distance between the two vectors represented by the norm of their difference.

[3] Scalar matrices are a special class of diagonal matrices whose elements along the diagonal are all the same.
[4] This property is mostly referred as *convergence of matrices* [64].

6.3 Example: automatic detection and severity estimation of grape diseases for enhanced precision agriculture

Effective and accurate plant disease detection and diagnosis at an early stage are essential in plant production, as they can lead to a reduction in both qualitative and quantitative losses in crop yield. Pathologists usually focus on pathogenic diseases appearing in various parts of the plant like roots, stem, and leaf, however, as discussed earlier, the present work concentrates particularly on leaves. Indeed, on this part of the plant, a large amount of information is available allowing an effective diagnosis [8, 53]. Major advances and technological developments in recent decades have allowed image analysis-based systems to automatically detect the symptoms that appear on the leaves and stem of a plant, thus helping in the cultivation of healthy plants and improving their quality [104]. These kind of systems may be trated as pattern recognition systems that automate the detection and severity estimation of pathogenic diseases through the analysis of digital images following a common scheme. In this regard, a large number of researches have been reported in the literature that employed machine learning-based techniques for plant disease detection [125]. This approach can aid typical steps of image analysis including segmentation and feature extraction, which are fundamentals to determine the applicability of a machine learning model which detection and severity estimation are generally based on [109]. Deep learning algorithms, such as methods based on convolutional neural network (CNN), allow to automatically extract the features directly from the input images bypassing the segmentation and feature extraction steps as well as providing more accurate results compared with traditional methods [121]. These approaches are remarkably powerful for solving classification problems but not all problems can be represented in this form. The major drawback is to choose the right (very large) set of data to train the models, but in practical terms that is not always possible. Moreover, it is extremely expensive to train due to complex data models and the strategies learnt by deep learning may be more superficial than they appear [101]. In this thesis, to identify potential plant pathogens as quickly as possible to reduce the likelihood of an infection spreading, it has been proposed a novel approach based on the dynamical systems theory that allows the detection and severity estimation of grape diseases regardless of disease type, even in adverse conditions.

6.3.1 Dataset used in simulation

The dataset used to assess the performance of the proposed system, is based on the unmodified colour version of grape leaf images in the PlantVillage dataset [68], which is worldwide shared for research purposes and whose details are given in Table 6.1. The dataset consists of images of single leaves removed from their plants with inoculated or naturally occurring disease. The subset of grape images contains 4 063 samples, of which 3 640 are diseased leaf images exhibiting three different conditions and 423 healthy leaves, with their conditions classified by expert plant pathologists by means of standard phenotyping approaches. Hence, only expertly identified leaves are present in the dataset. Leaf images have been captured through a twenty-megapixel camera (Sony DSC - Rx100/13 20.2 Mpx) using the automatic mode and collecting from four to seven different orientations to compensate for directional lighting variation. Indeed, all the images have been taken outside under nautral light in several different conditions (e.g., sunny, mostly/partly sunny, cloudy, and mostly/partly cloudy). The version of the dataset used in this study has been scaled down to 256×256 pixels and rotations of the same leaf have been removed [147].

Table 6.1: List of crops and their disease status in the PlantVillage database.

	Diseased					
	Fungi	**Bacteria**	**Mold**	**Virus**	**Mite**	**Healthy**
Apple (3 172)	*Gymnosporangium juniperi-virginianae* (276) *Venturia insequalis* (630) *Botryospaeria obtuse* (621)					(1 645)
Blueberry (1 502)						(1 502)
Cherry (1 906)	*Podosphaera spp.* (1052)					(854)
Corn ()3 852)	*Cercospora zeae-maydis* (513) *Puccinia sorghi* (1 192) *Exserohilum turcicum* (985)					(1 162)
Grape (4 063)	*Guignardia bidwellii* (1 180) *Phaeomoniella spp.* (1 384) *Pseudocercospora vitis* (1 076)					(423)

Continued on next page

Continuation of Table 6.1

	Diseased					
	Fungi	**Bacteria**	**Mold**	**Virus**	**Mite**	**Healthy**
Orange (5 507)		*Candidatus Liberibacter* (5 507)				
Peach (2 657)		*Xanthomonas campestris* (2 291)				(360)
Bell Pepper (2 475)		*Xanthomonas campestris* (997)				(1 478)
Potato (2 152)	*Alternaria solani* (1 000)		*Phytophthora Infestans* (1 000)			(152)
Raspberry (371)						(371)
Soybean (5 090)						(5 090)
Squash (1 835)	*Erysiphe cichoracearum* (1 835)					
Strawberry (1 565)	*Diplocarpon earlianum* (1 109)					(456)
Tomato (18 162)	*Alternaria solani* (1 000) *Septoria lycopersici* (1 771) *Corynespora cassiicola* (1 404) *Fulvia fulva* (952)	*Xanthomonas campestris pv. Vesicatoria* (2 127)	*Phytophthora Infestans* (1 910)	*Tomato Yello Leaf Curl Virus* (5 357) *Tomato Mosaic Virus* (373)	*Tetranychus urticae* (1 676)	(1 592)

6.3.2 Methodology

Figure 6.9 illustrates the overall block diagram of the proposed disease severity estimation system that follows a typical pattern recognition architecture. Hence, the main steps involved in such a system are: image acquisition of the grape leaf, preprocessing of the raw image, feature extraction, and disease detection through the dynamical system.

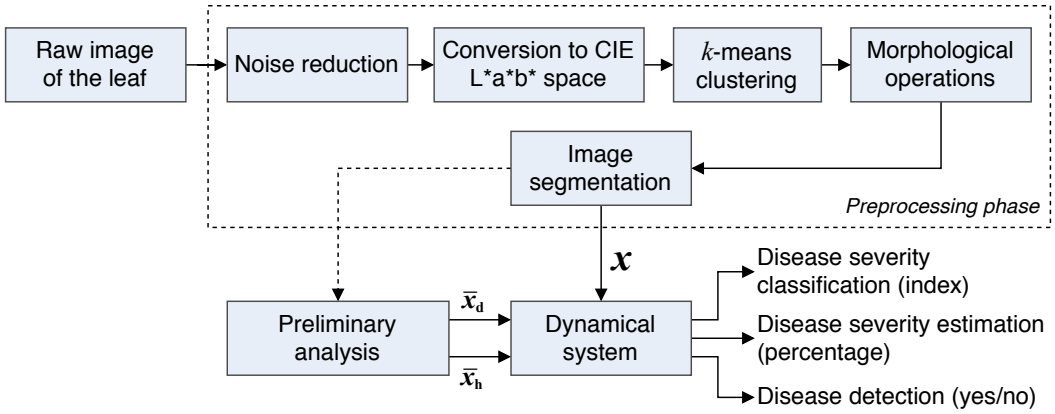


Figure 6.9: Block diagram of the proposed system.

6.3.3 Image acquisition

The primary source of data used in this thesis have been images from the unmodified colour version of grape leaves in the PlantVillage dataset [68], where all the images have been taken outside under natural light with a standard point and shoot digital camera in automatic mode, as described in Section 6.3.1. Data from visible light bands (the range of wavelengths humans can perceive) are composited in their respective red, green, and blue channels. The established wavelength range of visible light is approximately 380-740 nm [134], which equates to a frequency range of 789-405 THz, respectively, as illustrated in Figure 6.10. This range of wavelengths, frequencies, and energies manifests as a range of colours in the visual systems of humans [48]. The various colour ranges indicated in Figure 6.10(b) are an approximation: indeed, the spectrum is continuous with no clear boundaries between one colour and the next [21]. Figure 6.11 illustrates an example of leaf affected by Black rot (*Guignardia bidwellii*) disease along with the grey-scale representation of every channel, each of which corresponds to a range of wavelengths and contains different spectroscopic information.

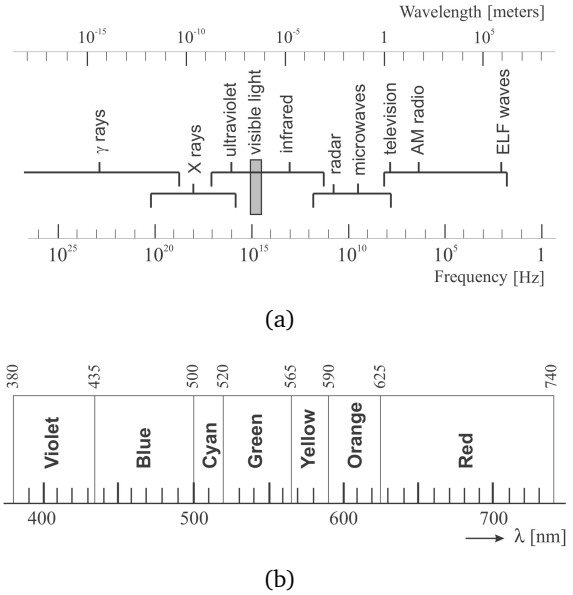


Figure 6.10: Photon wavelength and frequency of the electromagnetic spectrum [134]: (a) electromagnetic spectrum and (b) wavelengths of the visible spectrum.

(a)

(b)

(c)

(d)

Figure 6.11: Example of RGB image representing a grape leaf affected by Black rot disease: (a) original RGB image, (b) red channel (625 nm $\leqslant \lambda_R \leqslant$ 740 nm), (c) green channel (520 nm $\leqslant \lambda_G \leqslant$ 565 nm), and (d) blue channel (435 nm $\leqslant \lambda_B \leqslant$ 500 nm).

Preprocessing

The preprocessing elaboration is required to extract the region of interest from the input image removing the background. As outlined in the Figure 6.9, the major steps involved in the preprocessing of raw images are:

1. noise reduction by means of a non-linear spatial filter,
2. conversion to CIE L*a*b* space and normalisation
3. classification of background and foreground
4. ROI extraction.

Noise reduction As described in Section 4.3, noise can be systematically introduced into images during acquisition and transmission. [51, 96]. This kind of noise is independent, randomly distributed, and uncorrelated with the image, since it is charactesized by replacing a portion of the original pixel values of the image with intensity values drawn from some distribution, usually a uniform distribution over the intensity range. Hence, a common order-statistic (non-linear) spatial filter, i.e., median filter, can be used to remove unwanted information from noisy palmprint images preserving details. In our experiments, the kernel size has been set to 5×5 pixels.

Conversion to CIE L*a*b* space and normalisation Firstly, the RGB image is converted into the CIE L*a*b* colour space (see Appendix B for the details about the transformation) which consists of a three-dimensional model able to decoupling the lightness of the colour and the chromaticity components a* and b* which represent the red/green channel and yellow/blue channel, respectively. On the colour space diagram illustrated in Figure 6.12, L* is represented on a vertical axis with values from 0 (total absorption) to 100 (diffuse white), whilst the centre of the sphere is neutral or achromatic. Changing from Cartesian coordinates to polar coordinates we get the CIELCh colour space, where the distance from the central axis represents the chroma (C*), or relative saturation of the colour, whilst the angle on the chromaticity axes (angle of the hue in the CIELAB color plane) represents the hue (h^0). The lightness L* remains unchanged. The conversion of the chromaticity components a* and b* (Cartesian coordinates) to C* and h^0 (polar coordinates) is performed as follows:

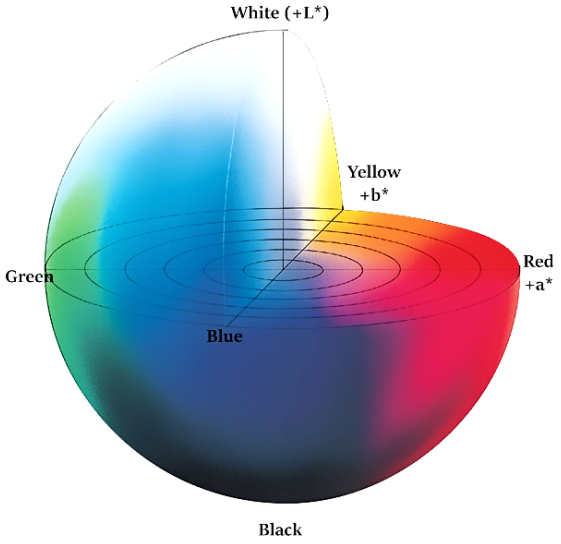


Figure 6.12: The CIE L*a*b* colour space [82].

$$C^* = \sqrt{(a^*)^2 + (b^*)^2}, \quad h^0 = \arctan\left(\frac{b^*}{a^*}\right). \tag{6.16}$$

Conversely, the conversion from polar to Cartesian coordinates is given by:

$$a^* = C^* \cos\left(h^0\right), \quad b^* = C^* \sin\left(h^0\right). \tag{6.17}$$

(a) (b) (c)

Figure 6.13: Example of RGB image representing a grape leaf affected by Black rot disease: (a) L* channel, (b) a* channel, and (c) b* channel.

Classification of background and foreground The goal of this step is that to classify the background and foreground in the a*b* two-dimensional data space using the k-means clustering in order to correctly identify the leaf. This method is a form of unsupervised learning whereby a set of data points is partitioned into clusters of patterns so that the similarity measure between any pair of observations assigned to each cluster minimises an objective function, turning the clustering task into an optimisation problem[5].
Thus, given a set $\{\mathbf{x}_i\}_{i=1}^{N}$ of multidimensional data to be partitioned into K clusters, where $K < n$, it is possible to iteratively find the closest centroid in the set $\{\mathbf{c}_i\}_{i=1}^{K}$ minimising the following objective function:

$$J\left(\mathbf{x}_j, \mathbf{c}_i\right) = \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in \mathbf{C}_i} \left\|\mathbf{x}_j - \mathbf{c}_i\right\|^2 \tag{6.18}$$

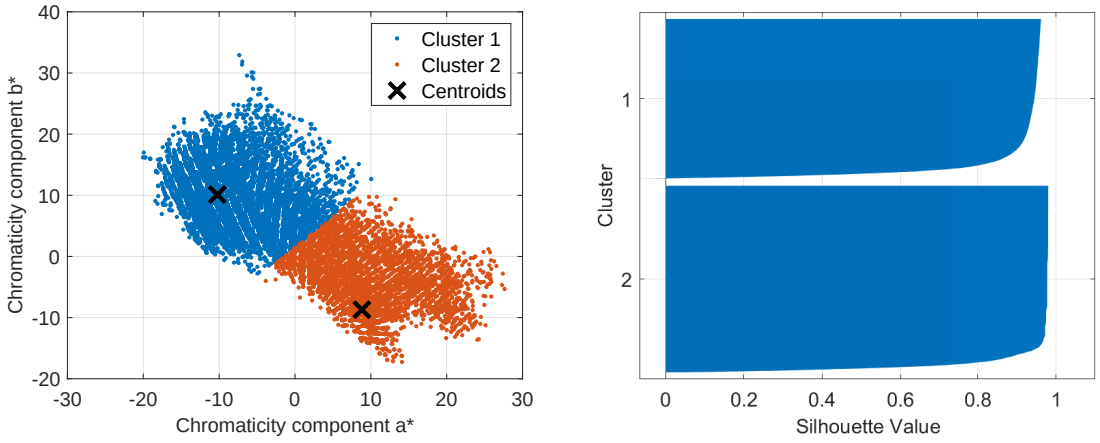
where $\left\|\mathbf{x}_j - \mathbf{c}_i\right\|^2$ is a chosen distance measure between a data point and the cluster centre. Then, the algorithm assigns each observation to exactly one cluster, i.e., it assigns each observation to the cluster with the closest centroid, as follows:

$$\mathbf{C}_i = \left\{ \mathbf{x}_j : \left\|\mathbf{x}_j - \mathbf{c}_i\right\|^2 = \min_k \left\|\mathbf{x}_j - \mathbf{c}_k\right\|^2 \, \forall k \in \mathbb{N} : 1 \leqslant k \leqslant K \right\}. \tag{6.19}$$

and recomputing the centroids (means) for observations assigned to each cluster as:

$$\frac{1}{\mathbf{C}_i} \sum_{\mathbf{x}_j \in \mathbf{C}_i} \mathbf{x}_j. \tag{6.20}$$

Figure 6.14 illustrates the k-means clustering results used to partition the two-dimensional data space observations into $k = 2$ clusters, background and foreground, respectively.



(a) (b)

Figure 6.14: Example of k-means clustering results: (a) the two clusters along with their centroids and (b) resulting silhouette graph.

[5]Although it can be proved that the algorithm will always terminate, the k-means algorithm does not necessarily find the optimal configuration, corresponding to the global objective function minimum [59, 98].

On the right hand, is present a so-called *silhouette graph*, which is based on the comparison of its tightness and separation. Precisely, it shows which objects lie well within their cluster, and which ones are not distinctly in one cluster or another. This measure ranges from 1 (indicating points that are very distant from neighbouring clusters) through 0 (points that are merely somewhere in between clusters) to -1 (points that are probably assigned to the wrong cluster) [127]. Figure 6.14(b) clearly shows that most points in both clusters have a large silhouette value, indicating that those points are well-separated from neighbouring clusters. However, each cluster also contains a few points with low silhouette values, indicating that they are nearby to points from other clusters. The resulting Boolean matrix $\mathbf{B} \in \{0,1\}^{n\times n}$ then undergoes a morphological filtering operation consisting in the *closing* of the matrix $\mathbf{B}$ by a cross shaped structuring element $\boldsymbol{\varpi} \subset \mathbb{Z}^2$:

$$\boldsymbol{\varpi} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{6.21}$$

Thus, the morphological closing of the set $\mathbf{B}$ (Boolean matrix) by $\boldsymbol{\varpi}$ is given by the erosion of the dilation of that set (see Appendix A.4 for further details):

$$\mathbf{B} \bullet \boldsymbol{\varpi} = (\mathbf{B} \oplus \boldsymbol{\varpi}) \ominus \boldsymbol{\varpi}. \tag{6.22}$$

However, due to the duality property of the morphological primitives and since the structuring element is symmetric with respect to its origin (the centre of ϖ), the Equation (6.22) can be expressed as

$$\begin{aligned} \mathbf{B} \bullet \boldsymbol{\varpi} = \mathbf{B}^c \circ \hat{\boldsymbol{\varpi}} &= (\mathbf{B}^c \ominus \hat{\boldsymbol{\varpi}}) \oplus \hat{\boldsymbol{\varpi}} \\ &= \underbrace{(\mathbf{B} \ominus \hat{\boldsymbol{\varpi}})^c}_{\boldsymbol{\Omega}} \oplus \hat{\boldsymbol{\varpi}} = (\boldsymbol{\Omega} \ominus \boldsymbol{\varpi})^c \end{aligned} \tag{6.23}$$

where $\mathbf{B}^c$ is the *complement* of $\mathbf{B}$ and $\hat{\boldsymbol{\varpi}}$ represents the *reflection* of $\boldsymbol{\varpi}$. The proof of Equation (6.23) is given in Appendix A.4.

(a) (b) (c)

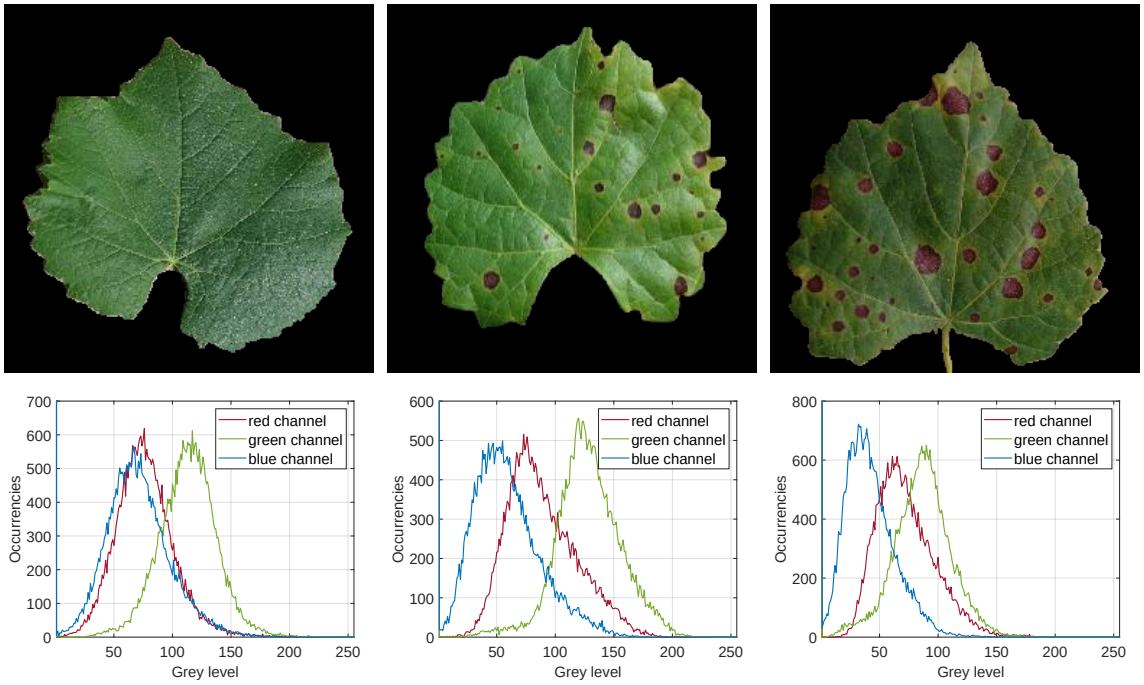
Figure 6.15: Example of leaf segmentation: (a) input image $\mathcal{X}$ representing a real leaf affected by *Black Rot* disease, (b) final Boolean mask $\mathbf{S}$, and (c) resulting image after segmentation by applying the mask $\mathbf{S}$ on each channel of the image $\mathcal{X}$.

6.3.4 Preliminary analysis

To carry out a histogram analysis, we shall describe the brightness variation in an image using its histogram.

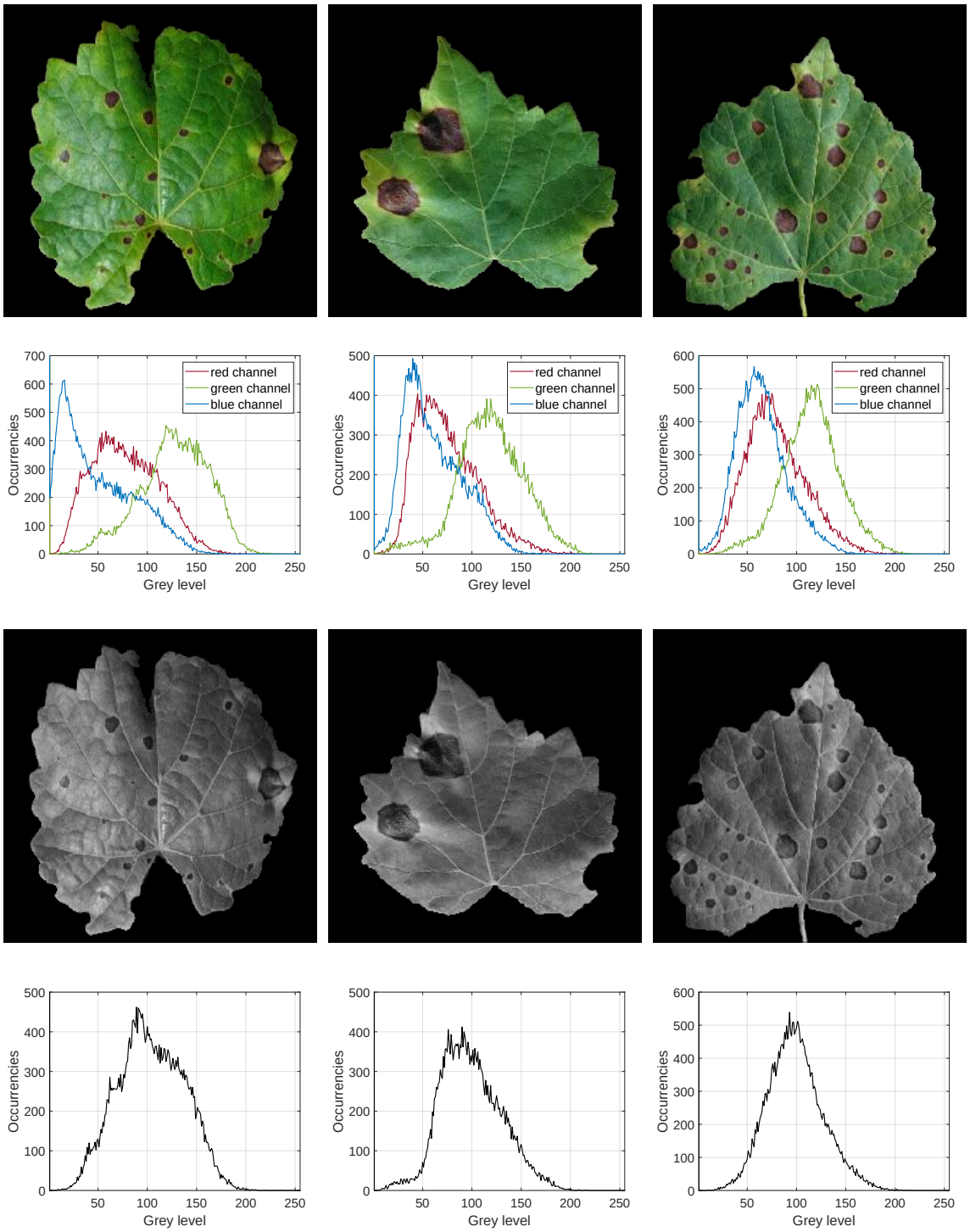
Definition 28. *The histogram of a digital image* $\mathbf{A} \in \mathbb{N}^{w \times h}$ *with intensity levels in* $[0, K-1]$*, is a one-dimensional discrete function such that* $h_{\mathbf{A}}(k) = n_k$*, where* k *is the* k*th intensity value and* n_k *is the number of pixels in the image with intensity* k*, that is, the histogram represents the frequency of occurrence of each gray level in* $\mathbf{A}$*.*

Thus, analysing the samples in Figure 6.16 by means of the histogram analysis method, it can be seen that each channel does not present a bimodal distribution thanks to the segmentation process, and obviously, the peak shown in $k = 0$ represents the background. Performing the analysis directly on the segmented image, it helps us to decide which values best suit the RGB triple to represent the diseased condition used to test the Criterion (6.4). However, as the shape of these histograms is not the same in different infectious diseases, several signatures corresponding to different pathological conditions should be defined a-priori. However, this method may produce misleading results due to non-uniform illumination and non-linear distorsion. Considering the green channel and the blue channel from the curves in Figure 6.16(a)–6.16(c) (bottom row), it can be found that they vary with the distribution of disease lesions. This is consistent with the process of disease appeared, which represents the evolution of pathological changes of the leaf from green to brown.



(a) (b) (c)

Figure 6.16: Examples of grape leaves with varying severity of Black Rot disease along with their histogram analysis: (a) healthy leaf, (b) diseased leaf with an estimated severity of 2.3%, and (c) diseased leaf with an estimated severity of 8.1%.



(a) DS = 2.62% (b) DS = 6.97% (c) DS = 9.08%

Figure 6.17: Histogram analysis of several grape leaves affected by infectious disease and relative disease severity estimation: (first row) segmented RGB image, (second row) histogram analysis of the three RGB channels, (third row) grey-scale version of the segmented image, and (last row) histogram analysis of the monochromatic image.

In Figure 6.17 has been reported the histogram analysis of several different grape leaves with disease symptoms upon pathogen infection.

The vector $\overline{x}_d$ that better approximates the desired one, which would be used to test the diseased condition of a point (i, j) in proximity of $\overline{x}_d$ as illustrated in Section 6.1, should be set considering the distributions of healthy and diseased regions of several digital leaf images and calculating common statistical parameters, e.g., mean, variance, and median.

However this approach may be time-consuming and unfeasible due to the lack of samples. From that, it follows that it is possible to find experimentally the most suitable value for each spot-based disease via grey histogram analysis. Indeed, the proposed approach relies on the identification of some diseased regions in order to compute the RGB vector we are looking for.

Firstly, to accomplish this goal, we consider the average grey level from the grey-scale leaf image $\mathbf{G} \in \mathbb{R}^{n\times n}$. Hence, let $\mathbf{A} \subset \mathbb{Z}^2$ be the set of points belonging to the leaf area, then

$$\overline{g} = \frac{1}{n(\mathbf{A})} \sum_{i,j\in\mathbf{A}} G_{i,j}. \tag{6.24}$$

Thus, the points that their gray level deviates from $\overline{g}$ by more than a threshold t, are assumed to be lesion spots.

Remark 8. *We stress that this kind of approach is not suitable to find all the lesions on the leaf surface. Indeed, this aspect is not required at all, since we are only interested in finding a suitable RGB vector that represents well enough the diseased condition. Hence, the precise values of the vector components are not critical if we consider an RGB triple far enough away from the colour representing the healthy condition in direction of that of the diseased condition.*

For instance, selecting the marked points $\tilde{\mathbf{p}}$ defined by their corresponding ones in the grey image $\mathbf{G}$ under the following condition:

$$\tilde{\mathbf{p}} = \mathbf{X}_{i,j,:} \iff G_{i,j} \leqslant \overline{g} - t \tag{6.25}$$

has worked satisfactorily for all the tested cases[6].

Note that the value of the green channel does not follow the frequency distribution of that reported in Figure 6.17 (second row), this is consistent with theoretical expectations as in symptomatic (diseased) regions the contribution in the green spectrum (520 nm $\leqslant \lambda_G \leqslant$ 565 nm) is far below with respect to that present in healthy regions.

6.3.5 Experimental results

Extensive experiments have been carried out to assess the performance and the effectiveness of the proposed algorithm, which are described in this section with particular regard to the dataset used in the experiments and the experimental setup, parameter optimisation, performance assessment, noise-rejection property, and computational efficiency of the algorithm.

[6]Note that this is the case of dark lesions over the leaf surface, in case of brigth lesions is required to negate the sign in front of the threshold.

Figure 6.18: Several examples of grape leaf image affected by different pathogenic diseases.

Dataset and experimental setup

The dataset used to evaluate the perfomrance of the proposed system consists of a subset of that described in Section 6.3.1. In particular, Table 6.2 summarises the subset of grape images used to test the system in disease detection configuration. Precisely, the subset consits of 4 063 samples, of which 3 640 are diseased leaf images exhibiting three different conditions and 423 healthy leaves. Figure 6.18 illustrates several examples of grape leaf images used in the experiments.

	Symptomatic			
	Black rot	Esca	Leaf blight	**Healthy**
	1 180	1 384	1 076	423
Total		3 640		423

Table 6.2: Dataset used in the experiments.

Hence, every single sample in the subset as defined in Table 6.2, has undergone the test procedure of the system which calculate the percentage of infection over the leaf surface as in the Equation (6.12). Finally, the decision with respect to the estimated severity of the disease is as follows.

Definition 29. *Given a set of disease severity estimations $\mathbf{x}$, the system has to determine if the input leaf belong to the healthy group or not. Formally, the classification problem consists of determining if a disease severity estimation x_i belongs to the class of the null hypothesis H_0 or not:*

$$(x_i, \psi) = \begin{cases} H_0 & \textit{if } x_i > \psi, \\ H_1 & \textit{otherwise.} \end{cases} \tag{6.26}$$

Parameter optimisation

Since the proposed approach for matching is based on a non-linear parameter-dependent system, it is very important to set its internal parameters in order to maximise the system performance. The parametrs to be fixed are: α, ρ, and ξ. Firstly, we recall that the parameter ρ is calculated as

$$\nu = \left\| \left[\frac{\mathbf{x}}{\|\mathbf{x}\|} - \frac{\overline{\mathbf{x}}_d}{\|\overline{\mathbf{x}}_d\|} \right] \right\|,$$

$$\rho = \frac{1}{1 + \alpha \nu^p}.$$

We proved in Section 6.2 that the parameter ν spans all the real values in the closed set $[0, \sqrt{3}]$. Thus, it is possible to test the behaviour of the function $\rho \in \mathbb{R}_+ : 0 \leqslant \rho \leqslant 1$ described in (6.7) based on different values of the parameter pair $(\alpha, p) \in \mathbb{R}_{++}$, however, it should consider that greater the colour discrepancy between the vectors $\mathbf{x}$ and $\overline{\mathbf{x}}_d$ (i.e., the observed point is far away from the *disease condition*), smaller the value of ρ, and vice versa. Indeed, since this is a strictly positive function and bounded from above by 1, the hiegher values should be achieved when the observed point is in proximity of diseased region.

Conversely, when a deep mismatch between the two vectors is observed, the function should achieve a low (strictly positive) value, whilst in between the two cases the function should behave as an inverse S-shaped softening function with the point of inflection to be in the middle of the domain. Hence, the desired behaviour is ensured for $\alpha > 1$.

Remark 9. *Let us consider the following function*

$$\rho = \frac{1}{1 + \alpha \nu^{p}}$$

where $\nu : \mathbb{R}_{++} \to [0, \sqrt{3}]$*, thus the problem is to find a parameter pair* $(\alpha, p) \in \mathbb{R}_{++}$ *such that when* $\nu = \frac{\sqrt{3}}{2}$ *we get* $\rho = \frac{1}{2}$*. Hence, by substitution we obtain*

$$\frac{1}{2} = \frac{1}{1 + \alpha\sqrt{3}^{p}} \;\Rightarrow\; \log_{\frac{\sqrt{3}}{2}} \alpha^{-1} = p$$

and fixing the numerical value of the constant $\alpha > 1$ *therefore defines the other one in the proper way. In fact, considering the last equation, if* $0 < \alpha < 1$ *we get negative values for* p *which yields curves that rise rather than fall.*

The desired behaviour of the monotonic function is illustrated in Figure 6.19.

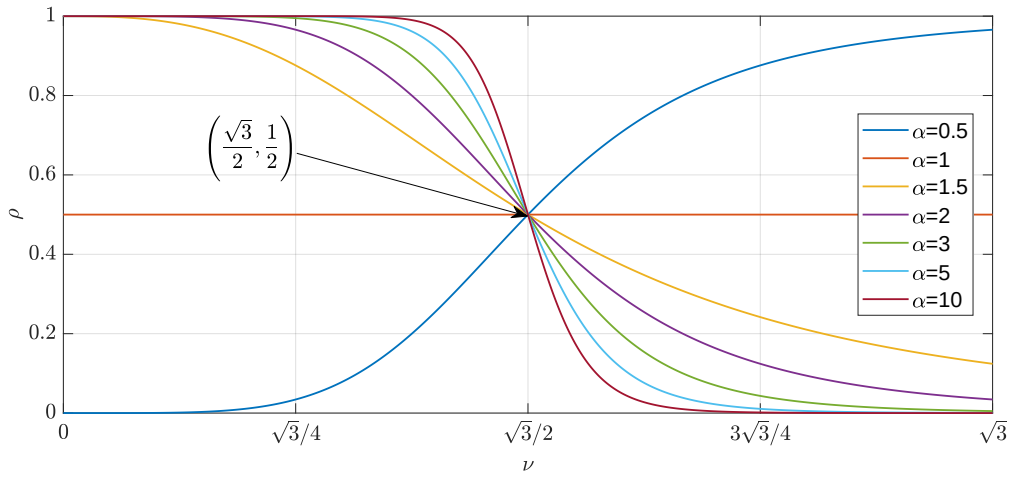


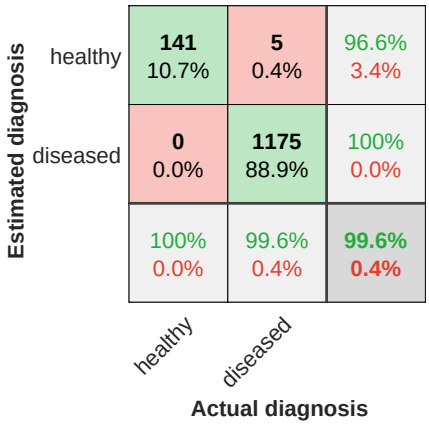
Figure 6.19: Example of several curves of the monotonic function ρ using several different parameters.

As described in section 6.1, the choice of the parameter ξ has to do with the Criterion (6.4), indeed the criterion would be satisfied if the discrepancy between the normalised vectors $\mathbf{x}$ (observed point) and $\overline{\mathbf{x}}_d$ (diseased condition) is less than or equal to the tolerance ξ. Hence, for high values of ξ the system will be prone to considerable risk of type-I error. Conversely, for low values of ξ the system will be prone to considerable risk of type-II error. Experimentally, we have fonud that an efficient range of tolerances to choose from is $\frac{1}{10} \leqslant \xi \leqslant \frac{1}{4}$. Precisely, in our experiments, the best choice of tolerance it turns out to be $\xi = \frac{1}{7}$.

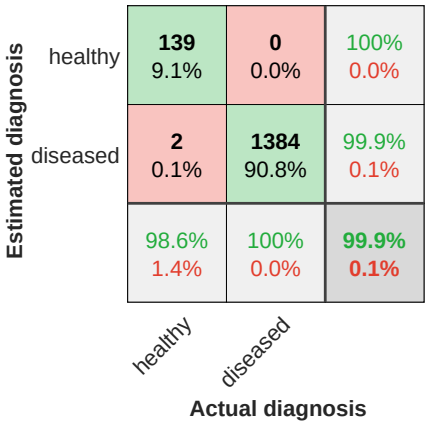
Performance assessment

In order to investigate and to assess the performance of the proposed detection method that is agnostic to the type of disease, the dataset has been split into three different disease-healthy binary subsets, each one considering only one specific disease. Figure 6.20 reports the confusion matrices for the proposed disease detection method.

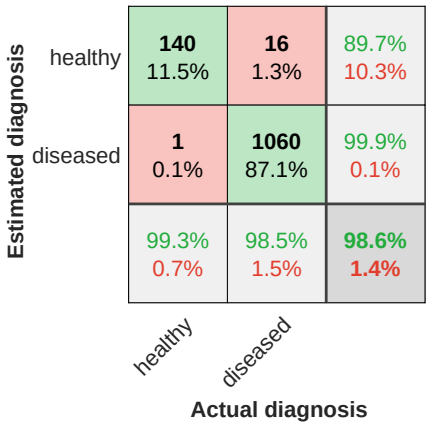
Remark 10. *Testing the proposed system with several variations of the original dataset do not affect the results, since the proposed algorithm is invariant to rotation and translation.*



(a) Pathogen: *Guignardia bidwelli*



(b) Pathogen: *Phaeomoniella spp.*



(c) Pathogen: *Pseudocercospora vitis*

Figure 6.20: Confusion matrices for the proposed disease detection method.

Accuracy and F-score computed on confusion matrices have been (and still are) among the most popular adopted metrics in binary classification tasks. However, these statistical measures can dangerously show overoptimistic inflated results, especially on imbalanced datasets [26]. Hence, among of all the parameters described in Section 5.3, Matthews correlation coefficient (MCC) is the only one that takes into account true and false positives and negatives and is generally regarded as a balanced measure which can be used even if the classes are of very different sizes [17]. However, for the sake of completeness, we have summarised in Table 6.3 all the main measures described in Section 5.3.

Actually, the results indicate that the algorithm performs incredibly well in localising symptomatic regions for disease detection achieving an average accuracy of 99.36% and thus demonstrating the effectiveness of the proposed system.

	Performance metrics				
Disease	Accuracy	Precision	Recall	F_1-measure	MCC
Black Rot	0.9962	1.0000	0.9658	0.9829	0.9806
Esca	0.9987	0.9858	1.0000	0.9929	0.9922
Leaf Blight	0.9860	0.9929	0.8974	0.9428	0.9364

Table 6.3: Summary of disease detection performance of the proposed system.

Despite the inability to ground-truth boundaries due to subjectivity, the proposed algorithm has been consistently robust quantifying disease lesion from symptomatic leaf images. Figure 6.21 presents a statistical comparison between the three diseases in terms of severity, whilst the Table 6.4 lists the statistics of disease severity for each grape disease dataset.

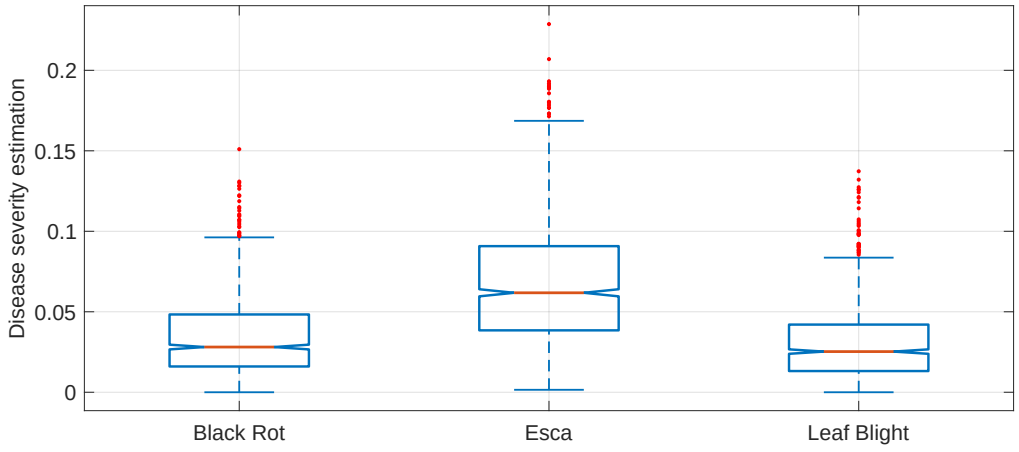


Figure 6.21: Disease severity statistical analysis through the boxplot of data results from each grape disease dataset.

	Disease severity statistics				
Disease	Mean	Standard deviation	Median	Maximum	Minimum
Black Rot	0.0345	0.0239	0.0281	0.1510	0.0000
Esca	0.0680	0.0371	0.0618	0.2287	0.0015
Leaf Blight	0.0305	0.0231	0.0253	0.1373	0.0000

Table 6.4: Summary of the disease severity statistics for each grape disease dataset.

Noise immunity

To conduct experiments on noisy leaf images and demonstrate the robustness of the dynamic algorithm with respect to noise, the system has been tested in noisy conditions. As argued in Section 4.3, impulse noise is a kind of noise which can have many different origins, often due to transmission errors, faulty memory locations, or timing errors in analog-to-digital conversion. The impulse noise model has been defined through the probability density function (4.24) in the same section and, in the case of the RGB colour space, it is always independent, randomly distributed, and uncorrelated with respect to each colour component. We can distinguish two cases, (i) the first one arises when the image affected by the noise represents a diseased leaf and (ii) the second one considers the case of an image with no diseased regions (healthy leaf). However, the second case is more challenging with respect to the first one, since the presence of noise in a healthy leaf image may lead to a type-I error (false positive) in the detection of the disease, whilst this is not a problem at all in the case of a symptomatic leaf image. Thus, the experiments to test the noise robustness of the system have been conducted considering only the dataset containing healthy leaves.

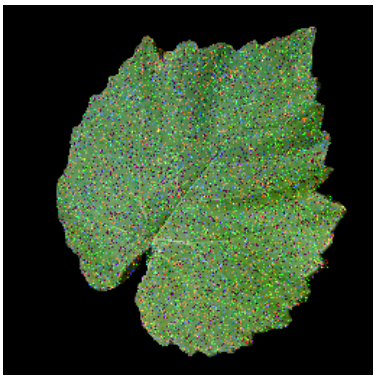


Figure 6.22: Example of healthy leaf image affected by impulse noise with $p = 15\%$.

	Performance metrics			
Noise probability	Accuracy	TP	FP	Error rate
$p = 5\%$	0.9740	412	11	0.0260
$p = 10\%$	0.9409	398	25	0.0591
$p = 15\%$	0.9149	387	36	0.0851

Table 6.5: Summary of the disease severity statistics for each grape disease dataset.

Table 6.5 reports the result of noise-rejection experiments and shows that the performance of the system is not that much degraded: even in presence of noise, the system is able to correctly detect a healthy leaf with an accuracy equal to 97.4%, 94.1%, and 91.5% for $p = 5\%$, $p = 10\%$, and $p = 15\%$, respectively.

6.4 Remarks

In Section 6.1 has been proposed a novel unified method based on the dynamical systems theory that allows the detection and severity estimation of pathogenic diseases regardless of disease type. The system does not require any training and the proposed experimental setup has allowed to assess the system ability to generalise symptoms beyond any previously seen conditions, also achieving excellent results, even in presence of significant noise. Even though this study is a first step towards a fully automatic diagnosis of plant disease severity, the preliminary results are very promising also allowing the potential to provide new applications for infectious disease screening that can be remotely assessed in a precision agriculture context.

7

Conclusions and outline

This final chapter summarises the principal findings of this thesis.

The first part opens with the Chapter 3 which presents an introduction to the fundamental concepts, theory, design, and application of biometric characterisation of human beings, including the main biometric traits along with their properties and the various biometric system operating modalities, as well as the primitives of performance metrics to determine the system accuracy and security which are related to the applicability in real-world deployments. The operational objectives of biometric security applications are just as variable as the technologies: some systems seek for known (or unknown) individuals; some verify a claimed (or unclaimed) identity; and some verify that the individual is not present at all in the system. To conduct the experiments presented in the Chapter 4, the chosen operational modality is the authentication mode, although there is no difference in using one of the other.

The Chapter 4 presents a noise-rejecting unified method for line-like feature matching, which relies on a recursive algorithm based on a monotone dynamical system whose output converges either to zero, when a deep mismatch exists between the samples to be compared, or to a high value, when a good matching is observed. The idea that characterises the algorithm is to recursively increase the value of a pixel i, j if in the complementary neighbourhood (namely, the neighbourhood of the corresponding pixel i, j in the comparison image) there are pixels with large values. Conversely, if the pixels in the complementary neighbourhood have low values, the value of pixel i, j converges to zero. In the enhanced version, instead, the dynamical system also includes a term that *initially* increases the value of a pixel if the pixels in a proper neighbourhood *in the same image* have large values, and then vanishes with time; it has the effect of initially thickening the relevant patterns. A considerable advantage of the proposed unified method for line-like feature matching is its robustness with respect to such a noise. Indeed, images corrupted by noise (e.g., impulse noise) are easily recognised, whilst an image randomly generated is rejected even if compared with itself. To consolidate the theorethical results presented above, Sections 4.4 and 4.5 illustrate two examples of biometric recognition systems where the first one is based on principal palm lines features acquired in the visible spectrum, whilst the second one is based on subcutaneous palm vascular patterns acquired in the near-infrared spectrum. The images provided as an input to the dynamic algorithm have undergone an image processing based on two phases: the first involving preprocessing operations to make the system invariant to rotation and translation of the palm with respect

to the image and the second consisting of a sequence of robust feature extraction steps that allow to detect the principal lines of the palm without introducing noise. Extensive experiments have been conducted to evaluate the performance of the system and the results obtained from the tests clearly demonstrate the effectivenes of the proposed technique, which is able to produce results in line with the state-of-the-art results in the current literature. It is worth of note to highlight that the system has undergone a parameter tuning step which is required only once and there is no need to perform this phase again, even using different databases with different wavelength illumination images, which proves the effectiveness and robustness of the proposed system. As a further work, it is possible to employ the system using different biometric traits (e.g., retina, dorsal veins, and finger veins) or even in other contexts.

The second part starts with the Chapter 5 which introduces the fundamental concepts in the field of phytopathology oriented to image-based diagnosis of plant disease symptoms, including the main grapevine pathogenic diseases along with their properties and the analysis of visual symptoms used for the assessment of disease severity, as well as the primitives of performance metrics to determine the system accuracy and reliability which are related to the applicability in real-world deployments. In particular, in the vineyard, symptoms are visually most noticeable on leaves, which may include a detectable change in color, shape, or function of the plant as it responds to the pathogen. Among the potential pathogens, phytoplasmas, viruses, bacteria, fungi, and nematodes can attack grapevine with different infection mechanisms and evasion strategies. An in-depth description of common signs and symptoms that appear on the leaves of the grapevine plant, caused by a few relevant pathogenic diseases, is also provided in this Chapter.

The Chapter 6 aimed at presenting a noise-rejecting unified method based on the dynamical systems theory that allows the detection and severity estimation of pathogenic diseases regardless of disease type. The system relies on a recursive algorithm based on a positive non-linear dynamical system whose evolution depends on the input tensor representing the leaf image to be analysed. In particular, the idea that characterises the algorithm is to recursively spread the disease to fill the infected regions of the leaf only if there are symptoms of the condition itself, otherwise the leaf will not be affected by any changes. Furthermore, the system does not require any training and the proposed experimental setup has allowed to assess the system ability to generalise symptoms beyond any previously seen conditions, also achieving excellent results, even in presence of significant noise. Even though this study is a first step towards a fully automatic diagnosis of plant disease severity, the model has proven to be highly accurate and robust and the experimental results are very promising also allowing the potential to provide new applications for infectious disease screening that can be remotely assessed in a precision agriculture context. Possible improvements from the research conducted in this thesis may be used in the future to further improve the model performances, also integrating the possibility of classifying the diseases detected on the basis of the symptoms analysis.

As argued in the relative Chapters, both the proposed unified methods, present a main advantage that relies in the robustness when dealing with low-resolution and noisy images. Indeed, an essential issue related to digital image processing is to effectively reduce noise from an image whilst keeping its features intact. The impact of noise (e.g., signal independent and uncorrelated noise) is effectively reduced and does not affect the final result allowing the proposed systems to ensure a high accuracy and reliability.

A

Mathematics in digital image processing

A.1 Optimum global thresholding using Otsu's method

Suppose that $i \in [0, L-1]$ is the range of grey-scale of an image $I(x, y)$ with N pixels and the number of pixels with grey level i is denoted by $h(i)$, then the probability of grey level i is given by

$$p(i) = \frac{h(i)}{N}, \qquad p(i) \geqslant 0, \qquad \sum_{i=0}^{L-1} p(i) = 1. \tag{A.1}$$

Now suppose that there is a threshold value $t \in [0, L-1]$ such that the pixels of the image are dichotomised into two classes, C_1 with grey levels $[0, \ldots, t]$ and C_2 with grey levels $[t+1, \ldots, L-1]$, then the probabilities of occurrence ω and mean μ of each class are:

$$\omega_1 = \sum_{i=0}^{t} p(i) = \omega(t) \tag{A.2}$$

$$\omega_2 = \sum_{i=t+1}^{L-1} p(i) = 1 - \omega(t) \tag{A.3}$$

and

$$\mu_1 = \sum_{i=0}^{t} i\frac{p(i)}{\omega_1} = \frac{\mu(t)}{\omega(t)} \tag{A.4}$$

$$\mu_2 = \sum_{i=t+1}^{L-1} i\frac{p(i)}{\omega_2} = \frac{\mu_T - \mu(t)}{1 - \omega(t)} \tag{A.5}$$

where

$$\omega(t) = \sum_{i=0}^{t} p(i), \qquad \mu(t) = \sum_{i=0}^{t} ip(i) \tag{A.6}$$

are respectively the zero order and the first order cumulative moments of the histogram up to the threshold, and

$$\mu_T = \mu(L) = \sum_{i=0}^{L-1} ip(i) \tag{A.7}$$

is the mean for the whole image.

Obviously, by summing the parts, it is easy to show that:

$$\omega_1\mu_1 + \omega_2\mu_2 = \mu_T, \qquad \omega_1 + \omega_2 = 1 \tag{A.8}$$

and it is clear that the total sum of the probabilities is always equal to one.
The class variances are given by

$$\sigma_1^2 = \sum_{i=0}^{t} (i-\mu_1)^2 \frac{p(i)}{\omega_1} \tag{A.9}$$

$$\sigma_2^2 = \sum_{i=t+1}^{L-1} (i-\mu_2)^2 \frac{p(i)}{\omega_2}. \tag{A.10}$$

As stated in [116], the within-class variance σ_W^2, the between-class variance σ_B^2 and the total variance σ_T^2 of grey levels are defined as follows

$$\sigma_W^2 = \omega_1\sigma_1^2 + \omega_2\sigma_2^2 \tag{A.11}$$

$$\begin{aligned}\sigma_B^2 &= \omega_1(\mu_1-\mu_T)^2 + \omega_2(\mu_2-\mu_T)^2 \\ &= \omega_1\omega_2(\mu_1-\mu_2)^2\end{aligned} \tag{A.12}$$

$$\sigma_T^2 = \sum_{i=0}^{L-1} (i-\mu_T)^2 p(i). \tag{A.13}$$

Hence, the relationship between σ_T^2, σ_B^2, and σ_W^2, can be expressed as

$$\begin{aligned}\sigma_T^2 &= \sum_{i=0}^{t} (i-\mu_1)^2 p(i) + \sum_{i=t+1}^{L-1} (i-\mu_2)^2 p(i) + \\ &\quad + (\mu_1-\mu_T)^2\omega_1 + (\mu_2-\mu_T)^2\omega_2 \\ &= \left[\omega_0\sigma_0^2 + \omega_1\sigma_1^2\right] + \left[(\mu_0-\mu_T)^2\omega_0 + (\mu_1-\mu_T)^2\omega_1\right] \\ &= \sigma_W^2 + \sigma_B^2\end{aligned} \tag{A.14}$$

The optimal threshold can be determined by maximising one of the following equivalent discriminant criterion with respect to t

$$\lambda = \frac{\sigma_B^2}{\sigma_W^2}, \quad \kappa = \frac{\sigma_T^2}{\sigma_W^2}, \quad \eta = \frac{\sigma_B^2}{\sigma_T^2} \tag{A.15}$$

where, by using the Equation [A.14], κ and η can be written as

$$\kappa = \frac{\sigma_W^2 + \sigma_B^2}{\sigma_W^2} = 1 + \frac{\sigma_B^2}{\sigma_W^2} = 1 + \lambda \tag{A.16}$$

$$\eta = \frac{\sigma_B^2}{\sigma_W^2 + \sigma_B^2} = \frac{\sigma_B^2}{\sigma_W^2}\frac{1}{1+\frac{\sigma_B^2}{\sigma_W^2}} = \frac{\lambda}{1+\lambda}. \tag{A.17}$$

As criterion we use η, and the optimal threshold t^* that maximises η (or equivalently maximises σ_B^2) is such that

$$\sigma_B^2(t^*) = \max_{0\leqslant t<L} \sigma_B^2(t). \tag{A.18}$$

A.2 Canny's edge detector operator

This edge detector consists of three steps, whose details are given below.

Step 1

There are three criteria relevant to edge detector performance:

- *good detection* – it is important that edges that occur in the image should not be missed and that there be no spurious responses;
- *good localisation* – the distance between the edge pixels marked by the edge detector and the "center" of the actual edge should be as small as possible;
- *one response to one edge* – the edge detector should not identify multiple edge pixels where only a single edge exists.

The problem is to identify a filter that optimises the three edge detection criteria. To facilitate the analysis, it has been considered a one-dimensional edge profile assuming an ideal step edge $Au_{-1}(x)$ affected by additive white Gaussian noise $n(x)$ (see Figures A.1(a) and A.1(b)), therefore the input signal $G(x)$ can be represented by the step:

$$G(x) = Au_{-1}(x) + n(x) \tag{A.19}$$

where A is the amplitude of the step, the variance of the input white noise is n_0^2 such that $\{n_0^2 = \langle n^2(x)\rangle \, \forall x\}$ and $u_{-1}(x)$ is the first derivative of the Dirac delta function

$$u_{-1}(x) = \begin{cases} 1 & \text{if } x \geqslant 0, \\ 0 & \text{if } x < 0. \end{cases} \tag{A.20}$$

Then the signal $G(x)$ is convolved with a filter whose impulse response and the output are illustrated in Figure A.1(c) and A.1(d). In one dimension, the response of the antisymmetric spatial filter $f(x)$ to the edge $G(x)$ is given by a convolution integral:

$$H = \int_{-W}^{+W} G(-x)f(x)dx \tag{A.21}$$

assuming the filter has a finite impulse response (FIR) bounded by $[-W, +W]$ and is zero outside this range. The true root mean square (RMS) value of the response to the noise $n(x)$ is

$$H_n = n_0 \sqrt{\int_{-W}^{+W} f^2(x)dx} \tag{A.22}$$

where n_0^2 represents the noise power spectral density (PSD).
Mathematically, the three edge detection criteria to be optimised are expressed as follows.

1. The first criterion corresponds to maximising signal-to-noise ratio (SNR), which is defined as:

$$SNR = \frac{H}{H_n} = \frac{A}{n_0} \cdot \frac{\left| \int_{-W}^{0} f(x)dx \right|}{\sqrt{\int_{-W}^{+W} f^2(x)dx}} = \frac{A}{n_0} \cdot \Sigma(f) \tag{A.23}$$

   then, the goal is to find the impulse response $f(x)$ wich maximises Σ.

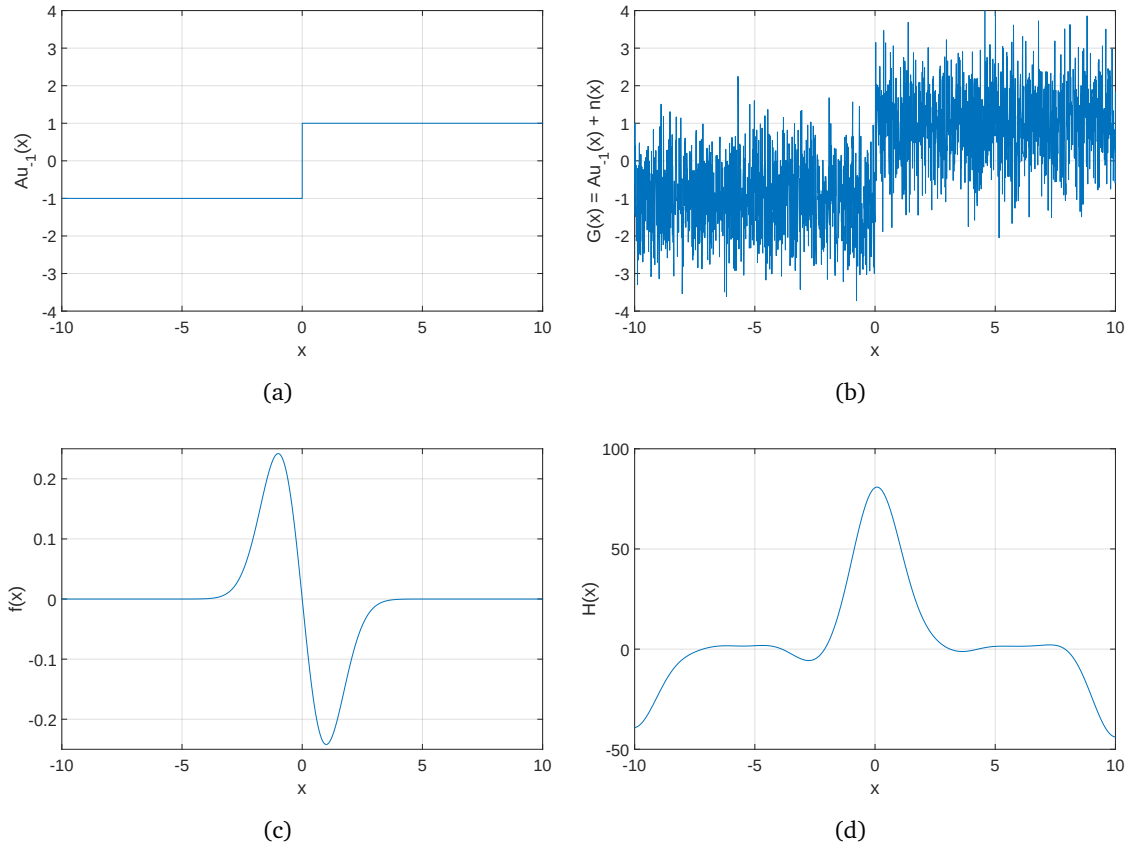


Figure A.1: Edge detector application: (a) ideal step and (b) ideal step affected by Gaussian noise, (c) first derivative of Gaussian operator, and (d) result of the convolution.

2. The second criterion corresponds to minimising the variance σ^2 of the zero-crossing position or maximising the localisation criterion LCZ that represents the reciprocal of σ which is the distance between the located edge and the true edge:

$$\mathrm{LCZ} = \frac{\left|\int_{-W}^{+W} G'(-x)f'(x)dx\right|}{n_0\sqrt{\int_{-W}^{+W} f'^2(x)dx}} = \frac{A}{n_0}\cdot\frac{|f'(0)|}{\sqrt{\int_{-W}^{+W} f'^2(x)dx}} = \frac{A}{n_0}\cdot\Lambda(f') \tag{A.24}$$

then, the goal is to find the impulse response $f(x)$ wich maximises Λ.

3. The last criterion correspond to limit the number of peaks in the response to a single edge, so there will be a low probability of declaring more than one edge, where x_{zc} represents a constraint that is the mean distance between zero-crossings of $f(x)$, in other words implies that there should be no multiple responses to the same edge in a small region.

$$x_{zc}(f) = \pi\left(\frac{\int_{-\infty}^{+\infty} f'^2(x)dx}{\int_{-\infty}^{+\infty} f''^2(x)dx}\right)^{\frac{1}{2}}. \tag{A.25}$$

The solution of these problems is to maximise the product of ((A.23)) and ((A.24)) (invariant under changes of scale or amplitude) under the multiple response constraint ((A.25)). An efficient approximation turns out to be the first derivative of a Gaussian function, so the first step of the Canny's operator is to process the input image with Gaussian convolution, and obtain the gradient image through differential operation to the smoothly image.
In order to do this, suppose to convolve the input image $I(x, y)$ with an operator $G_n(x, y)$ which is the first derivative of a two-dimensional Gaussian $G(x, y)$ in some direction $\boldsymbol{n}$:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{A.26}$$

and

$$G_n(x, y) = \frac{\partial G(x, y)}{\partial \boldsymbol{n}} = \boldsymbol{n} \cdot \nabla G \tag{A.27}$$

where $\boldsymbol{n}$ is oriented normal to the direction of an edge to be detected and can be estimated by the following:

$$\boldsymbol{n} = \frac{\nabla (G * I)}{|\nabla (G * I)|}. \tag{A.28}$$

Making use of Gaussian function's separability, the two convolutions can be decomposed to two one-dimensional filters as follows:

$$G_x = \frac{\partial G(x, y)}{\partial x} = -\frac{x}{\sigma^2} G(x, y) \tag{A.29}$$

$$G_y = \frac{\partial G(x, y)}{\partial y} = -\frac{y}{\sigma^2} G(x, y). \tag{A.30}$$

In order to reduce the computational cost, we use the separability property of the Gaussian variables $G(x, y) = G(x)G(y)$ wich allows to filter the input image with a sequence of convolutions with one-dimensional mask as follows:

$$G_x = -\frac{x}{2\pi\sigma^4} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2}{2\sigma^2}\right) \tag{A.31}$$

$$G_y = -\frac{y}{2\pi\sigma^4} \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2}{2\sigma^2}\right). \tag{A.32}$$

Then, after convoluting the input image $I(x, y)$ using the Equations ((A.31)) and ((A.32)), respectively, we obtain the partial derivatives of $I(x, y)$ in both directions x and y:

$$M_x(x, y) = \frac{\partial}{\partial x} (G(x, y) * I(x, y)) \tag{A.33}$$

$$M_y(x, y) = \frac{\partial}{\partial y} (G(x, y) * I(x, y)) \tag{A.34}$$

at such an edge point, the edge strength will be $M(x, y)$ that is:

$$M(x, y) = |\nabla (G * I)| = \sqrt{M_x^2(x, y) + M_y^2(x, y)}. \tag{A.35}$$

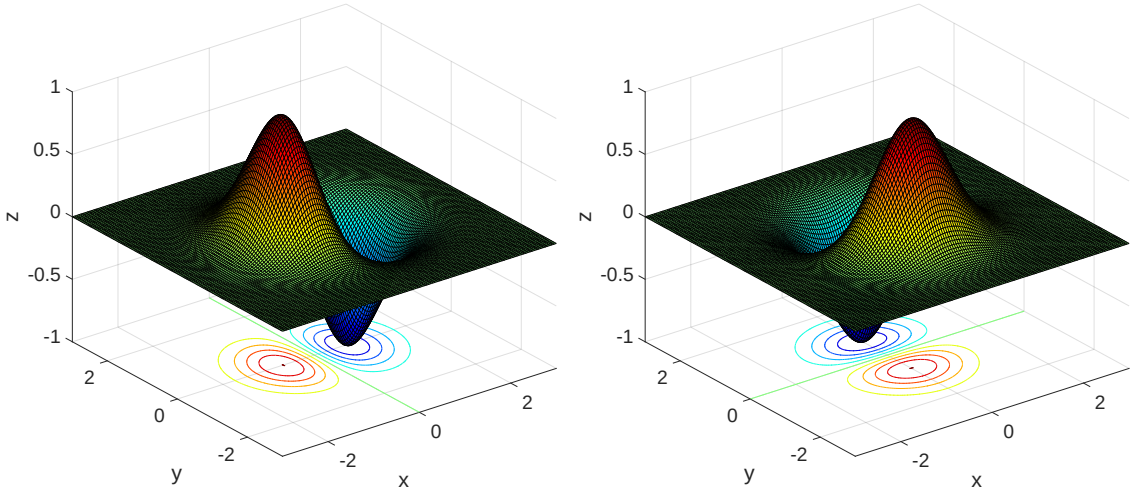


Figure A.2: The first order partial derivatives of the Gaussian function: (a) derivative with respect to x and (b) derivative with respect to y.

Step 2 – Due to the multiple response, edge magnitude $M(x, y)$ may contain wide ridges around the local maxima; what must be done is to threshold the image based partly on the direction of the gradient at each pixel using the "non-maximum suppression" algorithm, that removes the non-maximum pixels preserving the connectivity of the contours.

The basic idea is that edge pixels have a direction associated with them; the magnitude of the gradient at an edge pixel should be greater than the magnitude of the gradient of the pixels on each side of the edge.

The gradient direction is obtained by the follow computation:

$$\Theta(x, y) = \arctan\left(\frac{M_y(x, y)}{M_x(x, y)}\right). \tag{A.36}$$

Algorithm Non-maximum suppression

Input: Images $M(x, y)$ and $\Theta(x, y)$.
Output: Image $NMS(x, y)$.

1. Copy the gradient magnitude image $M(x, y)$ into the output image $NMS(x, y)$.
2. From each postion (x, y), step in the two orthogonal directions to edge orientation $\Theta(x, y)$ (see Figure A.3).
3. Denote the initial pixel (x, y) by Ω, the two neighbouring pixels in the orthogonal directions by Ψ and Φ.
4. **if** $(M(\Psi) > M(\Omega)) \vee (M(\Phi) > M(\Omega))$ **then**
   discard the pixel (x, y) by setting $NMS(x, y) = 0$.
   end if

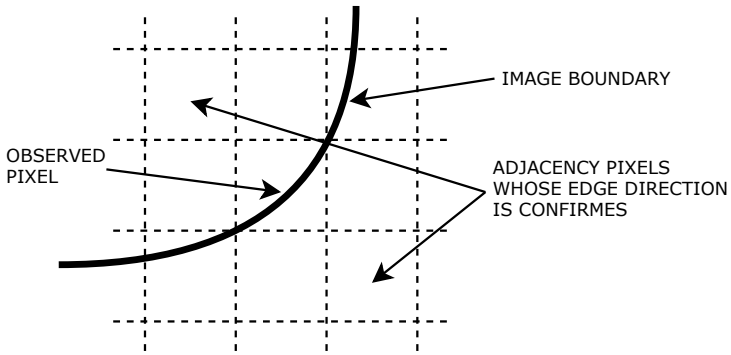


Figure A.3: Non-maximum suppression: adjacency pixels relationship.

Step 3 – The last image still contains noisy local maxima and needs to be thresholded to determine which pixels are edge pixels and which are not.

The study reported in [23] suggests thresholding using hysteresis rather than simply selecting a threshold value to apply everywhere.

The algorithm uses a high threshold t_h and a low threshold t_l such that: if any pixel (x, y) in the image has a value greater than t_h is presumed to be an edge pixel, and is marked as such immediately. Then, any pixel that is connected to this edge pixel and that has a value greater than $t_l o w$ is also selected as edge pixel, and is marked too.

The marking of neighbours can be done recursively, as it is in the function hysteresis, or by performing multiple passes through the image.

The relationship between the two threshold values is $t_l = t_h/2$ where t_h is calculated with the Otsu's algorithm [42].

Algorithm Hysteresis thresolding

Input: Image $NMS(x, y)$.
Output: Image $Y(x, y)$.

1. Copy the non-maximum suppression image $NMS(x, y)$ into the output image $Y(x, y)$.
2. **for each** postion (x, y) **do**
   if the pixel (x, y) has a value less than t_l **then**
   discard the pixel value
   else if the pixel (x, y) has a value greater than t_h **then**
   save the pixel value in the output image $Y(x, y)$
   else if the pixel (x, y) is a candidate **then**
   follow the chain of connected local maxima in both directions along the edge, as long as $NMS(x, y) > t_l$
   if the starting candidate pixel (x, y) is connected to a strong pixel **then**
   save the candidate pixel in the output image
   else
   do not save the candidate pixel.
   end if
   end if
   end for

A.3 Laplacian of Gaussian

The Laplacian of Gaussian filter is based on the second-order derivative of the Gaussian function and is invariant to rotation, meaning that it is insensitive to the direction in which the discontinuities (e.g., edge and corner-like image structures) run.
Considering a one-dimensional case, the zero-mean Gaussian function is defines as:

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{A.37}$$

and its first and second derivatives, whose behaviours are depicted in Figure A.4, are given by

$$\frac{dG(x)}{dx} = -\frac{x}{\sqrt{2\pi}\sigma^3} \exp\left(-\frac{x^2}{2\sigma^2}\right) = -\frac{x}{\sigma^2} G(x) \tag{A.38}$$

$$\frac{d^2G(x)}{dx^2} = \frac{x^2-\sigma^2}{\sqrt{2\pi}\sigma^5} \exp\left(-\frac{x^2}{2\sigma^2}\right) = \frac{x^2-\sigma^2}{\sigma^4} G(x)\,. \tag{A.39}$$

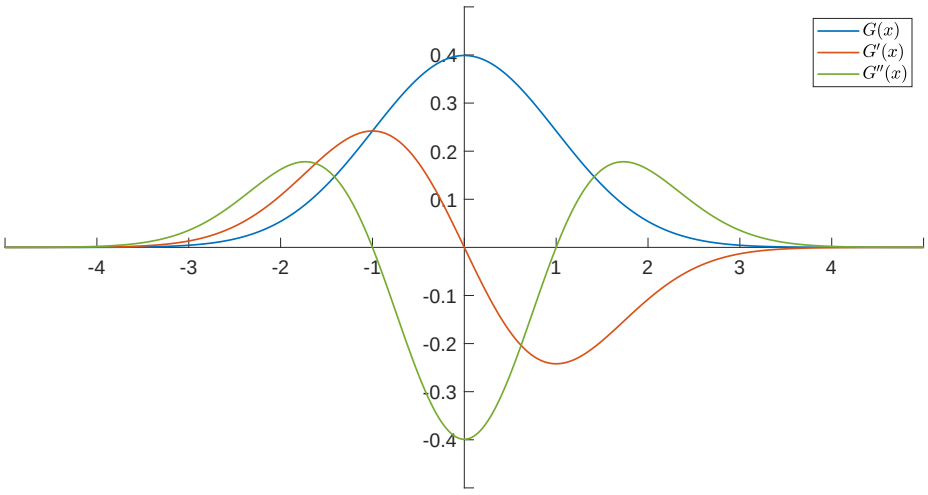


Figure A.4: Representation of a one-dimensional Gaussian function with zero mean and unit variance in blue, its first derivative in orange, and its second derivative in green.

Definition 30. *The Laplacian operator of a continuous, two-dimensional function* $f(x,y)$ *is defined as the divergence of the gradient of the function or, equivalently, the sum of the second partial derivatives of the function with respect to each independent variable (unmixed second partial derivatives):*

$$\begin{aligned}\nabla^2 f(x,y) &= \nabla \cdot \nabla f(x,y) \\ &= \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}\end{aligned} \tag{A.40}$$

where the result, unlike the gradient, is not a vector but a scalar quantity. This operator has the property of being isotropic, hence its value is invariant with respect to rotations of the coordinate system.

By applying the Laplacian operator to a rotationally symmetric two-dimensional Gaussian function

$$G(x,y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{A.41}$$

with zero mean and unit variance in both directions. Analogously to the previous one-dimensional case, it is possible to find an expression for the Laplacian of the Gaussian performing the following differentiations:

$$\begin{aligned}
\nabla^2 G(x,y) &= \nabla\cdot\nabla f(x,y) \\
&= \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2} \\
&= \frac{\partial}{\partial x}\left[\frac{-x}{2\pi\sigma^4}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right] + \frac{\partial}{\partial y}\left[\frac{-y}{2\pi\sigma^4}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)\right] \\
&= \left(\frac{x^2-\sigma^2}{2\pi\sigma^6}\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) + \left(\frac{y^2-\sigma^2}{2\pi\sigma^6}\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\
&= \left(\frac{x^2+y^2-2\sigma^2}{2\pi\sigma^6}\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\
&= \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] G(x,y).
\end{aligned} \tag{A.42}$$

The final expression is named Laplacian of Gaussian (LoG) and is illustrated in Figure A.5. Furthermore, the continuous Laplacian of Gaussian function in Equation (A.42) has the absolute value integral

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left|\nabla^2 G(x,y)\right|dxdy = \frac{4}{\sigma^2 e} \tag{A.43}$$

and zero mean, that is,

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\nabla^2 G(x,y)dxdy = 0. \tag{A.44}$$

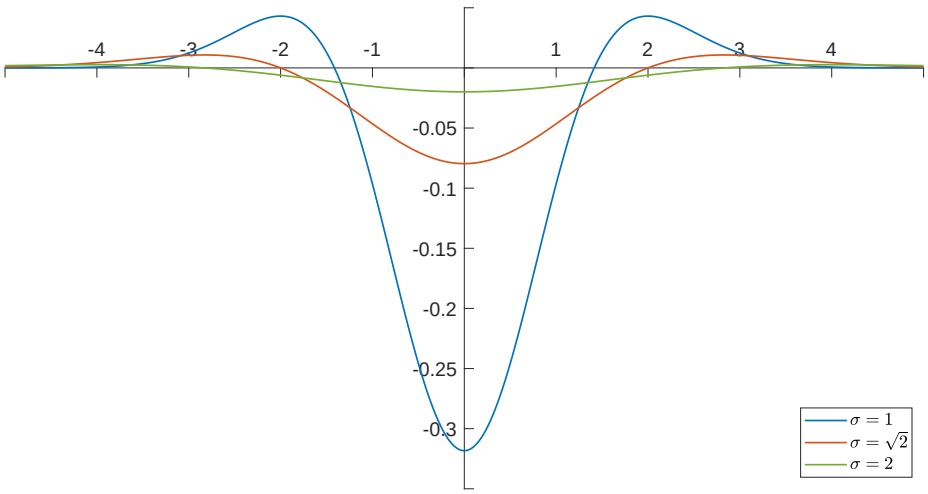
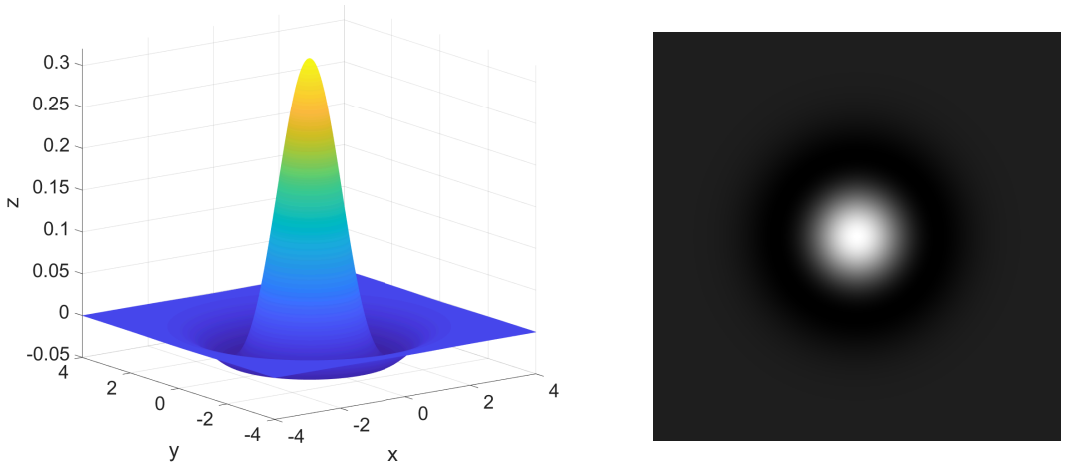


Figure A.5: Cross section of LoG function as defined in Equation (A.42) for different σ.

Note that in the cross section of the two-dimensional LoG function illustrated in Figure A.5, the zero crossings occur at $x^2 + y^2 = 2\sigma^2$, thus defining a circle of radius $\sqrt{2}\sigma$ centred on the origin. Figure A.6 illustrates a three-dimensional plot and the relative image of the negative of the LoG function[1].



(a) (b)

Figure A.6: LoG function as defined in Equation (A.42) with unit variance: 3D plot of the negative of the LoG, an (b) negative of the LoG displayed as an image.

LoG-approximation by the Difference of Gaussians (DoG)

The Marr-Hildreth algorithm [102] consists of convolving the Laplacian of Gaussian filter with an input image $I(x, y)$

$$H(x, y) = \left[\nabla^2 G(x, y)\right] * I(x, y) \tag{A.45}$$

and then finding the zero crossings of $H(x, y$ to determine the locations of edges in $I(x, y)$. Because convolution and differentiation are the only linear operators involved, it is possible to interchange them[2]:

$$H(x, y) = \nabla^2 \underbrace{\big(G(x, y) * I(x, y)\big)}_{\text{Gaussian smoothing}} = I(x, y) * \nabla^2\big(G(x, y)\big). \tag{A.46}$$

hence, it is possible to smooth the image first using a Gaussian filter and then apply the Laplacian operator to the result. The Marr-Hildreth edge-detection algorithm may be summarized as follows [55]:

1. smooth the input image $I(x, y)$ with a Gaussian lowpass filter $G(x, y)$,
2. compute the Laplacian of the result as described in the Equations (A.45)-(A.46),
3. eventually, find the zero crossings of the resulting image from the previous step.

[1]Because of the shape illustrated in Figure A.6(a), the Laplacian of Gaussian function sometimes is called the *Mexican hat* operator.

[2]The LoG is a symmetric filter, so spatial filtering using correlation or convolution yields the same result. Moreover, although this expression is implemented in the spatial domain, it can be implemented also in the frequency domain.

A useful approximation of the LoG filter in Equation (A.42) has been found to be represented by the Difference of Gaussians (DoG) of widths σ_1 and σ_2

$$D_{\sigma_1,\sigma_2}(x,y) = \frac{1}{2\pi\sigma_2^2}\exp\left(-\frac{x^2+y^2}{2\sigma_2^2}\right) - \frac{1}{2\pi\sigma_1^2}\exp\left(-\frac{x^2+y^2}{2\sigma_1^2}\right) \tag{A.47}$$

with $\sigma_2 > \sigma_1$. Even though Marr and Hildreth [102] suggested that using the ratio 1.6 preserves the basic characteristics of these observations and also provides a closer "engineering" approximation to the LoG function [55], this value does not yield a satisfactory approximation (possibly that value refers to the ratio of the variances and not of standard deviations). The LoG function in Equation (A.42) has a (negative) peak at the origin equal to

$$L(0,0) = \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] G(x,y)\Big|_{0,0} = -\frac{1}{\pi\sigma^4} \tag{A.48}$$

conversely, the DoG function has a peak value of

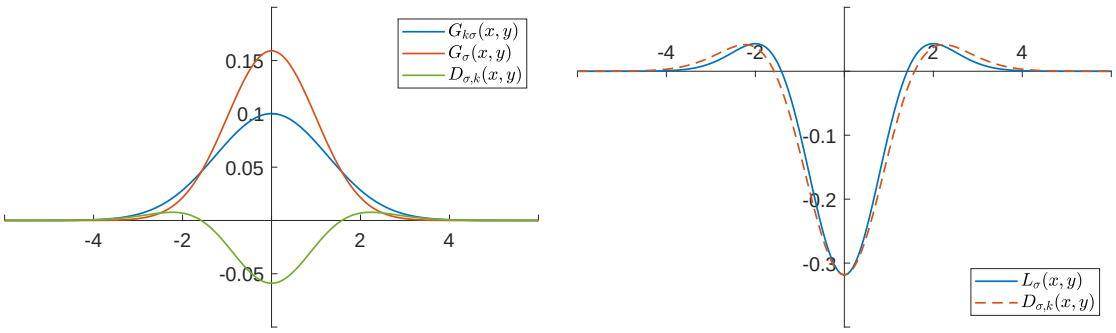
$$D_{\sigma_1,\sigma_2}(0,0) = \frac{1}{2\pi\sigma_2^2}\exp\left(-\frac{x^2+y^2}{2\sigma_2^2}\right) - \frac{1}{2\pi\sigma_1^2}\exp\left(-\frac{x^2+y^2}{2\sigma_1^2}\right)\Big|_{0,0} = \frac{1}{2\pi}\left(\frac{\sigma_1^2-\sigma_2^2}{\sigma_1^2\sigma_2^2}\right). \tag{A.49}$$

Let $k = \frac{\sigma_2}{\sigma_1} > 1$ denote the standard deviation ratio, then the DoG function is as follows

$$\begin{aligned} D_{\sigma,k}(x,y) &= G_{k\sigma}(x,y) - G_{\sigma}(x,y) \\ &= \frac{1}{2\pi k^2\sigma^2}\exp\left(-\frac{x^2+y^2}{2k^2\sigma^2}\right) - \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right). \end{aligned} \tag{A.50}$$

Hence, by scaling the DoG function to match the centre peak value of the LoG function, i.e., $L(0,0) = \lambda D_{\sigma,k}(0,0)$, the Equation (A.42) can be approximated by the DoG in the form

$$L(x,y) \approx \frac{2k^2}{\sigma^2(k^2-1)} D_{\sigma,k}(x,y). \tag{A.51}$$



(a) (b)

Figure A.7: LoG-approximation by the Difference of Gaussians (DoG): (a) the two original Gaussians using a ratio $k^2 = 1.6$ and standard deviation $\sigma = 1$, and (b) LoG in comparison to the DoG function, which has been scaled to match the magnitude of the LoG function.

A.4 Elements of mathematical morphology

Mathematical morphology is area of image-analysis that uses operators developed by means of the set theory [131]. In mathematical morphology we process images according to shape, by treating both as sets of points. In this way, morphological operators define local transformations that change pixel values that are represented as sets.
In binary images, the sets in question are subsets of the two-dimensional integer space $\mathbb{Z}^2$, where points are represented by a pair of integers that give coordinates with respect to the two axes. A morphological transformation is given by the relation between the image and the structuring element.
The **reflection** of a set X, denoted by $\hat{X}$, is as follows

$$\hat{X} = \{w | w = -x, \text{ for } x \in X\} \tag{A.52}$$

where $\hat{X}$ represents the set of points in X whose (x, y) coordinates have been replaced by $(-x, -y)$.
The **translation** of a set X by a point $z = (z_1, z_2)$, denoted by $(X)_z$, is as follows

$$(X)_z = \{c | c = b + z, \text{ for } x \in X\} \tag{A.53}$$

where $(X)_z$ represents the set of points in X whose (x, y) coordinates have been replaced by $(x + z_1, y + z_2)$.

Binary dilation and erosion

The primary morphological operations are dilation and erosion (primitive operations), and from those two, more complex morphological operations can be derived.

Dilation

The morphological transformation dilation $\oplus$ combines two sets X and B in $\mathbb{Z}^2$ as follows

$$X \oplus B = \{z | (B)_z \cap X \neq \varnothing\} \tag{A.54}$$

which means that this equation is based on reflecting B about its origin,and shifting this reflection by z. The dilation of X by B thus is the set of all displacements, such that X and B overlap by at least one element. Equivalently, the Equation (A.54) can be written as:

$$X \oplus B = \{z | [(B)_z \cap X] \subseteq X\}. \tag{A.55}$$

Example 2.

$$X = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\},$$
$$B = \{(0,0), (1,0)\},$$
$$X \oplus B = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4), (2,0), (2,1), (2,2), (3,2), (1,3), (1,4)\}.$$

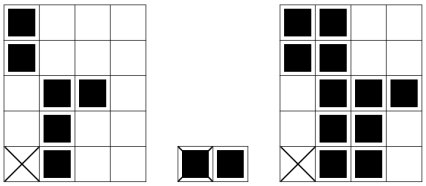


Figure A.8: Example of morphological transformation dilation.

The dilation operation is commutative

$$X \oplus B = B \oplus X \tag{A.56}$$

and is also associative

$$X \oplus (B \oplus D) = (X \oplus B) \oplus D. \tag{A.57}$$

This morphological operation may also be expressed as a union of shifted point sets

$$X \oplus B = \bigcup_{b \in B} X_b \tag{A.58}$$

and is invariant to translation

$$X_h \oplus B = (X \oplus B)_h. \tag{A.59}$$

Moreover, dilation is an increasing transformation

$$X \oplus B \subseteq Y \oplus B \iff X \subseteq Y \tag{A.60}$$

Erosion

The morphological transformation erosion $\ominus$ combines two sets X and B in $\mathbb{Z}^2$ as follows

$$X \ominus B = \{z | (B)_z \subseteq X\} \tag{A.61}$$

which means that this equation indicates that the erosion of X by B is the set of all points z such that B translated by z is contained in X and is the dual of the dilation. Equivalently, the Equation (A.61) can be written as:

$$X \ominus B = \{z | (B)_z \cap X^c = \varnothing\} \tag{A.62}$$

where X^c is the complement of X and $\varnothing$ is the empty set. Neither dilation nor erosion is an invertible transformation.
However, an image X eroded by a structuring element B can be expressed as an intersection of all translations of the image X by the vector $-b \in B$

$$X \oplus B = \bigcap_{b \in B} X_{-b}. \tag{A.63}$$

Erosion is also invariant to translation

$$X_h \ominus B = (X \ominus B)_h, \tag{A.64}$$

$$X \ominus B_h = (X \ominus B)_{-h}, \tag{A.65}$$

and, as the dilation operation, is an increasing transformation

$$X \ominus B \subseteq Y \ominus B \iff X \subseteq Y. \tag{A.66}$$

Example 3.

$$X = \{(1,0),(1,1),(1,2),(0,3),(1,3),(2,3),(3,3),(1,4)\},$$

$$B = \{(0,0),(1,0)\},$$

$$X \ominus B = \{(0,3),(1,3),(2,3)\}.$$

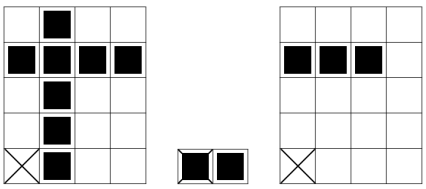


Figure A.9: Example of morphological transformation erosion.

Duality

As mentioned above, erosion and dilation are duals of each other with respect to set complementation and reflection

$$(X \oplus B)^c = X^c \ominus \hat{B} \tag{A.67}$$

and

$$(X \ominus B)^c = X^c \oplus \hat{B} \tag{A.68}$$

where the second equation indicates that the erosion of X by the structuring element B is the complement of the dilation of X^c by B, and vice versa. This property is useful particularly when the structuring element is symmetric with respect to its origin.

Proof. Let us consider the definition of erosion in Equation A.61, then it follows that

$$(X \ominus B)^c = \{z|(B)_z \subseteq X\}^c.$$

If set $(B)_z$ is contained in X, then $(B)_z \cap X^c = \varnothing$, in which case the preceding expression becomes

$$(X \ominus B)^c = \{z|(B)_z \cap X^c = \varnothing\}^c.$$

But the complement of the set of z's that satisfy $(B)_z \cap X^c = \varnothing$ is the set of z's such that $(B)_z \cap X^c = \varnothing$. Therefore,

$$\begin{aligned}(X \ominus B)^c &= \{z|(B)_z \cap X^c \neq \varnothing\} \\ &= X^c \oplus \hat{B}\end{aligned}$$

where the last step follows from Equations (A.54) and (A.55). □

Binary opening and closing

Dilation and erosion are not reversible transformations (once an image is has been eroded and then dilated, the result is not equivalent to the original image). *Closing* operation smoothes the contour of an object, fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour. *Opening* also tends to smooth sections of contours and it generally eliminates thin protrusions [55].
The opening of a set X by the structuring element B is denoted by $X \circ B$ and is defined as

$$X \circ B = (X \ominus B) \oplus B. \tag{A.69}$$

Hence, this operation consists of an erosion of X by B, followed by a dilation of the result by B.
Similarly, the closing of a set X by the structuring element B is denoted by $X \bullet B$ and is defined as

$$X \bullet B = (X \oplus B) \ominus B. \tag{A.70}$$

Hence, this operation consists of a dilation of X by B, followed by an erosion of the result by B.
Unlike erosion and dilation, closing and opening are invariant to translation of the structuring element. In view of the Equations (A.60) and (A.66), both closing and opening are increasing transformations. Closing is extensive ($X \subseteq X \bullet B$)and opening is anti-extensive ($X \circ B \subseteq X$). Moreover, like erosion and dilation, opening and closing are dual transformations:

$$(X \bullet B)^c = X^c \circ \hat{B}. \tag{A.71}$$

A.5 Affine transformations

An affine transform is an automorphism of an affine space, that is, a functional mapping between an affine space onto itself which preserves points, straight and parallel lines as well as ratios between points (but not necessarily distances and angles). Transformation in Cartesian coordinates are known as similarity transforms since they do not change angle values. They define rotations, changes in scale, and translations.
Affine transforms[3] enable basic image transformation, including translation, scaling, homothety, similarity, reflection, rotation, shear mapping, and compositions of them in any combination and sequence. Since translation is not a linear transform, to represent all of the affine transforms by matrices it is possible to use *homogeneous coordinates*:

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} \\ h_{2,1} & h_{2,2} & h_{2,3} \\ h_{3,1} & h_{3,2} & h_{3,3} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}. \tag{A.72}$$

Affine tranformations based on Equation (A.72) are as follows.

Transformation	Affine matrix	Coordinate equations
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x_2 = x_1$ $y_2 = y_1$
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x_2 = c_x x_1$ $y_2 = c_y y_1$
Rotation	$\begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x_2 = x_1 \cos(\theta) - y_1 \sin(\theta)$ $y_2 = x_1 \cos(\theta) + y_1 \sin(\theta)$
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x_2 = x_1 + t_x$ $y_2 = y_1 + t_y$
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x_2 = x_1 + s_v y_1$ $y_2 = y_1$
Shear (vertical)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x_2 = x_1$ $y_2 = s_h x_1 + y_1$

Table A.1: Affine transformations.

[3]Note that a similarity transform is a special case of an affine transform and that an affine transform is a special case of a homography, therefore, similarity and affine transforms can be expressed as homographies.

B

Colour space transformations

This section summarises the transformations between the CIE RGB, CIE XYZ, and CIE L*a*b* colour spaces.
According to the tri-stimulus theory [132], all the possible colours that human perceive can be defined in a three-dimensional linear space. That is, if $[c_1 c_2 c_3]^\top$ define colour components (or weights) and $[A_1 A_2 A_3]^\top$] some base colours (or primaries), then a colour is defined by the colorimetric equation as follows:

$$\mathbf{C} = c_1 A_1 + c_2 A_2 + c_3 A_3 \tag{B.1}$$

where the superposition is expressed as an algebraic sum according to the Grassmann's law of linearity. The equation describes the idea that three stimuli combined by superposition of lights re-create the perception of the colour $\mathbf{C} = [c_1 c_2 c_3]^\top$.
Conversion of RGB image pixel values to the CIE XYZ tri-stimulus values can be achieved using a two stage process. While the CIE RGB model uses visible physical colours, the CIE XYZ uses imaginary colours that only provide a theoretical basis with better descriptive properties.

B.1 CIE RGB to CIE XYZ transformation

The colour components in the XYZ model are obtained from the components of the RGB model by a linear transformation as follows [19, 41]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \underbrace{\begin{bmatrix} 0.49000 & 0.31000 & 0.20000 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00000 & 0.01000 & 0.99000 \end{bmatrix}}_{M} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{B.2}$$

where $\mathbf{M}$ is a non-singular 3×3 matrix. While the above matrix is *exact*, with the number of digits specified in CIE standards. With this version of the matrix $\mathbf{M}$, the Y value corresponding to pure red is 1, however, a more commonly used form is to omit the leading fraction, so that the second row adds up to one, i.e., the RGB triplet $(1, 1, 1)$ maps to a Y value of 1 [138]. Dividing each component XYZ by the sum of their values, we obtain the chromaticity coordinates

$$x = \frac{X}{X+Y+Z},\ y = \frac{Y}{X+Y+Z},\ z = \frac{Z}{X+Y+Z} \tag{B.3}$$

which eventually sum up to one.

B.2 CIE XYZ to CIE RGB transformation

The mapping from the XYZ colour model to the RGB is given by using the inverse of the matrix $\mathbf{M}$, that is not exactly specified, but is approximately [130]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = 0.17697 \underbrace{\begin{bmatrix} 2.36468 & -0.89656 & -0.46811 \\ -0.51527 & 1.42645 & 0.08882 \\ 0.00520 & -0.01441 & 1.00920 \end{bmatrix}}_{\mathbf{M}^{-1}} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{B.4}$$

B.3 CIE L*a*b* to CIE XYZ transformation

Because the response of the human visual system is roughly logarithmic the CIE defined a non-linear re-mapping of the XYZ space called CIE L*a*b* [138]. The lightness, or L* component, is defined as follows:

$$L^* = 116 f\left(\frac{Y}{Y_{ref}}\right) - 16 \tag{B.5}$$

where Y_{ref} is the luminance value of the reference white point and

$$f(\xi) = \begin{cases} \xi^{1/3} & \text{if } \xi > \delta^3, \\ \dfrac{\xi}{3\delta^2} + \dfrac{2\delta}{3} & \text{otherwise.} \end{cases} \tag{B.6}$$

In a similar fashion, the chromaticity components a* and b* are defined as:

$$a^* = 500\left[f\left(\frac{X}{X_{ref}}\right) - f\left(\frac{Y}{Y_{ref}}\right)\right] \qquad b^* = 200\left[f\left(\frac{Y}{Y_{ref}}\right) - f\left(\frac{Z}{Z_{ref}}\right)\right] \tag{B.7}$$

where $(X_{ref}, Y_{ref}, Z_{ref})$ are the values of the reference white point and $\delta = 6/29$.

B.4 CIE XYZ to CIE L*a*b* transformation

To calculate (X, Y, Z) from given CIE L*a*b* colour coordinates

$$Y = Y_{ref} f^{-1}(L'), \quad X = X_{ref} f^{-1}\left(L' + \frac{a^*}{500}\right), \quad Z = Z_{ref} f^{-1}\left(L' + \frac{b^*}{200}\right), \tag{B.8}$$

where

$$L' = \frac{L^* + 16}{116}, \qquad f^{-1}(\xi) = \begin{cases} \xi^3 & \text{if } \xi > \delta \\ 3\delta^2\left(\xi - \dfrac{2\delta}{3}\right) & \text{otherwise} \end{cases} \tag{B.9}$$

where $\delta = 6/29$. Note that the division of the domain of the f function into two parts has been done to prevent an infinite slope at $\xi = 0$.
For Standard Illuminant D65, the reference white point assumes the following values:

$$X_{ref} = 95.0489, \qquad Y_{ref} = 100, \qquad Z_{ref} = 108.8840. \tag{B.10}$$

C

Parameter optimisation experiments

In the following table, a performance comparative analysis of the results for the parameter optimisation of the palm vascular patterns recognition system has been reported. The subset of the database used to perform the experiments, considers half of the right hand samples of all the subjects in the CASIA database acquired in the near-infrared spectrum band at 940 nm.

		GAR at specific value of FAR						Paramter values		
id	**EER**	$\mathbf{GAR_1}$	$\mathbf{GAR_2}$	$\mathbf{GAR_3}$	$\mathbf{GAR_4}$	$\mathbf{GAR_5}$	$\mathbf{GAR_6}$	λ	μ	ν
1	0.002	1.000	0.995	0.993	0.988	0.978	0.953	0.631 2	0.029 5	0.292 0
2	0.003	0.999	0.994	0.993	0.989	0.974	0.946	0.548 7	0.034 6	0.423 0
3	0.003	0.998	0.994	0.990	0.983	0.972	0.940	0.254 0	0.049 6	0.676 1
4	0.004	0.999	0.994	0.990	0.988	0.972	0.938	0.229 8	0.055 8	0.637 0
5	0.004	1.000	0.994	0.993	0.988	0.963	0.936	0.429 8	0.042 4	0.464 3
6	0.005	0.999	0.994	0.992	0.987	0.969	0.935	0.094 8	0.070 2	0.635 3
7	0.006	0.999	0.994	0.991	0.985	0.961	0.935	0.287 4	0.049 5	0.604 9
8	0.006	0.999	0.994	0.990	0.983	0.938	0.912	0.471 4	0.033 4	0.520 0
9	0.007	0.996	0.994	0.991	0.983	0.972	0.951	0.387 3	0.048 9	0.602 7
10	0.007	0.998	0.994	0.990	0.988	0.971	0.945	0.279 6	0.052 9	0.618 4
11	0.007	0.998	0.994	0.993	0.986	0.943	0.935	0.266 7	0.056 9	0.479 0
12	0.007	0.999	0.994	0.992	0.983	0.950	0.910	0.323 4	0.040 4	0.624 1
13	0.007	0.998	0.994	0.989	0.983	0.942	0.903	0.381 1	0.044 6	0.468 7
14	0.007	0.999	0.993	0.990	0.983	0.950	0.899	0.138 9	0.051 9	0.704 2
15	0.007	0.998	0.993	0.992	0.982	0.932	0.898	0.096 1	0.070 0	0.510 8
16	0.007	0.998	0.995	0.992	0.980	0.929	0.893	0.067 7	0.073 6	0.481 6
17	0.007	0.998	0.995	0.986	0.975	0.905	0.867	0.430 7	0.043 8	0.326 1
18	0.008	0.996	0.993	0.988	0.978	0.956	0.932	0.177 4	0.044 2	0.765 4
19	0.008	0.996	0.993	0.988	0.975	0.953	0.926	0.108 5	0.058 8	0.752 1
20	0.008	0.998	0.994	0.983	0.969	0.887	0.854	0.184 4	0.061 0	0.465 8
21	0.009	0.997	0.993	0.988	0.977	0.954	0.927	0.151 4	0.046 9	0.767 9
22	0.009	0.998	0.993	0.983	0.968	0.903	0.842	0.069 9	0.057 3	0.684 5
23	0.009	0.998	0.993	0.981	0.968	0.866	0.839	0.062 3	0.064 8	0.603 1
24	0.009	0.999	0.993	0.976	0.955	0.858	0.837	0.367 2	0.048 3	0.244 1
25	0.010	0.995	0.991	0.986	0.973	0.943	0.917	0.203 1	0.045 8	0.762 1
26	0.010	0.998	0.990	0.981	0.963	0.888	0.836	0.177 5	0.061 8	0.441 1

		GAR at specific value of FAR						Paramter values		
id	EER	GAR_1	GAR_2	GAR_3	GAR_4	GAR_5	GAR_6	λ	μ	ν
27	0.011	0.996	0.991	0.973	0.951	0.849	0.807	0.114 8	0.070 2	0.279 7
28	0.012	0.996	0.989	0.975	0.953	0.865	0.803	0.126 2	0.049 2	0.685 3
29	0.013	0.995	0.988	0.968	0.945	0.813	0.795	0.174 0	0.053 5	0.580 0
30	0.013	0.997	0.988	0.972	0.939	0.833	0.775	0.437 9	0.036 2	0.455 9
31	0.013	0.998	0.986	0.968	0.933	0.833	0.733	0.624 3	0.024 3	0.327 0
32	0.014	0.995	0.985	0.958	0.922	0.772	0.742	0.538 7	0.032 7	0.284 5
33	0.016	0.994	0.984	0.960	0.926	0.776	0.748	0.232 7	0.045 6	0.060 3
34	0.016	0.995	0.983	0.958	0.923	0.816	0.717	0.483 5	0.039 5	0.194 0
35	0.017	0.996	0.977	0.957	0.908	0.825	0.748	0.102 8	0.039 3	0.373 2
36	0.018	0.997	0.974	0.957	0.901	0.830	0.764	0.184 5	0.037 3	0.673 8
37	0.018	0.994	0.980	0.961	0.917	0.829	0.734	0.254 4	0.049 5	0.500 8
38	0.018	0.994	0.978	0.956	0.902	0.753	0.718	0.344 1	0.048 9	0.282 8
39	0.019	0.994	0.980	0.945	0.902	0.748	0.717	0.381 1	0.048 2	0.169 1
40	0.019	0.995	0.974	0.944	0.882	0.759	0.622	0.807 6	0.014 0	0.100 4
41	0.021	0.993	0.971	0.952	0.895	0.786	0.698	0.097 9	0.069 7	0.244 1
42	0.021	0.994	0.975	0.952	0.900	0.787	0.669	0.256 8	0.033 8	0.687 3
43	0.027	0.992	0.964	0.910	0.841	0.718	0.623	0.255 7	0.040 6	0.058 4
44	0.029	0.989	0.958	0.906	0.844	0.665	0.622	0.432 5	0.039 7	0.297 9
45	0.029	0.990	0.955	0.893	0.808	0.680	0.602	0.095 0	0.043 4	0.062 1
46	0.030	0.990	0.962	0.929	0.849	0.737	0.688	0.098 7	0.059 9	0.532 8
47	0.030	0.988	0.962	0.912	0.821	0.673	0.567	0.034 9	0.046 4	0.739 3
48	0.032	0.983	0.957	0.895	0.814	0.686	0.602	0.135 6	0.068 5	0.037 6
49	0.033	0.987	0.955	0.898	0.826	0.703	0.614	0.056 1	0.071 8	0.205 0
50	0.033	0.985	0.952	0.903	0.807	0.714	0.595	0.089 9	0.070 6	0.114 7
51	0.037	0.981	0.942	0.869	0.790	0.658	0.582	0.004 8	0.074 8	0.234 1
52	0.037	0.983	0.947	0.868	0.758	0.649	0.552	0.205 2	0.055 7	0.366 8
53	0.037	0.984	0.940	0.880	0.743	0.579	0.493	0.502 5	0.037 9	0.088 9
54	0.037	0.990	0.946	0.838	0.728	0.602	0.429	0.052 6	0.048 0	0.298 9
55	0.039	0.981	0.938	0.861	0.758	0.623	0.531	0.076 1	0.065 7	0.379 9
56	0.039	0.980	0.934	0.854	0.738	0.634	0.529	0.213 5	0.057 5	0.256 9
57	0.040	0.980	0.934	0.848	0.729	0.627	0.520	0.169 2	0.053 7	0.502 2
58	0.044	0.975	0.922	0.833	0.719	0.602	0.491	0.026 8	0.069 7	0.360 7
59	0.044	0.973	0.923	0.830	0.704	0.598	0.487	0.455 2	0.023 9	0.355 3
60	0.044	0.971	0.923	0.826	0.689	0.594	0.482	0.138 7	0.057 9	0.449 1
61	0.045	0.981	0.920	0.805	0.673	0.542	0.398	0.505 6	0.034 0	0.236 5
62	0.046	0.972	0.917	0.826	0.701	0.575	0.470	0.067 8	0.068 2	0.266 0
63	0.048	0.968	0.911	0.792	0.658	0.554	0.444	0.158 6	0.061 8	0.206 7
64	0.048	0.968	0.907	0.818	0.652	0.505	0.399	0.216 4	0.033 2	0.698 7
65	0.049	0.972	0.923	0.844	0.780	0.668	0.607	0.017 7	0.040 8	0.441 3
66	0.050	0.968	0.907	0.813	0.664	0.559	0.437	0.218 1	0.054 6	0.325 9
67	0.052	0.968	0.904	0.782	0.678	0.537	0.423	0.142 9	0.065 3	0.070 1
68	0.053	0.970	0.880	0.772	0.574	0.470	0.279	0.479 6	0.028 6	0.459 4
69	0.054	0.970	0.892	0.785	0.631	0.533	0.403	0.182 8	0.060 9	0.134 6
70	0.054	0.980	0.889	0.686	0.511	0.293	0.140	0.079 5	0.044 9	0.355 6
71	0.058	0.965	0.891	0.728	0.602	0.477	0.308	0.230 6	0.040 2	0.617 3

		GAR at specific value of FAR						Paramter values		
id	**EER**	**GAR_1**	**GAR_2**	**GAR_3**	**GAR_4**	**GAR_5**	**GAR_6**	**λ**	**μ**	**ν**
72	0.058	0.978	0.833	0.655	0.379	0.216	0.143	0.061 2	0.045 1	0.157 1
73	0.058	0.978	0.833	0.655	0.379	0.216	0.143	0.162 3	0.044 7	0.210 6
74	0.059	0.958	0.896	0.792	0.691	0.492	0.407	0.117 6	0.055 5	0.823 4
75	0.060	0.968	0.792	0.512	0.340	0.120	0.089	0.345 3	0.028 7	0.382 8
76	0.061	0.972	0.852	0.687	0.487	0.305	0.162	0.334 3	0.032 0	0.343 8
77	0.061	1.000	1.000	0.904	0.808	0.808	0.086	0.198 5	0.035 6	0.532 7
78	0.062	0.960	0.837	0.546	0.288	0.090	0.068	0.080 2	0.054 3	0.009 3
79	0.062	0.446	0.200	0.178	0.167	0.157	0.136	0.158 0	0.043 0	0.411 5
80	0.063	0.960	0.837	0.704	0.636	0.578	0.470	0.596 4	0.025 3	0.148 2
81	0.063	0.958	0.870	0.656	0.453	0.292	0.210	0.648 6	0.024 2	0.125 5
82	0.069	0.948	0.863	0.713	0.570	0.460	0.348	0.046 5	0.068 7	0.233 4
83	0.069	0.945	0.838	0.706	0.550	0.421	0.280	0.184 7	0.059 6	0.149 2
84	0.070	0.947	0.871	0.727	0.607	0.462	0.368	0.009 5	0.057 1	0.838 3
85	0.070	0.952	0.828	0.697	0.508	0.354	0.195	0.362 5	0.037 0	0.473 2
86	0.070	0.968	0.801	0.597	0.434	0.236	0.094	0.155 4	0.043 2	0.113 3
87	0.072	0.949	0.806	0.639	0.451	0.302	0.193	0.431 4	0.026 7	0.385 6
88	0.073	0.949	0.758	0.551	0.354	0.133	0.081	0.611 6	0.018 4	0.135 9
89	0.075	0.958	0.766	0.539	0.281	0.118	0.071	0.363 9	0.032 4	0.398 6
90	0.076	0.933	0.807	0.643	0.531	0.379	0.278	0.010 8	0.066 2	0.437 5
91	0.076	0.949	0.811	0.643	0.447	0.284	0.172	0.326 6	0.048 5	0.135 5
92	0.076	0.960	0.777	0.561	0.353	0.223	0.135	0.449 3	0.022 1	0.277 3
93	0.077	0.954	0.789	0.597	0.402	0.254	0.159	0.248 6	0.036 5	0.359 4
94	0.077	0.948	0.800	0.633	0.450	0.285	0.183	0.340 6	0.028 0	0.193 7
95	0.078	0.946	0.771	0.574	0.381	0.265	0.161	0.013 4	0.041 9	0.672 7
96	0.079	0.965	0.793	0.621	0.449	0.277	0.135	0.097 2	0.040 6	0.651 7
97	0.080	0.948	0.770	0.515	0.290	0.168	0.083	0.218 0	0.039 3	0.455 5
98	0.080	0.963	0.758	0.515	0.340	0.203	0.071	0.141 4	0.043 0	0.428 2
99	0.081	0.442	0.229	0.196	0.145	0.108	0.071	0.076 4	0.039 1	0.565 4
100	0.082	0.932	0.791	0.554	0.405	0.231	0.106	0.932 0	0.003 3	0.032 5
101	0.082	0.951	0.782	0.534	0.306	0.180	0.072	0.050 6	0.038 3	0.185 2
102	0.083	0.935	0.767	0.595	0.397	0.227	0.143	0.040 2	0.059 7	0.127 3
103	0.083	0.932	0.773	0.580	0.410	0.237	0.132	0.396 0	0.042 2	0.164 8
104	0.084	0.935	0.748	0.553	0.353	0.192	0.098	0.584 7	0.021 3	0.401 9
105	0.087	0.937	0.778	0.563	0.348	0.179	0.139	0.683 7	0.017 2	0.282 0
106	0.087	0.938	0.756	0.418	0.226	0.082	0.047	0.130 8	0.040 6	0.510 2
107	0.090	0.944	0.712	0.519	0.294	0.144	0.071	0.064 9	0.039 5	0.233 8
108	0.091	0.932	0.732	0.397	0.213	0.067	0.043	0.550 5	0.019 6	0.212 9
109	0.092	0.927	0.740	0.404	0.222	0.078	0.034	0.283 9	0.042 1	0.051 0
110	0.095	0.926	0.723	0.556	0.392	0.252	0.162	0.054 6	0.065 7	0.257 9
111	0.101	0.925	0.723	0.445	0.232	0.095	0.049	0.272 0	0.030 5	0.410 3
112	0.102	0.911	0.683	0.475	0.296	0.127	0.106	0.553 8	0.032 2	0.040 8
113	0.102	0.927	0.738	0.511	0.338	0.162	0.074	0.384 1	0.033 0	0.493 4
114	0.103	0.921	0.732	0.538	0.353	0.189	0.101	0.138 5	0.040 4	0.676 5
115	0.105	0.899	0.663	0.475	0.252	0.113	0.088	0.714 0	0.019 4	0.067 2
116	0.106	0.901	0.709	0.461	0.268	0.126	0.056	0.129 3	0.043 7	0.520 0

		GAR at specific value of FAR						Paramter values		
id	EER	GAR_1	GAR_2	GAR_3	GAR_4	GAR_5	GAR_6	λ	μ	ν
117	0.106	0.901	0.709	0.461	0.268	0.126	0.056	0.771 9	0.012 1	0.095 0
118	0.109	0.912	0.663	0.421	0.193	0.055	0.033	0.709 2	0.017 9	0.044 9
119	0.115	0.883	0.680	0.492	0.334	0.193	0.121	0.067 5	0.063 3	0.293 1
120	0.116	0.883	0.678	0.493	0.329	0.189	0.118	0.078 1	0.060 8	0.370 8
121	0.117	0.884	0.675	0.488	0.326	0.183	0.118	0.028 2	0.054 9	0.613 8
122	0.119	0.893	0.655	0.440	0.216	0.114	0.069	0.478 1	0.033 3	0.234 3
123	0.122	0.848	0.644	0.345	0.197	0.058	0.029	0.236 3	0.035 4	0.380 6
124	0.123	0.843	0.511	0.309	0.207	0.120	0.003	0.329 7	0.041 0	0.175 2
125	0.126	0.898	0.651	0.434	0.269	0.139	0.048	0.205 0	0.034 1	0.673 6
126	0.133	0.858	0.630	0.417	0.265	0.161	0.090	0.126 1	0.062 5	0.048 6
127	0.134	0.847	0.660	0.405	0.262	0.144	0.058	0.051 5	0.040 9	0.723 9
128	0.135	0.847	0.466	0.269	0.087	0.015	0.015	0.126 3	0.067 1	0.836 9
129	0.138	0.873	0.626	0.395	0.223	0.135	0.072	0.207 5	0.056 3	0.052 9
130	0.145	0.869	0.580	0.340	0.203	0.118	0.037	0.240 2	0.039 2	0.570 0
131	0.146	0.861	0.615	0.396	0.236	0.151	0.064	0.047 7	0.046 7	0.437 6
132	0.150	0.821	0.551	0.354	0.208	0.066	0.038	0.385 2	0.040 2	0.199 7
133	0.152	0.725	0.499	0.262	0.106	0.028	0.007	0.341 6	0.033 8	0.263 6
134	0.155	0.733	0.517	0.524	0.212	0.056	0.037	0.377 0	0.029 2	0.012 3
135	0.158	0.832	0.606	0.351	0.180	0.096	0.038	0.229 6	0.042 9	0.509 9
136	0.159	0.818	0.585	0.334	0.168	0.083	0.043	0.272 8	0.045 5	0.333 6
137	0.159	0.833	0.574	0.371	0.194	0.083	0.028	0.576 4	0.023 6	0.175 4
138	0.160	0.792	0.512	0.272	0.162	0.045	0.030	0.583 1	0.023 2	0.289 2
139	0.163	0.860	0.782	0.546	0.288	0.126	0.096	0.743 8	0.012 5	0.049 7
140	0.163	0.822	0.539	0.331	0.148	0.058	0.028	0.391 4	0.042 3	0.046 8
141	0.165	0.829	0.595	0.387	0.190	0.078	0.048	0.194 2	0.033 1	0.334 0
142	0.170	0.810	0.519	0.338	0.176	0.071	0.039	0.151 9	0.042 9	0.612 5
143	0.173	0.811	0.533	0.305	0.123	0.032	0.014	0.169 9	0.042 3	0.502 3
144	0.182	0.775	0.495	0.215	0.092	0.022	0.007	0.455 4	0.038 0	0.007 5
145	0.185	0.756	0.496	0.226	0.108	0.032	0.017	0.398 2	0.037 5	0.233 1
146	0.186	0.758	0.518	0.297	0.156	0.066	0.032	0.071 0	0.061 3	0.233 4
147	0.187	0.740	0.487	0.222	0.107	0.034	0.012	0.366 6	0.041 6	0.130 3
148	0.188	0.779	0.511	0.275	0.125	0.049	0.022	0.190 3	0.053 8	0.160 4
149	0.193	0.732	0.475	0.213	0.097	0.028	0.013	0.365 4	0.037 5	0.328 6
150	0.202	0.733	0.421	0.233	0.087	0.022	0.011	0.217 3	0.039 9	0.555 0
151	0.205	0.710	0.410	0.197	0.101	0.029	0.010	0.170 1	0.054 0	0.183 0
152	0.206	0.657	0.358	0.096	0.033	0.005	0.002	0.623 0	0.020 6	0.208 7
153	0.213	0.680	0.385	0.176	0.081	0.024	0.009	0.117 0	0.052 2	0.427 0
154	0.232	0.616	0.305	0.159	0.050	0.014	0.004	0.032 5	0.039 6	0.717 2
155	0.271	0.637	0.472	0.307	0.142	0.097	0.041	0.255 6	0.031 0	0.612 7
156	0.273	0.467	0.270	0.198	0.092	0.049	0.016	0.234 4	0.034 6	0.362 3
157	0.289	0.462	0.261	0.197	0.162	0.134	0.070	0.141 2	0.037 5	0.204 5
158	0.304	0.457	0.253	0.197	0.185	0.144	0.110	0.272 1	0.032 2	0.173 4
159	0.306	0.438	0.169	0.169	0.169	0.169	0.169	0.022 6	0.059 0	0.874 7
160	0.309	0.507	0.313	0.231	0.165	0.147	0.107	0.225 4	0.041 6	0.023 6
161	0.313	0.445	0.273	0.207	0.181	0.173	0.122	0.081 6	0.050 7	0.882 4

		GAR at specific value of FAR						Paramter values		
id	**EER**	**GAR_1**	**GAR_2**	**GAR_3**	**GAR_4**	**GAR_5**	**GAR_6**	**λ**	**μ**	**ν**
162	0.317	0.595	0.502	0.449	0.369	0.296	0.223	0.059 0	0.072 5	0.926 1
163	0.320	0.452	0.245	0.197	0.205	0.156	0.113	0.195 4	0.036 4	0.420 7
164	0.336	0.447	0.237	0.197	0.145	0.114	0.080	0.310 9	0.029 5	0.064 4
165	0.356	0.491	0.296	0.206	0.139	0.118	0.085	0.393 0	0.030 5	0.426 3
166	0.403	0.475	0.278	0.180	0.113	0.089	0.063	0.281 0	0.037 1	0.030 0
167	0.410	0.437	0.220	0.196	0.140	0.121	0.095	0.619 7	0.015 3	0.092 7
168	0.414	0.432	0.212	0.193	0.157	0.124	0.092	0.400 1	0.025 4	0.233 8
169	0.420	0.392	0.147	0.117	0.087	0.057	0.028	0.050 9	0.050 4	0.910 7
170	0.430	0.397	0.155	0.126	0.097	0.068	0.039	0.008 0	0.073 2	0.899 7
171	0.432	0.402	0.163	0.135	0.107	0.079	0.051	0.021 4	0.042 3	0.890 6
172	0.436	0.422	0.196	0.171	0.146	0.121	0.096	0.214 4	0.039 7	0.418 8
173	0.441	0.427	0.204	0.178	0.152	0.126	0.100	0.099 0	0.052 9	0.331 4
174	0.445	0.407	0.171	0.144	0.117	0.089	0.062	0.261 4	0.043 5	0.129 3
175	0.447	0.417	0.188	0.162	0.136	0.110	0.085	0.622 2	0.017 6	0.293 3
176	0.458	0.412	0.179	0.153	0.126	0.100	0.073	0.073 6	0.054 4	0.360 5

Table C.1: Performance comparative analysis of the results for the parameter optimisation of the palm vascular patterns recognition system.

List of publications

A threshold mechanism ensures minimum-path flow in lightning discharge

F. Blanchini, D. Casagrande, F. Fabiani, G. Giordano, **D. Palma**, and R. Pesenti, "A threshold mechanism ensures minimum-path flow in lightning discharge", *Scientific Reports (Nature)*, 11, 280 (2021)
doi: 10.1038/s41598-020-79463-z

Abstract A well-known property of linear resistive electrical networks is that the current distribution minimizes the total dissipated power. When the circuit includes resistors with nonlinear monotonic characteristic, the current distribution minimizes in general a different functional. We show that, if the nonlinear characteristic is a threshold-like function and the current generator is concentrated in a single point, as in the case of lightning or dielectric discharge, then the current flow is concentrated along a single path, which is a minimum path to the ground with respect to the threshold. We also propose a dynamic model that explains and qualitatively reproduces the lightning transient behavior: initial generation of several plasma branches and subsequent dismissal of all branches but the one reaching the ground first, which is the optimal one.

A dynamic biometric authentication algorithm for near-infrared palm vascular patterns

D. Palma, F. Blanchini, G. Giordano, and P. L. Montessoro, "A dynamic biometric authentication algorithm for near-infrared palm vascular patterns", *IEEE Access*, vol. 8, pp. 118978–118988, 2020.
doi: 10.1109/ACCESS.2020.3005460

Abstract In this paper we apply a novel approach to near-infrared subcutaneous palm vascular pattern authentication. The proposed method relies on a recursive algorithm based on a positive linear dynamical system whose evolution depends on the two matrices representing the vein patterns to be compared. The output of the system reaches a high value when a good matching between the two matrices is observed, otherwise it converges rapidly to zero, even in presence of noise. With respect to another algorithm we recently introduced, this approach achieves not only a better authentication performance but also a drastic reduction in terms of computation time. These improvements are demonstrated by means of extensive experiments conducted on challenging datasets.

Biometric palmprint verification: A dynamical system approach

D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "Biometric palmprint verification: A dynamical system approach", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 12, pp. 2676–2687, 2019.
doi: 10.1109/tsmc.2017.2771232

Abstract Most of the existing techniques for palmprint recognition rely on metrics, typically based on static functions, which evaluate the distance between a pair of features. In this paper, we propose a new technique for palmprint verification based on a dynamical system approach for principal palm lines matching. The proposed dynamic algorithm is recursive and involves a positive linear dynamical system, whose evolution depends on the matching level between the two input images. In a preprocessing phase, the procedure iteratively erodes both of the images to be compared, by eliminating points in each image that do not have enough close neighboring points both in the image itself and the comparison image. As a result of the iterations, only the points that have enough neighboring points in both the image itself and in the comparison image can survive. Thus, the output of the dynamical system converges either to zero, when a deep mis- match exists between the two images, or to a high value, when a good matching is observed. The results, in terms of verification, are in line with the state-of-the-art results in the current literature. The main advantage of the approach is its robustness when dealing with low-resolution and noisy images. The impact of noise (e.g., salt and pepper noise) is effectively reduced: images corrupted with such noise are easily recognized, while a randomly generated image is rejected even when compared with itself.

A dynamic algorithm for palmprint recognition

D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "A dynamic algorithm for palmprint recognition", *IEEE Conference on Communications and Network Security (CNS)*, pp. 659–662, 2015.
doi: 10.1109/cns.2015.7346883

Abstract Most of the existing techniques for palmprint recognition are based on metrics that evaluate the distance between a pair of features. These metrics are typically based on static functions. In this paper we propose a new technique for palmprint recognition based on a dynamical system approach, focusing on preliminary experimental results. The essential idea is that the procedure iteratively eliminates points in both images to be compared which do not have enough close neighboring points in the image itself and in the comparison image. As a result of the iteration, in each image the surviving points are those having enough neighboring points in the comparison image. Our preliminary experimental results show that the proposed dynamic algorithm is competitive and slightly outperforms some state–of– the–art methods by achieving a higher genuine acceptance rate.

Bibliography

[1] G. N. Agrios, *Plant pathology*, 5th ed. Academic Press, 2005.

[2] F. Ahmad, L.-M. Cheng, and A. Khan, "Lightweight and privacy-preserving template generation for palm-vein-based human recognition," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 184–194, 2020.

[3] A. H. Akkermans, T. A. Kevenaar, and D. W. Schobben, "Acoustic ear recognition for person identification," in *Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID)*, 2005, pp. 219–223.

[4] A. M. Al-juboori, W. Bu, X. Wu, and Q. Zhao, "Palm vein verification using gabor filter," *International Journal of Computer Science Issues*, vol. 10, no. 1, pp. 678–684, 2013.

[5] A. D. Andrushia and A. T. Patricia, "Artificial bee colony optimization (ABC) for grape leaves disease detection," *Evolving Systems*, vol. 11, no. 1, pp. 105–117, 2020.

[6] D. Angeli and E. D. Sontag, "Monotone control systems," *IEEE Transactions on automatic control*, vol. 48, no. 10, pp. 1684–1698, 2003.

[7] G. Armijo, R. Schlechter, M. Agurto, D. Muñoz, C. Nuñez, and P. Arce-Johnson, "Grapevine pathogenic microorganisms: understanding infection strategies and host response scenarios," *Frontiers in plant science*, vol. 7, p. 382, 2016.

[8] J. G. A. Barbedo, "Digital image processing techniques for detecting, quantifying and classifying plant diseases," *SpringerPlus*, vol. 2, no. 1, pp. 1–12, 2013.

[9] J. G. A. Barbedo, "An automatic method to detect and measure leaf disease symptoms using digital image processing," *Plant Disease*, vol. 98, no. 12, pp. 1709–1716, 2014.

[10] J. G. A. Barbedo, "A new automatic method for disease symptom segmentation in digital photographs of plant leaves," *European journal of plant pathology*, vol. 147, no. 2, pp. 349–364, 2017.

[11] A. Berman and R. J. Plemmons, *Nonnegative matrices in the mathematical sciences*. SIAM, 1994.

[12] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008.

[13] F. Blanchini, P. Colaneri, and M. E. Valcher, *Switched positive linear systems*. Now Publishers, 2015.

[14] C. Bock, G. Poole, P. Parker, and T. Gottwald, "Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging," *Critical Reviews in Plant Sciences*, vol. 29, no. 2, pp. 59–107, 2010.

[15] T. Boekestijn and R. Visser, "A comparison of state-of-the-art vision-based biometric analysis methods," *Proceedings of the 15th SC@RUG 2018*, p. 59, 2018.

[16] W. W. Boles and S. Chu, "Personal identification using images of the human palm," in *Proceedings of IEEE TENCON Speech and Image Technologies for Computing and Telecommunications*, vol. 1, 1997, pp. 295–298.

[17] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using matthews correlation coefficient metric," *PloS one*, vol. 12, no. 6, p. e0177678, 2017.

[18] N. V. Boulgouris, K. N. Plataniotis, and E. Micheli-Tzanakou, *Biometrics: theory, methods, and applications*. John Wiley & Sons, 2009, vol. 9.

[19] M. H. Brill, "How the cie 1931 color-matching functions were derived from wright-guild data," *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, vol. 23, no. 4, pp. 259–259, 1998.

[20] A. Bruno, P. Carminetti, V. Gentile, M. La Cascia, and E. Mancino, "Palmprint principal lines extraction," in *IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications Proceedings*, 2014, pp. 50–56.

[21] T. J. Bruno and P. D. Svoronos, *CRC handbook of fundamental spectroscopic correlation charts*. CRC Press, 2005.

[22] H. Cai, J. Caswell, and J. Prescott, "Nonculture molecular techniques for diagnosis of bacterial disease in animals: a diagnostic laboratory perspective," *Veterinary pathology*, vol. 51, no. 2, pp. 341–350, 2014.

[23] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[24] R. Cappelli, M. Ferrara, and D. Maio, "A fast and accurate palmprint recognition system based on minutiae," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 3, pp. 956–962, 2012.

[25] H. Chen, G. Lu, and R. Wang, "A new palm vein matching method based on icp algorithm," in *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, 2009, pp. 1207–1211.

[26] D. Chicco and G. Jurman, "The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation," *BMC genomics*, vol. 21, no. 1, p. 6, 2020.

[27] Chinese Academy of Sciences, Institute of Automation (CASIA), "Casia multi-spectral palmprint image database v1.0," http://biometrics.idealtest.org/.

[28] Chinese Academy of Sciences, Institute of Automation (CASIA), "Casia palmprint image database," http://biometrics.idealtest.org/.

[29] T. Connie, A. Teoh, M. Goh, and D. Ngo, "Palmprint recognition with pca and ica," in *Proceedings in Image and Vision Computing*, 2003.

[30] R. Das, E. Piciucco, E. Maiorana, and P. Campisi, "Convolutional neural network for finger-vein-based biometric identification," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 360–373, 2019.

[31] S. Das Choudhury, A. Samal, and T. Awada, "Leveraging image analysis for high-throughput plant phenotyping," *Frontiers in plant science*, vol. 10, p. 508, 2019.

[32] S. C. Dass, Y. Zhu, and A. K. Jain, "Validating a biometric authentication system: Sample size requirements," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1902–1319, 2006.

[33] J. Daugman, "How iris recognition works," in *The essential guide to image processing*. Elsevier, 2009, pp. 715–739.

[34] K. Delac and M. Grgic, "A survey of biometric recognition methods," in *Proceedings Elmar-2004, 46th International Symposium on Electronics in Marine*, 2004, pp. 184–193.

[35] M. R. Deore and S. M. Handore, "A survey on offline signature recognition and verification schemes," in *International conference on industrial instrumentation and control (ICIC)*, 2015, pp. 165–169.

[36] G. Dhingra, V. Kumar, and H. D. Joshi, "Study of digital image processing techniques for leaf disease detection and classification," *Multimedia Tools and Applications*, vol. 77, no. 15, pp. 19 951–20 000, 2018.

[37] M. R. Diaz, C. M. Travieso, J. B. Alonso, and M. A. Ferrer, "Biometric system based in the feature of hand palm," in *38th Annual 2004 International Carnahan Conference on Security Technology*, 2004, pp. 136–139.

[38] M. Ekinci and M. Aykut, "Palmprint recognition by applying wavelet-based kernel pca," *Journal of Computer Science and Technology*, vol. 23, no. 5, pp. 851–861, 2008.

[39] S. Elnasir, S. M. Shamsuddin, and S. Farokhi, "Accurate palm vein recognition based on wavelet scattering and spectral regression kernel discriminant analysis," *Journal of Electronic Imaging*, vol. 24, no. 1, pp. 1–14, 2015.

[40] A. J.-C. Eun, L. Huang, F.-T. Chew, S. F.-Y. Li, and S.-M. Wong, "Detection of two orchid viruses using quartz crystal microbalance (qcm) immunosensors," *Journal of Virological Methods*, vol. 99, no. 1-2, pp. 71–79, 2002.

[41] H. S. Fairman, M. H. Brill, and H. Hemmendinger, "How the cie 1931 color-matching functions were derived from wright-guild data," *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, vol. 22, no. 1, pp. 11–23, 1997.

[42] M. Fang, G. Yue, and Q. Yu, "The study on an application of otsu method in canny operator," in *International Symposium on Information Processing*, 2009, pp. 109–112.

[43] L. Farina, "Positive systems in the state space approach: main issues and recent results," in *Proceedings of MTNS*, 2002, pp. 14 900–1.

[44] Federal Bureau of Investigation, "Next generation identification (ngi)," https://www.fbi.gov/services/cjis/fingerprints-and-other-biometrics/ngi.

[45] L. Fei, Y. Xu, W. Tang, and D. Zhang, "Double-orientation code and nonlinear matching scheme for palmprint recognition," *Pattern Recognition*, vol. 49, no. C, pp. 89–101, 2016.

[46] L. Fei, Y. Xu, and D. Zhang, "Half-orientation extraction of palmprint features," *Pattern Recognition Letters*, vol. 69, pp. 35–41, 2016.

[47] J. Fierrez, A. Pozo, M. Martinez-Diaz, J. Galbally, and A. Morales, "Benchmarking touchscreen biometrics for mobile authentication," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2720–2733, 2018.

[48] A. Furusawa, "Quantum states of light," in *Quantum States of Light*. Springer, 2015, pp. 1–67.

[49] G. Gan, C. Ma, and J. Wu, *Data clustering: theory, algorithms, and applications*. SIAM, 2020.

[50] Y. Gao and M. K. Leung, "Face recognition using line edge map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 764–779, 2002.

[51] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1747–1754, 2005.

[52] D. Gawkrodger and M. R. Ardern-Jones, *Dermatology: An Illustrated Colour Text*. Elsevier Health Sciences, 2016.

[53] S. N. Ghaiwat and P. Arora, "Detection and classification of plant leaf diseases using image processing techniques: a review," *International Journal of Recent Advances in Engineering & Technology*, vol. 2, no. 3, pp. 1–7, 2014.

[54] E. Gomès and P. Coutos-Thévenot, "Molecular aspects of grapevine-pathogenic fungi interactions," in *Grapevine molecular physiology & biotechnology*. Springer, 2009, pp. 407–428.

[55] R. Gonzalez and R. Woods, *Digital Image Processing*. Pearson, 2018.

[56] Government Office for Science, "Biometrics: a guide," https://www.gov.uk/go-science.

[57] P. E. Green, *Mathematical tools for applied multivariate analysis*. Academic Press, 2014.

[58] C.-C. Han, H.-L. Cheng, C.-L. Lin, and K.-C. Fan, "Personal authentication using palm-print features," *Pattern Recognition*, vol. 36, no. 2, pp. 371–381, 2003.

[59] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[60] D. Hong, W. Liu, J. Su, Z. Pan, and G. Wang, "A novel hierarchical approach for multispectral palmprint recognition," *Neurocomputing*, vol. 151, pp. 511–521, 2015.

[61] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777–789, 1998.

[62] Hong Kong Polytechnic University (PolyU), Biometric Research Centre, "Polyu multispectral palmprint database," https://www.comp.polyu.edu.hk/~biometrics/MultispectralPalmprint/MSP.htm.

[63] Hong Kong Polytechnic University (PolyU), Biometric Research Centre, "Polyu palmprint database," https://www.comp.polyu.edu.hk/~biometrics/.

[64] R. A. Horn, R. A. Horn, and C. R. Johnson, *Topics in matrix analysis*. Cambridge university press, 1994.

[65] D. Houle, D. R. Govindaraju, and S. Omholt, "Phenomics: the next challenge," *Nature reviews genetics*, vol. 11, no. 12, pp. 855–866, 2010.

[66] D.-S. Huang, W. Jia, and D. Zhang, "Palmprint verification based on principal lines," *Pattern Recognition*, vol. 41, no. 4, pp. 1316–1328, 2008.

[67] D. Huang, Y. Tang, Y. Wang, L. Chen, and Y. Wang, "Hand-dorsa vein recognition by matching local features of multisource keypoints," *IEEE Transactions on Cybernetics*, vol. 45, no. 9, pp. 1823–1837, 2015.

[68] D. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2015.

[69] ISO/IEC JTC 1/SC 37 Biometrics, "Information technology – biometric performance testing and reporting – part 1: Principles and framework," *ISO/IEC*, pp. 19 795–1, 2006.

[70] A. K. Jain and J. Feng, "Latent palmprint matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 1032–1047, 2009.

[71] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, 2004.

[72] A. K. Jain, A. A. Ross, and K. Nandakumar, *Introduction to biometrics*. New York: Springer, 2011.

[73] A. J. Jeffreys, V. Wilson, and S. L. Thein, "Individual-specific 'fingerprints' of human dna," *Nature*, vol. 316, no. 6023, pp. 76–79, 1985.

[74] W. Jia, D.-S. Huang, and D. Zhang, "Palmprint verification based on robust line orientation code," *Pattern Recognition*, vol. 41, no. 5, pp. 1504–1513, 2008.

[75] W. Jia, Y.-H. Zhu, L.-F. Liu, and D.-S. Huang, "Fast palmprint retrieval using principal lines," in *IEEE International Conference on Systems, Man and Cybernetics*, 2009, pp. 4118–4123.

[76] W. Kang and Q. Wu, "Contactless palm vein recognition using a mutual foreground-based local binary pattern," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 11, pp. 1974–1985, 2014.

[77] W. Kang, Y. Liu, Q. Wu, and X. Yue, "Contact-free palm-vein recognition based on local invariant features," *PLOS ONE*, vol. 9, no. 5, pp. 1–12, 2014.

[78] Z. Khan, A. Mian, and Y. Hu, "Contour code: Robust and efficient multispectral palmprint encoding for human recognition," in *International Conference on Computer Vision*, 2011, pp. 1935–1942.

[79] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *IEEE/IFIP International Conference on Dependable Systems & Networks*, 2009, pp. 125–134.

[80] A. Kong, D. Zhang, and M. Kamel, "A survey of palmprint recognition," *Pattern Recognition*, vol. 42, no. 7, pp. 1408–1418, 2009.

[81] A.-K. Kong and D. Zhang, "Competitive coding scheme for palmprint verification," in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 1, 2004, pp. 520–523.

[82] Konica Minolta, https://www.konicaminolta.com/.

[83] K. Krishneswari and S. Arumugam, "Intramodal feature fusion based on pso for palmprint authentication," *International Journal on Image and Video Processing*, vol. 2, pp. 435–440, 2012.

[84] J. Kuckenberg, I. Tartachnyk, and G. Noga, "Temporal and spatial changes of chlorophyll fluorescence as a basis for early and precise detection of leaf rust and powdery mildew infections in wheat leaves," *Precision agriculture*, vol. 10, no. 1, pp. 34–44, 2009.

[85] A. Kumar and D. Zhang, "Personal authentication using multiple palmprint representation," *Pattern Recognition*, vol. 38, pp. 1695–1704, 2005.

[86] A. Kumar, D. C. Wong, H. C. Shen, and A. K. Jain, "Personal verification using palmprint and hand geometry biometric," in *Lecture Notes in Computer Science*, vol. 2688, 2003, pp. 668–678.

[87] A. Kumar, M. Hanmandlu, and H. Gupta, "Online biometric authentication using hand vein patterns," in *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, 2009, pp. 1–7.

[88] S.-Y. Kung, S.-H. Lin, and M. Fang, "A neural network approach to face/palm recognition," in *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, 1995, pp. 323–332.

[89] L. Lam, S.-W. Lee, C. Y. Suen *et al.*, "Thinning methodologies-a comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 869–885, 1992.

[90] V. Laxmi *et al.*, "Palmprint matching using lbp," in *International Conference on Computing Sciences*, 2012, pp. 110–115.

[91] M. K. Leung, A. C. M. Fong, and S. C. Hui, "Palmprint verification for controlling access to shared computing resources," *IEEE Pervasive Computing*, vol. 6, no. 4, pp. 40–47, 2007.

[92] Z. Li, R. Paul, T. B. Tis, A. C. Saville, J. C. Hansel, T. Yu, J. B. Ristaino, and Q. Wei, "Non-invasive plant disease diagnostics enabled by smartphone-based fingerprinting of leaf volatiles," *Nature plants*, vol. 5, no. 8, pp. 856–866, 2019.

[93] C.-L. Lin and K.-C. Fan, "Biometric verification using thermal images of palm-dorsa vein patterns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 2, pp. 199–213, 2004.

[94] J. A. Lucas, *Plant pathology and plant pathogens*. John Wiley & Sons, 2020.

[95] D. G. Luenberger, *Introduction to dynamic systems; theory, models, and applications*. John Wiley & Sons Chichester, 1979.

[96] W. Luo, "Efficient removal of impulse noise from digital images," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 2, pp. 523–527, 2006.

[97] X. Ma, X. Jing, H. Huang, Y. Cui, and J. Mu, "Palm vein recognition scheme based on an adaptive gabor filter," *IET Biometrics*, vol. 6, no. 5, pp. 325–333, 2017.

[98] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14, 1967, pp. 281–297.

[99] A.-K. Mahlein, "Plant disease detection by imaging sensors–parallels and specific demands for precision agriculture and plant phenotyping," *Plant disease*, vol. 100, no. 2, pp. 241–251, 2016.

[100] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.

[101] G. Marcus, "Deep learning: A critical appraisal," *arXiv preprint arXiv:1801.00631*, 2018.

[102] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.

[103] J. E. Mason, I. Traoré, and I. Woungang, *Machine Learning Techniques for Gait Biometric Recognition*. Springer, 2016.

[104] A. Matese and S. F. Di Gennaro, "Technology in precision viticulture: A state of the art review," *International journal of wine research*, vol. 7, pp. 69–81, 2015.

[105] G. K. O. Michael, T. Connie, and A. B. J. Teoh, "Touch-less palm print biometrics: novel design and implementation," *Image and Vision Computing*, vol. 26, no. 12, pp. 1551–1560, 2008.

[106] S. A. Miller, F. D. Beed, and C. L. Harmon, "Plant disease diagnostic capabilities and networks," *Annual Review of Phytopathology*, vol. 47, no. 1, pp. 15–38, 2009.

[107] M. Minervini, H. Scharr, and S. A. Tsaftaris, "Image analysis: the new bottleneck in plant phenotyping," *IEEE signal processing magazine*, vol. 32, no. 4, pp. 126–131, 2015.

[108] N. Miura, A. Nagasaka, and T. Miyatake, "Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification," *Machine Vision and Applications*, vol. 15, no. 4, pp. 194–203, 2004.

[109] K. Mochida, S. Koda, K. Inoue, T. Hirayama, S. Tanaka, R. Nishii, and F. Melgani, "Computer vision-based phenotyping for improvement of plant productivity: a machine learning perspective," *GigaScience*, vol. 8, no. 1, p. giy153, 2019.

[110] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in plant science*, vol. 7, p. 1419, 2016.

[111] T. Momal, L. Richeil, and H. Dankers, "Florida plant disease management guide: Grape (vitis spp.)," *University of Florida*, 2007.

[112] V. Mondello, A. Songy, E. Battiston, C. Pinto, C. Coppin, P. Trotel-Aziz, C. Clément, L. Mugnai, and F. Fontaine, "Grapevine trunk diseases: a review of fifteen years of trials for their control with chemicals and biocontrol agents," *Plant Disease*, vol. 102, no. 7, pp. 1189–1217, 2018.

[113] H. Ng, H.-L. Tong, W.-H. Tan, T. T.-V. Yap, P.-F. Chong, and J. Abdullah, "Human identification based on extracted gait features," *International Journal on New Computer Architectures and Their Applications*, vol. 1, no. 2, pp. 358–370, 2011.

[114] A. Nigam and P. Gupta, "Palmprint recognition using geometrical and statistical constraints," in *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS)*, 2014, pp. 1303–1315.

[115] F. Nutter Jr, P. Teng, and F. Shokes, "Disease assessment terms and concepts," *Plant disease*, 1991.

[116] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[117] D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "A dynamic algorithm for palmprint recognition," in *IEEE Conference on Communications and Network Security (CNS)*, 2015, pp. 659–662.

[118] D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "Biometric palmprint verification: A dynamical system approach," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 12, pp. 2676–2687, 2019.

[119] D. Palma, F. Blanchini, G. Giordano, and P. L. Montessoro, "A dynamic biometric authentication algorithm for near-infrared palm vascular patterns," *IEEE Access*, vol. 8, pp. 118 978–118 988, 2020.

[120] A. Pandey and K. K. Singh, "Analysis of noise models in digital image processing," *International Journal of Science, Technology & Management*, vol. 4, no. 01, pp. 2394–1537, 2015.

[121] F. Perez-Sanz, P. J. Navarro, and M. Egea-Cortines, "Plant phenomics: an overview of image acquisition technologies and image data analysis algorithms," *GigaScience*, vol. 6, no. 11, p. gix092, 2017.

[122] E. Piciucco, E. Maiorana, and P. Campisi, "Palm vein recognition using a high dynamic range approach," *IET Biometrics*, vol. 7, no. 5, pp. 439–446, 2018.

[123] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *arXiv preprint arXiv:2010.16061*, 2020.

[124] R. Raghavendra, B. Dorizzi, A. Rao, and G. K. Hemantha, "Pso versus adaboost for feature selection in multimodal biometrics," in *IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, 2009, pp. 1–7.

[125] T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, "Current and future applications of statistical machine learning algorithms for agricultural machine vision systems," *Computers and electronics in agriculture*, vol. 156, pp. 585–605, 2019.

[126] P. Revathi and M. Hemalatha, "Sms based hpccdd algorithm for the identification of leaf spot diseases," in *Proceedings of the Fourth International Conference on Signal and Image Processing 2012 (ICSIP)*. Springer, 2013, pp. 49–57.

[127] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[128] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos, "Biometric identification through hand geometry measurements," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 10, pp. 1168–1171, 2000.

[129] B. Sayed, I. Traoré, I. Woungang, and M. S. Obaidat, "Biometric authentication using mouse gesture dynamics," *IEEE Systems Journal*, vol. 7, no. 2, pp. 262–274, 2013.

[130] J. Schanda, "Cie colorimetry," *Colorimetry: Understanding the CIE system*, pp. 25–78, 2007.

[131] J. Serra and P. Soille, *Mathematical morphology and its applications to image processing*. Springer Science & Business Media, 2012, vol. 2.

[132] P. D. Sherman, *Colour vision in the nineteenth century: the Young-Helmholtz-Maxwell theory*. Adam Hilger, Bristol, 1981.

[133] H. L. Smith, *Monotone Dynamical Systems: An Introduction to the Theory of Competitive and Cooperative Systems*. American Mathematical Society, 2008, no. 41.

[134] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis, and machine vision*. Nelson Education, 2014.

[135] B. J. Staskawicz, F. M. Ausubel, B. J. Baker, J. G. Ellis, and J. Jones, "Molecular genetics of plant disease resistance," *Science*, vol. 268, no. 5211, pp. 661–667, 1995.

[136] J. Sun and W. Abdulla, "Palm vein recognition using curvelet transform," in *Proceedings of the 27th Conference on Image and Vision Computing*, 2012, pp. 435–439.

[137] Z. Sun, T. Tan, Y. Wang, and S. Z. Li, "Ordinal palmprint represention for personal identification," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2005, pp. 279–284.

[138] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[139] D. Tautz, "Hypervariability of simple sequences as a general source for polymorphic dna markers," *Nucleic acids research*, vol. 17, no. 16, pp. 6463–6471, 1989.

[140] M. L. Vijilious, S. Ganapathy, and V. S. Bharathi, "Palmprint feature extraction approach using nonsubsampled contourlet transform and orthogonal moments," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, 2012, pp. 735–739.

[141] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Computational intelligence and neuroscience*, vol. 2017, 2017.

[142] J. Wang and G. Wang, "Quality-specific hand vein recognition system," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2599–2610, 2017.

[143] L. Wang, G. Leedham, and S.-Y. Cho, "Infrared imaging of hand vein patterns for biometric purposes," *IET Computer Vision*, vol. 1, no. 3-4, pp. 113–122, 2007.

[144] E. Ward, S. J. Foster, B. A. FRAAIJE, and H. A. MCCARTNEY, "Plant pathogen diagnostics: immunological and nucleic acid-based approaches," *Annals of Applied Biology*, vol. 145, no. 1, pp. 1–16, 2004.

[145] J. L. Wayman, "Fundamentals of biometric authentication technologies," *International Journal of Image and Graphics*, vol. 1, no. 01, pp. 93–113, 2001.

[146] A. Wincy and G. C. Chandran, "Palmprint recognition using pcf and surf," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 10, pp. 996–1001, 2013.

[147] P. Wspanialy and M. Moussa, "A detection and severity estimation system for generic diseases of tomato greenhouse plants," *Computers and Electronics in Agriculture*, vol. 178, p. 105701, 2020.

[148] X. Wu, K. Wang, and D. Zhang, "Line feature extraction and matching in palmprint," in *Second International Conference on Image and Graphics*, vol. 4875, 2002, pp. 583–590.

[149] X. Wu, K. Wang, and D. Zhang, "Fuzzy directional element energy feature (fdeef) based palmprint identification," in *Proceedings of 16th International Conference on Pattern Recognition*, vol. 1, 2002, pp. 95–98.

[150] X. Wu, K. Wang, and D. Zhang, "Hmms based palmprint identification," in *Lecture Notes in Computer Science*, vol. 3072, 2004, pp. 775–781.

[151] G. Wübbeler, M. Stavridis, D. Kreiseler, R.-D. Bousseljot, and C. Elster, "Verification of humans using the electrocardiogram," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1172–1175, 2007.

[152] S. Xie, T. Yang, X. Wang, and Y. Lin, "Hyper-class augmented and regularized deep learning for fine-grained image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2645–2654.

[153] X. Yan, W. Kang, F. Deng, and Q. Wu, "Palm vein recognition based on multi-sampling and feature-level fusion," *Neurocomputing*, vol. 151, pp. 798–807, 2015.

[154] D. Zhang and W. Shu, "Two novel characteristics in palmprint verification: datum point invariance and line feature matching," *Pattern Recognition*, vol. 32, no. 4, pp. 691–702, 1999.

[155] D. Zhang, W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1041–1050, 2003.

[156] D. Zhang, Z. Guo, G. Lu, L. Zhang, Y. Liu, and W. Zuo, "Online joint palmprint and palmvein verification," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2621–2631, 2011.

[157] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[158] V. P. Zharov, S. Ferguson, J. F. Eidt, P. C. Howard, L. M. Fink, and M. Waner, "Infrared imaging of subcutaneous veins," *Lasers in Surgery and Medicine: The Official Journal of the American Society for Laser Medicine and Surgery*, vol. 34, no. 1, pp. 56–61, 2004.

[159] X. Zhou, K. Zhou, and L. Shen, "Rotation and translation invariant palmprint recognition with biologically inspired transform," *IEEE Access*, vol. 8, pp. 80 097–80 119, 2020.

[160] Y. Zhou and A. Kumar, "Human identification using palm-vein images," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, pp. 1259–1274, 2011.