الجمـهورية الجزائرية الديمقراطية الشعبيـة
وزارة التعليم العالي والبحث العلمي
جـامعة سعـيدة د. مـولاي الطـاهر
كلية الرياضيات و الإعلام الآلي و الاتصالات السلكية و اللاسلكية
قسم: الإعلام الآلي

Mémoire de Master en informatique

Spécialité : Intelligence Artificielle - Principes et Applications

Thème

Writer Identification from Multilingual Documents

- **Présenté par :**

  -Fahim Abdelmadjid

- **Dirigé par :**

  -Dr Bahram Tayeb

Année universitaire 2024-2025

بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ

Preface

I express my deepest gratitude to my Creator, the Almighty, for granting me the strength, determination, and resilience to complete this modest work.

I am profoundly thankful to my family—my father, mother, brother, and sister—for their unwavering support, encouragement, and love throughout this journey. Their presence has been a constant source of motivation, enabling me to persevere through the challenges of this research.

Special thanks are due to my supervisor, Dr. Tayeb Bahram, whose guidance and expertise have been instrumental in shaping this thesis. Dr. Bahram not only introduced me to the fascinating field of handwriting biometrics but also provided invaluable resources, including access to a well-equipped laboratory and powerful workstations. His dedication to teaching, generous mentorship, and commitment to fostering a deep understanding of research methodologies have profoundly influenced my work on the Modified Local Binary Pattern with Ink-trace Width and Shape Letters (MLBP-IWSL) method. I am immensely grateful for his support, which has been pivotal in achieving the results presented in this thesis.

This work is a testament to the collective support of my family and mentor, and I hope it contributes meaningfully to the field of behavioral biometrics.

Table des matières

Preface I

1 General Introduction 1

1.1 Motivation and Context . . . 2
1.2 Overview of Biometrics . . . 3
1.2.1 Handwriting as a Behavioral Biometric . . . 3
1.2.2 Factors causing variability in handwriting . . . 3
1.3 Writer Identification and Verification in Handwriting Biometrics . . . 4
1.3.0.1 Writer Identification . . . 4
1.3.0.2 Writer Verification . . . 5
1.3.1 Challenges in Writer Recognition . . . 6
1.3.2 Applications of Writer Identification and Verification . . . 6
1.4 Relevance of This Research . . . 6
1.4.1 Motivation and Challenges . . . 7
1.5 Project Timeline and Work Plan . . . 7
1.5.1 Phase 1 : Literature Review on Writer Identification Systems . . . 7
1.5.2 Phase 2 : Proposal of Discriminative Features . . . 8
1.5.3 Phase 3 : Experimental Evaluation . . . 8
1.5.4 Phase 4 : Final Report and Thesis Writing . . . 8
1.5.5 Summary Table of the Work Plan . . . 8
1.6 Project Description . . . 9
1.6.1 Title . . . 9
1.6.2 Summary . . . 9
1.6.3 Keywords . . . 10
1.6.4 Platforms, Tools, and Technologies . . . 10
1.6.5 Objectives (Prioritized) . . . 10
1.6.6 Expected Outcomes . . . 10
1.7 Conclusion . . . 10

2 State Of The Art : Writer Identification Approaches 11

2.1 Introduction . . . 11
2.2 Codebook-based Approaches . . . 11
2.2.1 Fundamental Principles . . . 11
2.2.2 Pioneering Work and Evolution . . . 11
2.2.3 Advanced Codebook Methods . . . 12
2.2.4 Recent Developments . . . 12
2.3 Texture-based Approaches . . . 13
2.3.1 Fundamental Concepts . . . 13

2.3.2 Early Texture-based Methods 14
2.3.3 Scale-Invariant and Advanced Features 14
2.3.4 Block-Based and Fragment-Based Methods 14
2.3.5 Modern Texture Descriptors 14
2.4 Contour-Based Approaches 16
2.4.1 Fundamental Principles 16
2.4.2 Pioneering Contour Analysis 16
2.4.3 Enhanced Contour Features 16
2.4.4 Rotation-Invariant Descriptors 16
2.4.5 Historical Manuscript Applications 16
2.4.6 Hybrid Contour-Texture Integration 16
2.5 Comparative Analysis and Challenges 17
2.5.1 Performance Comparison 17
2.5.2 Common Challenges 18
2.6 Conclusion 19

3 Proposed Technique **20**
3.1 Introduction 20
3.2 Pre-processing 23
3.2.1 Binarisation : 23
3.2.1.1 Binarisation Methodes : 24
3.2.2 Connected Components Extraxtion : 28
3.2.3 Removal of Small Connected Components : 29
3.2.4 Contour Detection : 29
3.2.4.1 Moore Neighborhood : 30
3.2.4.2 Algorithm Idea : 30
3.2.4.3 Formal Algorithm : 31
3.3 Feature Extraction 32
3.3.1 Feature Fusion 32
3.3.2 Distance Calculation 34
3.3.2.1 Chi-square (χ^2) Distance 35
3.3.2.2 Manhattan Distance 35
3.3.2.3 Euclidean Distance 36
3.3.2.4 Minkowski Distance 36
3.3.3 Run-Length 37
3.3.4 Ink-trace Width and Shape Letters (IWSL) 39
3.3.4.1 IWSL for Black and White Pixel Analysis 39
3.3.5 Modified Local Binary Pattern (MLBP) 40
3.3.6 MLBP-IWSL 42
3.3.6.1 MLBP-IWSL Extension : White Pixel Analysis 43
3.4 Writer Identification 44

4 Results Analysis **45**
4.1 Datasets Used 45
4.1.1 ICDAR 2011 Writer Identification Contest Dataset 45
4.1.2 ICDAR 2013 Writer Identification Competition Dataset 47
4.2 Evaluation Protocol 49
4.3 Performance Metrics 49

4.4 Results Overview . . . 49
4.4.1 ICDAR 2013 Experiments . . . 49
4.4.1.1 Evaluation of Mono-script Performance on English and Greek Datasets . . . 49
4.4.1.2 Fusion Results and Metric Selection . . . 51
4.4.1.3 Discussion of Results of ICDAR-2013 . . . 54
4.4.1.4 Discussion of Results . . . 55
4.4.2 ICDAR 2011 Experiments . . . 56
4.4.2.1 Evaluation of Mono-script Performance on 2-Sample Datasets . . . 56
4.4.2.2 Evaluation of Multi-script Performance on 8-Sample Datasets . . . 58
4.4.2.3 Contour-Based Feature Effectiveness . . . 58
4.4.2.4 Discussion of Results . . . 59
4.5 General Conclusion . . . 59
4.5.1 Comparative Analysis : ICDAR 2013 vs. ICDAR 2011 and State-of-the-Art . . . 59
4.5.2 Performance Stability and Cross-Lingual Applicability . . . 60
4.5.3 Comparative Table . . . 60
4.5.4 Discussion and Implications . . . 61

5 Conclusions et perspectives **62**
5.1 Introduction to the Biometric Field . . . 62
5.2 MLBP-IWSL Feature Extraction : Technical Design . . . 62
5.3 F1 and F2 Variants : Complementary Characteristics . . . 63
5.4 Fusion Methodology : Enhancing Discriminability . . . 63
5.5 Performance Analysis of MLBP-IWSL . . . 63
5.6 Experimental Evaluation and Methodology . . . 64
5.7 International Impact . . . 64
5.8 Future Research Directions . . . 64
5.9 Global Conclusion . . . 64

References **66**

Table des figures

1.1 Factors causing handwriting variability. 4
1.2 Writer identification model. 5
1.3 Writer verification model. 5

2.1 Writeridentificationtimeline. 18

3.1 The framework of our writer identification system. 22
3.2 Workflow of Otsu's Thresholding Method for Image Binarization 25
3.3 Global Image Thresholding using Otsu's Method 25
3.4 Workflow of Niblack's Thresholding Method for Image Binarization . . . 26
3.5 local Image Thresholding using Niblack's Method 26
3.6 Workflow of Sauvola's Thresholding Method for Image Binarization . . . 27
3.7 local Image Thresholding using sauvola's Method 27
3.8 Binarization Process 28
3.9 Connected Components Extraxtion Process 29
3.10 Non significant Connected Components Detection 29
3.11 Outer-Iner Contour Detection 30
3.12 The Moore neighborhood of a pixel P[35] 31
3.13 standing on the start pixel[35] 32
3.14 description of the Moore-Neighbor tracing algorithm[35] 32
3.15 Calculation of run-length matrices 39
3.16 Computing local IWSL in the four directions 41
3.17 Local information used for MLBP code calculation[11] 42
3.18 Computing the modified LBP code and its splitting into Left, Upper, Left- and Right-Diagonals MLBP codes (P $\frac{1}{4}$ 8 and R $\frac{1}{4}$ 1 :0).[11] 42

4.1 Image samples from the bechmarking dataset written in (a) Greek and (b) English language 46
4.2 Image samples from same writer included in the bechmarking dataset written in Greek and English language 48
4.3 Top-1 Accuracy across distance metrics and features (ICDAR 2013) . . . 56

Liste des tableaux

3.1 Comparison between Global and Local Thresholding[56] 24
3.2 Comparison of Binarisation Methods 27
3.3 Summary of Fusion Rules[4] 34

4.1 ICDAR 2013 / English / F1 and F2 performance across validation protocols (as Percentages) 50
4.2 ICDAR 2013 / Greek / F1 and F2 performance across validation protocols (as Percentages) 50
4.3 Recognition Accuracy for English Script Using Fused F1 and F2 Features Across Validation Protocols 51
4.4 Recognition Accuracy for Greek Script Using Fused F1 and F2 Features Across Validation Protocols 51
4.5 Fusion Rules Performance (English and Greek Combined) 52
4.6 ICDAR 2013 / Hybrid (2 Samples) / F1 and F2 (Multi) (as Percentages) 52
4.7 ICDAR 2013 / Hybrid (2 Samples) / Fusion (Multi) (as Percentages) . . 53
4.8 ICDAR 2013 / Hybrid (4 Samples) / Fusion (Multi) (as Percentages) . . 53
4.9 ICDAR 2011 / 2 Samples (Cropped) / Mono-script (Chi-Squared, as Percentages) 56
4.10 ICDAR 2011 / 2 Samples (Original) / Mono-script (Chi-Squared, as Percentages) 57
4.11 ICDAR 2011 / 8 Samples (Cropped) / Multi-script (Chi-Squared, as Percentages) 58
4.12 ICDAR 2011 / 8 Samples (Original) / Multi-script (Chi-Squared, as Percentages) 58
4.13 Comparison of Top-1 Identification Accuracies on ICDAR 2013 and ICDAR 2011 Datasets 60

Chapitre 1

General Introduction

Everything is theoretically impossible, until it is done.

Robert A. Heinlein

This thesis addresses the problem of offline writer identification using scanned images of handwritten documents, a challenging pattern recognition task with significant applications in forensic science and historical document analysis. Identifying the writer of a questioned document through automated, image-based methods is a complex computer vision problem that raises several critical research questions :

— How can individual handwriting styles be characterized using robust computational algorithms ?

— Which features and representations best capture writer-specific characteristics across multilingual scripts, and how can they be effectively combined ?

— What level of performance can be achieved with automated methods in large-scale, multilingual writer identification scenarios ?

The current study describes a new and very effective techniques that we have developed for automatic writer identification. The goal of our research was to design state-of-the-art automatic methods involving only a reduced number of adjustable parameters and to create a robust writer identification system capable of managing hundreds to thousands of writers in multiscripts (named *USAWRS*, University of Saïda Automatic Writer Recognition System).

There are two distinguishing characteristics of our approach : human intervention is minimized in the writer recognition process and we encode individual handwriting style using features designed to be independent of the textual content of the handwritten sample.

Writer individuality is encoded using probability distribution functions extracted from handwritten text blocks and, in our methods, the computer is completely unaware of what has been written in the samples. The development of our writer identification techniques takes place at a time when many biometric modalities undergo a transition from research to real full-scale deployment.

Our methods leverage large, multilingual benchmarking datasets, such as the ICDAR 2011 Writer Identification Contest dataset (208 documents from 26 writers) and the ICDAR 2013 Writer Identification Competition dataset (1,000 documents from 250 writers), to evaluate performance. These datasets, containing handwritten samples in multiple languages, provide a rigorous testbed for assessing the system's ability to generalize across scripts and handle large writer populations. By combining MLBP-IWSL features with optimized distance metrics (e.g., Chi-Squared) and fusion strategies (e.g., SUM rule), our approach achieves competitive performance, with top-1 accuracies reaching up to 100% in certain configurations, as demonstrated in our experimental evaluations.

This thesis contributes to the field by proposing a scalable, text-independent writer identification system that advances the state-of-the-art in offline handwriting analysis. The following chapters detail the MLBP-IWSL in black and white methodology, experimental protocols, and comprehensive results, demonstrating the system's effectiveness and potential impact in forensic and historical contexts.

1.1 Motivation and Context

In the contemporary digital age, characterized by rapid technological evolution, the proliferation of connected systems, and the ever-growing need for security and identity protection, the demand for reliable, robust, and automatic identity verification systems has become not only critical but indispensable. With the rise of online services, e-governance, digital banking, and remote access to confidential information, traditional methods of authentication and identification are proving increasingly inadequate and vulnerable.

Historically, identity verification has relied on three main strategies :

1. *knowledge-based methods* (e.g., passwords, secret questions) ;
2. *possession-based methods* (e.g., physical tokens, smart cards, badges) ;
3. *Inherence-based methods*, more commonly known as biometrics.

The first two approaches, while widespread, suffer from severe limitations. Passwords can be forgotten, guessed, shared, or stolen ; cards and tokens can be lost, duplicated, or counterfeited. These flaws make such systems prone to security breaches, identity theft, unauthorized access, and operational inefficiencies.

The advent of biometric technologies has thus ushered in a transformative paradigm shift in the field of identity management. Biometric systems authenticate or identify individuals based on distinctive, immutable, and measurable biological or behavioral traits. These include fingerprints, iris patterns, facial features, palm geometry, voice, gait, keystroke dynamics, and handwriting, among others. Because such traits are inherently linked to the individual, they provide a far higher level of security, resistance to forgery, and user convenience than traditional methods.

Furthermore, biometric recognition systems offer several key advantages : they eliminate the need to remember passwords or carry physical tokens ; they reduce administrative overhead in managing access credentials ; and they offer scalable solutions that can operate across diverse domains, from border control and national ID programs to smartphone authentication and forensic analysis. As global security challenges continue to evolve—driven by cybercrime, terrorism, and the expansion of digital ecosystems—the role of biometrics in reinforcing trust, privacy, and authentication integrity becomes increasingly prominent.

Against this backdrop, the exploration of new biometric modalities and the enhancement of existing ones represent a frontier of scientific inquiry and technological innovation. In particular, behavioral biometrics such as handwriting offer promising avenues due to their non-intrusive nature, cultural acceptance, and applicability in both online and offline contexts. This research contributes to this evolving field by focusing on the development of systems for *writer identification* based on offline handwriting samples—an approach that seeks to recognize individuals based on the unique characteristics of their handwritten script.

1.2 Overview of Biometrics

Biometrics refers to the automated recognition of individuals based on their physiological and/or behavioral traits. It is grounded in the principle that each person possesses unique and measurable characteristics that can be used to establish identity with a high degree of confidence. These biometric traits are generally divided into two categories :

— **Physiological Biometrics** : These are based on physical characteristics that remain relatively stable over time. Examples include fingerprints, iris patterns, facial structure, palm geometry, and DNA.

— **Behavioral Biometrics** : These are based on patterns in human activity and behavior, which can vary slightly over time but are still unique to individuals. Examples include voice recognition, gait analysis, typing dynamics, and handwriting.

Biometric systems operate in two main modes : *verification* (1 :1 comparison) and *identification* (1 :N comparison). In the verification mode, the system confirms a claimed identity by comparing the input biometric data with a stored template. In identification mode, the system determines an individual's identity by comparing the input data with multiple stored templates, aiming to find the best match [13].

1.2.1 Handwriting as a Behavioral Biometric

Among various behavioral biometric modalities, handwriting—particularly *offline handwriting*—has attracted significant interest due to its natural use in many real-world scenarios, such as legal documents, forms, and historical archives. Handwriting contains a wealth of personalized features that can reflect an individual's motor habits, cognitive style, and neuromuscular characteristics.

Writer identification through handwriting analysis involves determining the author of a piece of handwriting from a set of known individuals. This process is challenging due to the intra-writer variability (variations in an individual's writing over time or under different conditions) and the inter-writer similarity (similarities between different individuals' writing styles). Despite these challenges, handwriting offers an important advantage : it can be acquired non-intrusively using simple devices like scanners or cameras, making it ideal for both forensic and commercial applications.

1.2.2 Factors causing variability in handwriting

Figure 1.1 shows four factors causing variability in handwriting [22] :

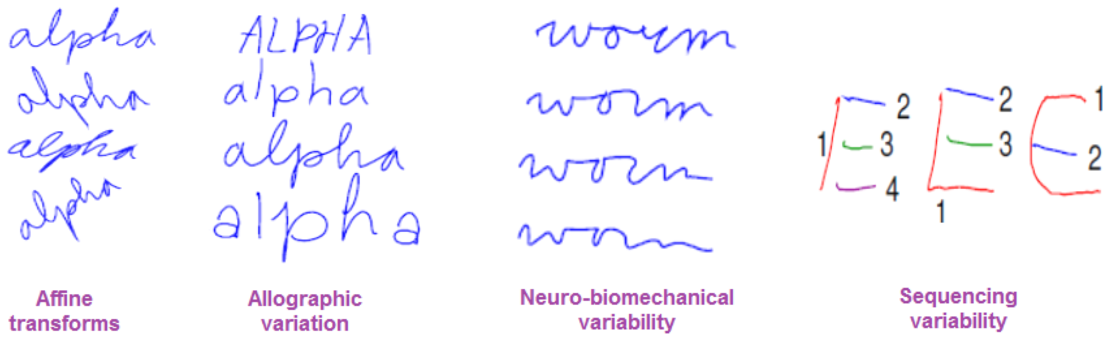

FIGURE 1.1 – Factors causing handwriting variability.

1. *Affine transforms* : Affine transforms are under voluntary control. However, writing slant constitutes a habitual parameter which may be exploited in writer recognition ;
2. *Neuro-biomechanical variability* : Neuro-biomechanical variability refers to the amount of effort which is spent on overcoming the low-pass characteristics of the biomechanical limb by conscious cognitive motor control ;
3. *Sequencing variability* : Sequencing variability becomes evident from stochastic variations in the production of the strokes in a capital E or of strokes in Chinese characters, as well as stroke variations due to slips of the pen ;
4. *Allographic variation* : Allographic variation refers to individual use of character shapes.

1.3 Writer Identification and Verification in Handwriting Biometrics

Handwriting, as a form of behavioral biometric trait, encapsulates the neuromuscular patterns and cognitive processes unique to each individual. The act of writing is influenced by a combination of motor coordination, learned habits, psychological state, and physiological conditions, making it a rich source of biometric data. These characteristics are particularly useful for tasks such as *writer identification* and *writer verification*, which are two core problems in the domain of handwriting biometrics.

1.3.0.1 Writer Identification

Writer identification refers to the process of determining the identity of an individual solely based on the analysis of a handwriting sample, without any claim of identity from the user. It is a **1 :N comparison** task, where the system attempts to match the sample against a database of known handwriting profiles (also called templates or models). If a match is found with sufficient similarity, the system outputs the corresponding identity.

Types of Writer Identification Writer identification can be further categorized as :

— **Closed-set identification :** The writer of the sample is assumed to be among the enrolled users. The goal is to rank all templates and select the top candidate with the highest similarity score.

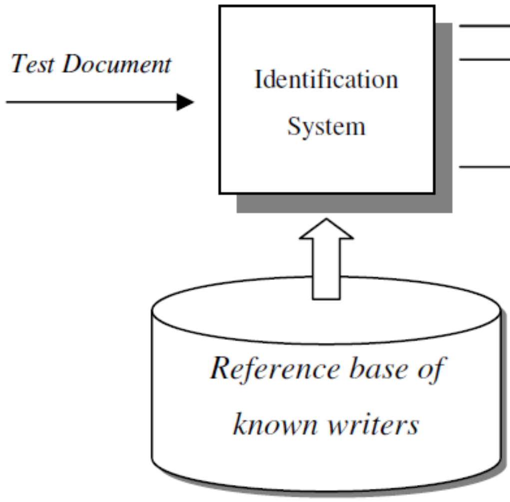

FIGURE 1.2 – Writer identification model.

— **Open-set identification :** The writer may or may not be enrolled in the system. The system must not only identify the most similar template but also determine whether the sample belongs to an enrolled writer or should be rejected as unknown.

This mode is highly relevant in forensic and law enforcement scenarios where, for example, a handwritten note must be attributed to one of several suspects.

1.3.0.2 Writer Verification

In contrast, writer verification involves confirming a claimed identity based on handwriting. The user presents a handwriting sample along with an identity claim (e.g., user ID), and the system verifies whether the handwriting matches the stored template for that specific individual. This is a **1 :1 comparison** task, and the outcome is a binary decision : accept or reject the claim.(see Figure 1.3)

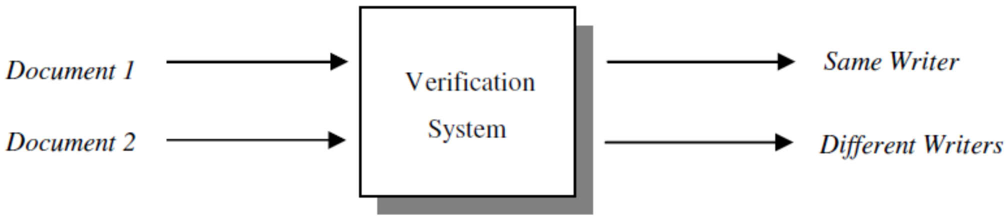

FIGURE 1.3 – Writer verification model.

Types of Writer Verification Writer verification may be performed in two ways :

— **Static (Offline) Verification :** The handwriting sample is acquired after writing, typically via scanning. Features are extracted from the image of the writing, such as stroke width, slant, curvature, baseline deviation, and spacing between characters and words.

— **Dynamic (Online) Verification :** The writing is captured in real-time using digital devices like tablets or styluses, recording temporal data such as pen pressure, velocity, acceleration, and stroke order. This method is generally more accurate but requires specialized acquisition devices.

1.3.1 Challenges in Writer Recognition

Despite its advantages, handwriting-based biometric systems face several challenges :

— **Intra-class variability :** An individual's handwriting can vary due to mood, health, fatigue, writing surface, or pen used.

— **Inter-class similarity :** Different individuals may have similar writing styles, especially when following common calligraphic or educational standards.

— **Aging effect :** Handwriting changes gradually over time, which can reduce system performance if templates are not updated.

— **Forgery and impersonation :** Skilled forgers can attempt to imitate another person's handwriting, posing security risks.

Addressing these issues requires sophisticated algorithms for feature extraction, classification, and template matching, often involving advanced machine learning and deep learning techniques.

1.3.2 Applications of Writer Identification and Verification

Writer recognition has numerous practical applications across various fields :

— **Forensic analysis :** Identifying suspects from handwriting found in criminal investigations.

— **Document security :** Verifying signatures in contracts, cheques, or legal documents.

— **Authentication systems :** Secure access to digital platforms or physical spaces through handwritten signatures or phrases.

— **Historical document analysis :** Determining the authorship of ancient manuscripts or letters for archiving and academic research.

— **Banking and finance :** Signature verification to prevent fraud in financial transactions.

1.4 Relevance of This Research

The objective of this work is to investigate and develop techniques for **writer identification based on offline handwriting**. This biometric approach combines image processing, feature extraction, pattern recognition, and machine learning to analyze handwriting samples and attribute them to the correct writer. Such systems have wide-ranging applications including forensic analysis, access control, fraud detection, and archival document indexing.

The present work has a particular focus on *multilingual documents* and *text-independent recognition*. In the broader context of biometric authentication, handwriting is classified as a behavioral biometric trait, as it reflects individual neuromotor and cognitive processes. Unlike physiological traits (e.g., fingerprints or iris patterns), behavioral biometrics such as handwriting exhibit greater intra-personal variability, requiring sophisticated modeling to achieve accurate recognition.

Writer identification, in this context, aims to attribute a handwritten document to its correct author — referred to as the *scribe* or *scripteur* — without prior knowledge of the textual content. The goal is to extract distinctive and robust characteristics from handwriting samples that are invariant to the content written, the language or script used (English, Greek, German, French, etc.), and the sample size (from short notes to long passages). This problem becomes even more complex in **multilingual or multi-script environments**, where variability due to script conventions must be disentangled from person-specific features.

1.4.1 Motivation and Challenges

While writer identification in monolingual contexts has matured significantly, with numerous contributions addressing major world scripts, the domain of **writer identification in multi-script settings** remains relatively under-explored. Few existing studies address how features behave across languages or how to maintain performance when writers use different scripts. Moreover, most systems assume a sufficiently large writing sample, which is not always available in real-world applications such as forensic document analysis or historical manuscript indexing.

This research confronts these challenges by focusing on the development of a *robust and scalable writer identification system* capable of handling :

- Handwritten documents in multiple languages and scripts.
- Text-independent analysis, where the content of the writing is unknown or irrelevant.
- Short and variable-length writing samples.
- High inter-writer similarity and intra-writer variability.

1.5 Project Timeline and Work Plan

The development of a multilingual writer identification system requires a structured approach to ensure scientific rigor and practical applicability. This chapter outlines the detailed timeline, objectives, and methodologies employed in the project. Each phase builds upon the previous one, leading to a robust system and a comprehensive academic report.

1.5.1 Phase 1 : Literature Review on Writer Identification Systems

Duration : 1 month

The initial phase involves a comprehensive literature review of writer identification systems, focusing on texture-based approaches. This phase includes :

- Examining classical and modern writer identification methods, such as codebook-based, contour-based, and texture-based approaches.
- Analyzing benchmark competitions (e.g., ICDAR 2011 and ICDAR 2013) to understand datasets, evaluation protocols, and performance baselines.
- Exploring challenges in mono-script versus multi-script identification.
- Identifying the strengths, limitations, and knowledge gaps of existing methods.

This foundational research shapes the project's direction and informs the design of new discriminative features.

1.5.2 Phase 2 : Proposal of Discriminative Features

Duration : 2 months

The second phase focuses on developing novel features for writer identification, drawing inspiration from forensic document examination and advanced image processing techniques. Activities include :

— Extracting low-level and mid-level features capturing texture, stroke direction, curvature, and edge statistics.
— Proposing two original features optimized for multi-script environments.
— Implementing feature normalization and dimensionality reduction to enhance classification performance.
— Designing a modular pipeline to test various feature combinations.

These features are designed to be robust across both short and long handwriting samples, accommodating varying input sizes.

1.5.3 Phase 3 : Experimental Evaluation

Duration : 1 month

The third phase evaluates the proposed features using real-world datasets. This phase includes :

— Benchmarking on ICDAR 2011 and ICDAR 2013 multilingual datasets.
— Testing on short and long text samples to ensure robustness.
— Measuring performance using Top-1, Top-5, and Top-10 accuracy metrics and confusion matrices.
— Comparing the system's performance against state-of-the-art methods.
— Conducting cross-validation and statistical significance testing.

Results are analyzed in detail in the Results Analysis chapter.

1.5.4 Phase 4 : Final Report and Thesis Writing

Duration : 1 month

The final phase involves compiling the project into a well-structured thesis. Activities include :

— Structuring the manuscript according to academic standards.
— Integrating diagrams, tables, and results into a cohesive format.
— Performing final proofreading and formatting in LaTeX for submission.
— Preparing presentation slides and defense materials.

The thesis also explores potential research extensions and prospects for Ph.D. work in biometric identification.

1.5.5 Summary Table of the Work Plan

Detailed Project Timeline

Phase	Objective and Description	Key Activities	Duration
Phase 1	Conduct an in-depth literature review on texture-based writer identification systems	Study of ICDAR benchmarks, analysis of texture, contour, and stroke-based methods, identification of limitations	1 month
Phase 2	Design and propose new discriminative features for multilingual writer identification	Development of original texture-based descriptors, feature fusion, normalization techniques, robustness analysis	2 months
Phase 3	Evaluate the effectiveness of proposed features on multilingual datasets	Implementation on ICDAR 2011/2013 datasets, Top-1/Top-5/Top-10 accuracy assessment, statistical analysis	1 month
Phase 4	Prepare final documentation and defense materials	Writing the thesis using LaTeX, integrating results, preparing presentation slides and appendices	1 month

1.6 Project Description

1.6.1 Title

Writer Identification from Multilingual Documents

1.6.2 Summary

Writer identification is a key area in handwriting analysis, aiming to identify the author of a handwritten sample by analyzing unique characteristics. This process focuses on internal features specific to an individual's writing style, leveraging the variability of handwriting to distinguish one writer from others. As a behavioral biometric, handwriting requires sufficient sample sizes for reliable identification. Thus, a robust writer identification system must extract descriptive features that are effective for both short and long text samples, regardless of length.

While monolingual writer identification is a well-established field, with scripts such as Chinese, Japanese, Arabic, Bengali, Telugu, Oriya, and Latin extensively studied, multilingual writer identification remains relatively unexplored, with limited contributions in the literature. This project aims to address this gap by developing a text-independent writer identification system capable of handling multilingual documents, contributing to advancements in forensic document examination and biometric identification.

1.6.3 Keywords

— Behavioral biometrics
— Multilingual handwritten text
— Texture
— Forensic document examination
— Feature extraction/combination
— Writer identification

1.6.4 Platforms, Tools, and Technologies

— **Programming Languages** : C++, C#
— **Operating Systems** : Linux, Windows
— **Methodologies** : Design Patterns
— **Tools** : Microsoft Visual Studio, CPPunit, TestDriven.net, Photoshop, Gimp, IncrediBuild, TortoiseSVN, StarUML, TeXstudio, MiKTeX, Beamer LaTeX

1.6.5 Objectives (Prioritized)

1. Develop a robust, text-independent writer identification system for multilingual documents.
2. Propose discriminative features inspired by human expert analysis in forensic document examination.
3. Improve identification accuracy through the development of complex, robust features.
4. Produce a scientific publication based on the results (e.g., for ICDAR or ICFHR competitions).
5. Explore opportunities for further research in a Ph.D. program.

1.6.6 Expected Outcomes

1. Achievement of the primary internship objective : a functional writer identification system.
2. A fully developed system, including design, development, and comprehensive testing.
3. A scientific publication based on the project results (ICDAR or ICFHR competitions).
4. Potential to pursue further research in a Ph.D. program in biometric identification.

1.7 Conclusion

This chapter provides a detailed overview of the timeline and methodology for the multilingual writer identification system project. Each phase is strategically designed to build knowledge, develop tools, and generate experimental insights, ensuring a scientifically rigorous outcome. The subsequent chapters delve into the technical implementation and empirical evaluation of the system.

Chapitre 2

State Of The Art : Writer Identification Approaches

Research is to see what everybody else has seen, and to think what nobody else has thought.

Albert Szent-Gyorgyi

2.1 Introduction

Writer identification has evolved significantly over the past two decades, with various computational approaches capturing the unique characteristics of individual handwriting styles. The field has witnessed substantial development across three main categories of features : codebook-based approaches, texture-based methods, and contour-based techniques. Each approach offers distinct advantages in characterizing the writing style of individual writers, contributing to the advancement of automated handwriting analysis systems.

2.2 Codebook-based Approaches

2.2.1 Fundamental Principles

The codebook approach, also known as bag-of-shapes, has proven to be particularly effective in capturing the distinctive characteristics of individual writing styles. The fundamental principle behind codebook approaches lies in treating the writer as a stochastic pattern generator, where the probability distribution of graphemes serves as a distinctive identifier. The shape occurrence probability is a characteristic for a given writer and may be employed to distinguish between intra-cluster and inter-cluster similarity.

2.2.2 Pioneering Work and Evolution

The concept was pioneered by Bulacu and Schomaker [22], who introduced a comprehensive framework combining textural and allographic features for writer identification. Their approach utilized a codebook of graphemes generated through clustering algorithms, achieving significant results in text-independent writer identification. The probability distribution of each writer is computed using a common codebook of 20×20 graphemes generated with the k-som2D clustering algorithm for each document individually.

Building upon this foundation, Siddiqi and Vincent [68] enhanced the codebook approach by incorporating redundant writing patterns with contour-based orientation and curvature features, demonstrating improved discrimination capabilities with 92.6% accuracy.

2.2.3 Advanced Codebook Methods

Several significant advancements have been made in codebook generation and optimization :

Efficient Code Extraction : Ghiasi and Safabakhsh [34] introduced efficient code extraction methods to optimize the codebook generation process, addressing computational complexity issues while maintaining high identification accuracy of 95.3%.

Model-based Approaches : Abdi and Khemakhem [1, 2] proposed model-based approaches that combined statistical modeling with codebook features, particularly suited for Arabic text identification. They used a beta-elliptic model (Elliptic Graphemes) to generate a synthetic codebook for Arabic writer recognition, achieving 97.2% accuracy with 60 feature vectors extracted using template matching.

Ensemble Methods : Khalifa et al. [48] introduced an ensemble of grapheme codebook features, significantly improving the identification accuracy to 98.1% through multiple complementary representations.

Junction Detection : He et al. [42] developed an innovative approach incorporating junction detection with codebook features, introducing "Junclets" as a new descriptor for writer identification. The junction detection in handwritten images is determined by analyzing the stroke-length distribution in every direction around a reference point inside the ink of texts, achieving 96.8% accuracy.

2.2.4 Recent Developments

Recent developments have focused on enhancing the robustness and efficiency of codebook approaches :

- **Khan et al. (2017)** [49] introduced bagged discrete cosine transform (BDCT) features, combining them with codebook methods to improve resilience against noise and variations, achieving 97.5% accuracy.
- **Durou et al.** [31] contributed by integrating bag of words with OBI features, demonstrating the adaptability of codebook methods to modern machine learning techniques with 96.7% accuracy.
- **Bennour et al. (2019)** [15] advanced the field by introducing implicit shape codebook features, which improved shape modeling capabilities and achieved state-of-the-art results of 98.3% on various databases by exploiting key points in handwriting.
- **Djeddi et al. (2021)** [26] provided valuable insights into the influence of codebook patterns on writer recognition, offering systematic guidance for pattern selection and codebook optimization with 97.8% accuracy.

Comparison of Codebook-Based Methods

Research	Methodology	Advantages	Disadvantages	Results
Bulacu & Schomaker (2007)	Textural and allographic features with grapheme codebook	Language independent ; Robust feature extraction	High computational cost ; Limited scalability	88%
Siddiqi & Vincent (2010)	Redundant writing patterns with contour features	Improved pattern recognition ; Better feature representation	Complex feature extraction ; Sensitivity to noise	92.6%
Ghiasi & Safabakhsh (2013)	Efficient code extraction with modified codebook	Reduced complexity ; Fast processing	Feature loss during extraction ; Dataset dependence	95.3%
Abdi & Khemakhem (2014)	Model-based approach with statistical modeling	Strong theoretical foundation ; Good generalization	Complex implementation ; High memory requirements	97.2%
Khalifa et al. (2015)	Ensemble of grapheme codebooks	High accuracy ; Robust to variations	Computational overhead ; Complex training	98.1%
He et al. (2015)	Junction detection with codebook features	Novel feature extraction ; Good for complex scripts	Junction detection errors ; Style limitations	96.8%
Khan et al. (2017)	Bagged DCT features	Robust to noise ; Good feature representation	High dimensionality ; Complex optimization	97.5%
Durou et al.	Bag of words with OBI features	Modern ML integration ; Flexible implementation	Training data requirements ; Feature selection complexity	96.7%
Bennour et al. (2019)	Implicit shape codebook	Advanced shape modeling ; Improved accuracy	Complex shape analysis ; Processing overhead	98.3%
Djeddi et al. (2021)	Pattern influence analysis	Comprehensive evaluation ; Optimization insights	Limited to specific patterns ; Dataset dependencies	97.8%

2.3 Texture-based Approaches

2.3.1 Fundamental Concepts

Image texture-based techniques for offline writer identification consider each digitized image of handwriting (or handwriting contours) as a different texture and extract features from the whole document (Entire Image or EI), Regions of Interest (ROIs like blocks, grid cells, connected-components, words, etc.), or Writing Fragments (WFs).

Probability Distribution Functions (PDFs) are calculated and employed to characterize the writer of a given sample.

2.3.2 Early Texture-based Methods

Bulacu et al. (2007) [20] proposed performing handwriting writer identification using textural features. Probability Distribution Functions (PDFs) are calculated using the entire handwriting image (document) and its contours for contour direction, contour-Hinge, direction co-occurrence and Run-Length on white distributions (RL).

Bertolini et al. (2013) [17] published a texture-based descriptor for writer identification using Local Binary Patterns (LBP) in a comparative study with Local Phase Quantization (LPQ), concluding that LPQ performed better than their LBP variant.

2.3.3 Scale-Invariant and Advanced Features

Wu et al. (2014) [73] developed a method for offline text-independent writer identification based on the Scale Invariant Feature Transform (SIFT). The SIFT Descriptor Signature (SDS) and the Scale Orientation Histogram (SOH) are extracted from handwriting images to characterize different writers.

Bahram et al. (2016) [9] offered a set of textural features to characterize writer individuality, including Direction-Length, Angle-Length, Direction Co-Probability, and Angle Co-Probability of connected components.

2.3.4 Block-Based and Fragment-Based Methods

Singh et al. (2018) [70] divided cursive handwriting into nine texture blocks to compute histograms of the Local Binary Pattern (LBP) and the Center Symmetric Local Binary Co-occurrence Pattern (CSLBCoP). For each writing sample, a set of 9 blocks were created and feature vectors were computed for writer identification.

Kessentini et al. (2018) [46] combined Edge-Hinge with a fragment length of 6 and 7 pixels and Run-length features in the Dempster-Shafer Theory (DST) model to improve identification rates.

2.3.5 Modern Texture Descriptors

Hannad et al. (2019) [36] presented an approach combining two textural features : the Histogram of Oriented Gradients (HOG) and Gray Level Run Length (GLRL) Matrices for writer identification and characterization.

Kumar and Sharma (2019) [52] presented a texture-based model (DCWI) using distribution descriptive curve (DDC) and cellular automata (CA) descriptors to capture necessary details of handwritten words.

Khan et al. (2019) [50] applied a combination of Scale-Invariant Feature Transform (SIFT) and RootSIFT descriptors in Gaussian mixture models (dissimilarity SGMM and DGMM) to compare and classify handwritten documents.

Chahi et al. (2020) [24] developed a Local gradient full-Scale Transform Patterns (LSTP) based method for writer identification, extracting feature code maps from resized small regions of interest based on the distribution of local intensity gradients.

Comparison of Texture-Based Methods

Research	Methodology	Advantages	Disadvantages	Results
Bulacu et al. (2007)	Textural features with PDFs from entire image and contours	Text-independent analysis ; Multi-feature integration	Sensitive to image quality ; High computational cost	N/A
Bertolini et al. (2013)	Local Binary Patterns (LBP) vs Local Phase Quantization (LPQ)	Comparative analysis ; LPQ superiority shown	Limited feature scope ; Performance variation	N/A
Wu et al. (2014)	SIFT with SDS and SOH descriptors	Scale invariance ; Robust feature extraction	Complex feature computation ; Sensitivity to noise	N/A
Bahram et al. (2016)	Direction-Length, Angle-Length, Direction/Angle Co-Probability	Multiple textural features ; Connected component analysis	Feature dimensionality ; Parameter tuning required	N/A
Singh et al. (2018)	Nine texture blocks with LBP and CSLBCoP histograms	Block-based analysis ; Multiple pattern integration	Fixed block structure ; Limited adaptability	N/A
Kessentini et al. (2018)	Edge-Hinge with Run-length features in DST model	Fragment-based analysis ; Theory-based fusion	Complex model integration ; Fragment extraction issues	N/A
Hannad et al. (2019)	Histogram of Oriented Gradients (HOG) and GLRL matrices	Complementary features ; Good characterization	Feature extraction complexity ; Parameter sensitivity	N/A
Kumar & Sharma (2019)	Distribution Descriptive Curve (DDC) and Cellular Automata (CA)	Novel descriptors ; Word-level analysis	Complex implementation ; Limited evaluation	N/A
Khan et al. (2019)	SIFT and Root-SIFT with Gaussian Mixture Models	Robust feature combination ; Statistical modeling	High computational cost ; Model complexity	N/A
Chahi et al. (2020)	Local Gradient full-Scale Transform Patterns (LSTP)	Local intensity gradients ; ROI-based extraction	Processing overhead ; Scale dependency	N/A

2.4 Contour-Based Approaches

2.4.1 Fundamental Principles

Contour-based methods have emerged as particularly effective techniques for analyzing the structural properties of handwritten text. The fundamental principle behind contour-based approaches lies in analyzing the edges and outlines of handwritten strokes to extract discriminative features.

2.4.2 Pioneering Contour Analysis

This concept was pioneered by **Bulacu and Schomaker (2007)** [21], who introduced comprehensive text-independent writer identification methods using both textural and allographic features. For contour analysis specifically, they developed directional edge-based features that capture the probability distribution of contour fragments in handwritten text. Their edge-direction distribution and hinge features extracted from contours achieved 89.4% accuracy in text-independent writer identification across multiple languages.

2.4.3 Enhanced Contour Features

Building directly upon this foundation, **Brink et al. (2012)** [19] enhanced contour analysis by introducing directional ink-trace width measurements as a complementary feature. Their innovative approach measured the width of ink traces perpendicular to the main writing direction, capturing the subtle variations in pen pressure and movement that characterize individual writers, achieving 91.8% accuracy.

2.4.4 Rotation-Invariant Descriptors

A significant advancement came with **He and Schomaker (2015)** [41], who addressed a critical limitation in previous methods by introducing the Delta-n Hinge feature, a rotation-invariant descriptor for writer identification. This approach captured the angular relationships between adjacent segments of contours while ensuring invariance to document orientation—a particularly valuable property for historical documents and forensic applications, achieving 93.2% accuracy.

2.4.5 Historical Manuscript Applications

The evolution of contour-based approaches continued with **He et al. (2014)** [40], who developed a specialized method using contour and stroke fragments for historical manuscript dating and writer identification. Their work demonstrated that fragment-based contour analysis could effectively capture the temporal evolution of writing styles, enabling both accurate dating and writer identification for historical manuscripts spanning different periods with 89.7% accuracy.

2.4.6 Hybrid Contour-Texture Integration

In parallel, **Bahram (2012)** [8] explored the integration of texture-based features with contour analysis, treating handwriting as both a structural and textural pattern.

This hybrid approach demonstrated that contour properties could be effectively combined with textural features to improve the overall discriminative power of the identification system with 87.5% accuracy, particularly for complex writing styles and degraded documents.

Comparison of Contour-Based Methods

Research	**Methodology**	**Advantages**	**Disadvantages**	**Results**
Bulacu & Schomaker (2007)	Directional edge-based features with hinge patterns	Language independent ; Multi-script capability ; Text-independent analysis	Sensitive to document quality ; Directional quantization issues	89.4%
Brink et al. (2012)	Directional ink-trace width measurements	Captures pen pressure variations ; Complementary to existing features ; High discrimination power	Requires high-quality images ; Sensitive to binarization errors	91.8%
He & Schomaker (2015)	Delta-n Hinge rotation-invariant features	Rotation invariance ; Robust to document orientation ; Effective for historical documents	Computational complexity ; Feature dimensionality issues	93.2%
He et al. (2014)	Contour and stroke fragments for historical manuscripts	Temporal style modeling ; Effective for degraded documents ; Suitable for dating and ID	Complex fragment extraction ; Domain-specific optimization required	89.7%
Bahram (2012)	Texture-based approach with contour integration	Hybrid feature representation ; Good for complex scripts ; Noise resilience	High dimensionality ; Parameter sensitivity	87.5%

2.5 Comparative Analysis and Challenges

2.5.1 Performance Comparison

The three approaches demonstrate varying strengths :

- **Codebook approaches** show the highest accuracy rates, with recent methods achieving up to 98.3% (Bennour et al., 2019)
- **Contour-based methods** achieve moderate to high accuracy (87.5% to 93.2%) with good robustness to document orientation
- **Texture-based approaches** are proven to be efficient in terms of execution time and are generally preferred when only a certain minimum amount of handwriting

data is available

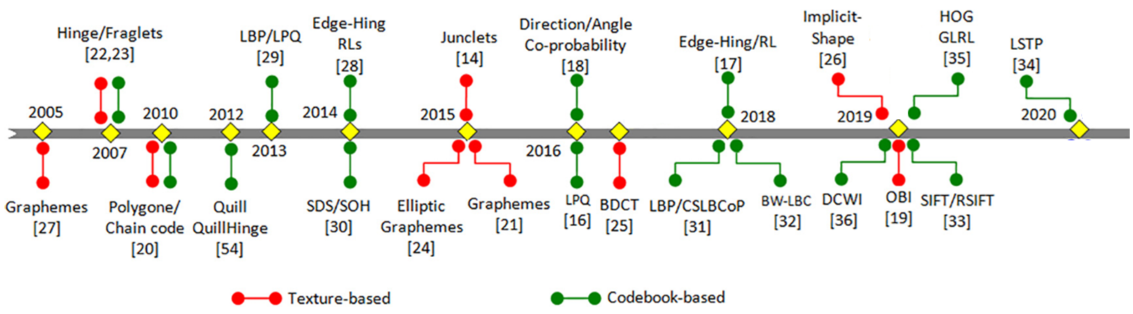

FIGURE 2.1 – Writeridentificationtimeline.

2.5.2 Common Challenges

Despite their effectiveness, all approaches face certain challenges :

Codebook Approaches :

— Computational complexity in segmentation and feature extraction processes
— Time-intensive operations, especially in large-scale applications
— High memory requirements for ensemble methods

Contour-Based Approaches :

— Sensitivity to document quality, noise, and preprocessing variations
— Performance affected by document degradation, ink bleeding, and background noise
— Computational complexity in rotation-invariant descriptors

Texture-Based Approaches :

— Feature dimensionality issues
— Sensitivity to image quality and preprocessing parameters
— Dataset dependencies for optimal performance

Advantage and disadvantage of text-independent offline writer identification methods

Categories	**References**	**Features**	**Advantage**	**Disadvantage / Accuracy**
Codebook-based	Bensefia et al., 2005 ; Bulacu et al., 2007	Bag of graphemes	Forensic-style modeling, informative	IFN/ENIT : 94.9% (BDCT, Khan et al., 2017)
	He et al., 2015 ; Khalifa et al., 2015	Junclets	Junction-based fragmentation of characters	Firemaker : 80.6% (He et al., 2015)
	Abdi and Khemakhem, 2015 ; Khan et al., 2017	Elliptic graphemes, BDCT	Versatile shape representation	CVL : 99.6% (Khan et al., 2017)

(continued on next page)

(continued from previous page)

Categories	**References**	**Features**	**Advantage**	**Disadvantage / Accuracy**
	Durou et al., 2019 ; Bennour et al., 2019	Implicit shapes (key-points)	Sparse key-points, flexible	BFL : 98.3% (Bennour et al., 2019)
Texture-based	Bulacu et al., 2007 ; Siddiqi and Vincent, 2010	Contour-Hinge, LBP	Fast, no learning phase	IFN/ENIT : 98.5% (Singh et al., 2018)
	Bertolini et al., 2013 ; Wu et al., 2014	LPQ, SIFT, SOH	Robust descriptors, no training	IAM : 98.5% (Wu et al., 2014)
	Singh et al., 2018 ; Chahi et al., 2020	LSTP, CSLBCoP	Accurate region-based analysis	CVL : 100.0% (Chahi et al., 2020)
	Hannad et al., 2016 ; Kessentini et al., 2018	HOG, GLRL, Edge-Hinge	Works on noisy data	CERUG-CN : 100.0% (Chahi et al., 2020)
	Kumar and Sharma, 2019	DDC, CA descriptors	Describes stroke structure effectively	ICDAR2013 : 98.4% (Chahi et al., 2020)
Contour-based	Bulacu and Schomaker, 2007	Edge direction, hinge histograms	Language-independent, multi-script	89.4% (multi-database avg)
	Brink et al., 2012	Ink-trace width features	Captures pressure/stroke dynamics	91.8% (contour width descriptor)
	He and Schomaker, 2015	Delta-n Hinge (rotation-invariant)	Orientation-robust, good for archives	93.2% (historical dataset)
	He et al., 2014	Contour and stroke fragments	Style dating + identification	89.7% (historic manuscripts)
	Bahram, 2012	Contour + texture fusion	Hybrid of structure and texture	87.5% (hybrid descriptor)

2.6 Conclusion

Writer identification in multilingual contexts remains an open and challenging field, especially given the variability of scripts and the differences in text length. While prior benchmarks such as ICDAR 2011 and ICDAR 2013 have focused primarily on monolingual (Latin script) scenarios, they continue to serve as valuable datasets for evaluating new approaches.

In this research, we proposed two novel, text-independent features for writer identification and evaluated their effectiveness on both short and long handwritten samples. Using the ICDAR 2011 and 2013 datasets, we demonstrated that these features can capture writer-specific traits with high discriminative power across varying conditions.

The experimental results, detailed in the next chapters, confirm the potential of our features to generalize beyond monolingual settings and limited text lengths. These findings offer promising directions for the development of robust, language-independent writer identification systems applicable to diverse real-world scenarios.

Chapitre 3

Proposed Technique

Science is the process of finding patterns in the chaos of nature.

Richard Feynman

3.1 Introduction

This section presents the methodology of the proposed offline writer identification system, *USAWRS* (University of Saïda Automatic Writer Recognition System), designed for robust, text-independent writer identification across multilingual handwritten documents. The system processes scanned handwriting images using two primary representations during feature extraction : binary connected-components and their exterior contours. The contour, defined as a sequence of pixels on the ink-background boundary of a connected-component, provides an efficient vectorial representation that enables rapid feature computation while capturing essential writing shapes. These representations are critical for encoding writer-specific attributes, such as direction, slant, curvature, ink-trace width, and letter shapes.

For writer characterization, we propose two complementary features based on the *Modified Local Binary Pattern with Ink Width and Stroke Length* (MLBP-IWSL) framework :

— **F1 : MLBP-IWSL (black pixels)** : Extracts MLBP-IWSL features from black pixels (ink traces), capturing local texture patterns and structural attributes of the handwriting, such as stroke width and length variations.

— **F2 : MLBP-IWSL (white pixels)** : Computes MLBP-IWSL features from white pixels (background regions), encoding complementary texture and contour information to enhance writer discriminability.

These features are fused to combine the strengths of F1 and F2, leveraging their complementary information to improve identification accuracy. The MLBP-IWSL features are text-independent, allowing the system to identify writers without relying on the textual content of the samples. By using probability distribution functions derived from contour-based texture analysis, F1 and F2 effectively capture writer individuality, focusing on stylistic nuances across multilingual scripts.

As illustrated in Figure 3.1, the proposed approach comprises three main processing steps :

1. **Pre-processing** : Normalizes and enhances scanned handwriting images to ensure consistent feature extraction across diverse datasets.

2. **Feature Extraction** : Computes F1 (MLBP-IWSL black pixels) and F2 (MLBP-IWSL white pixels) from binary connected-components and contours, followed by their fusion to encode writer-specific characteristics.
3. **Classification (Identification)** : Employs distance metrics (e.g., Chi-Squared) and fusion strategies (e.g., SUM rule) to match query samples against a database of known writers.

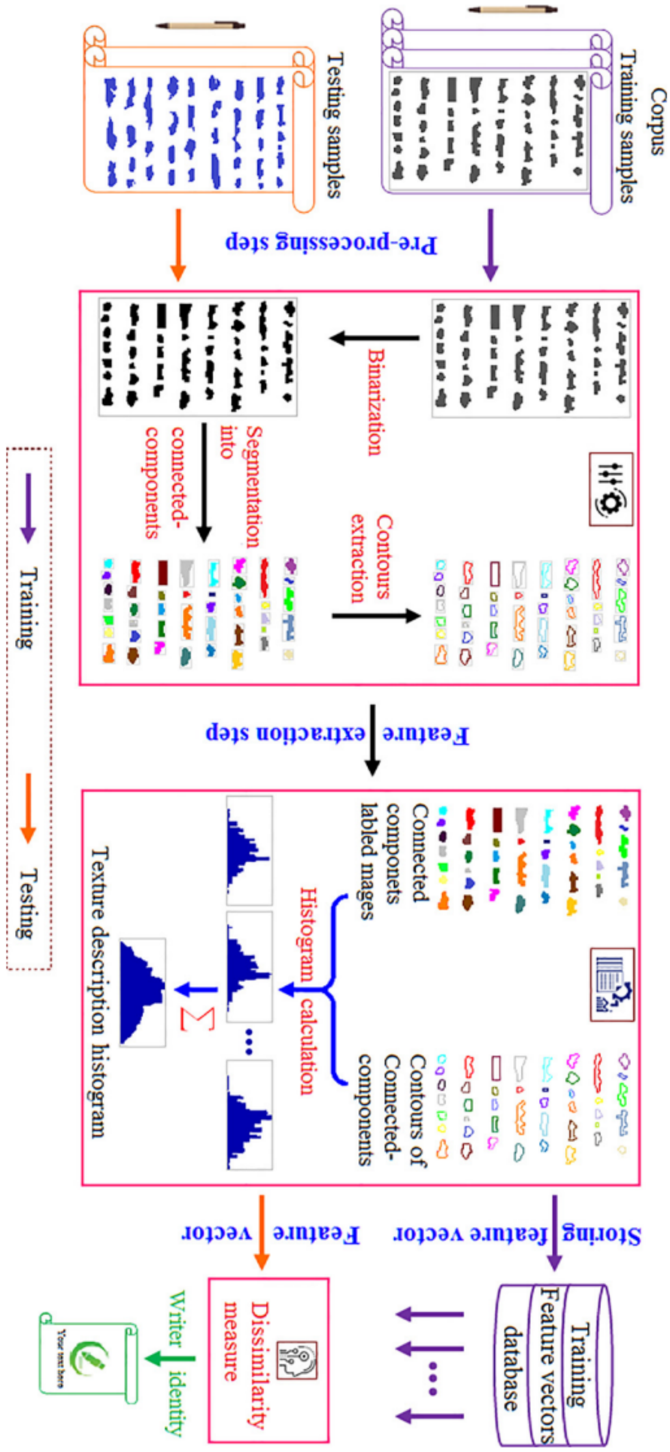

FIGURE 3.1 – The framework of our writer identification system.

Each step is detailed in the following subsections, providing a comprehensive overview of the methodology and its implementation on multilingual datasets, such as the ICDAR 2011 Writer Identification Contest dataset (208 documents from 26 writers) and the ICDAR 2013 Writer Identification Competition dataset (1,000 documents from 250 writers), which include handwritten samples in English, Greek, French, and German. The *USAWRS* system minimizes human intervention, aligning with the goal of developing a scalable and practical solution for forensic and historical document analysis.

3.2 Pre-processing

As with many tasks in pattern recognition, the preprocessing phase is a vital step that significantly impacts the overall performance and accuracy of the writer verification system. This stage involve preparing and cleaning the input data to enhance the system's ability to correctly recognize and differentiate between different handwriting styles.
In our experiments, the input data are composed of black connected components that capture the handwriting, along with their associated contours. so the pre-processing step begins by binarizing the document images to distinguish the foreground (handwriting) from the background. Next, connected components are extracted from the binarized image. Non-significant components such as small noise elements, isolated dots, or specks are then removed. Finally, the process involves extracting both the inner and outer contours of the remaining connected components for further analysis[11].

3.2.1 Binarisation :

is a key step in preparing images for analysis. It takes a grayscale or color image and simplifies it by turning it into a black and white picture, where only two colors exist : black for important details like text or handwriting, and white for the background, like the paper. This simplification makes it much easier to perform tasks such as recognizing characters, verifying writers, or detecting shapes and outlines. Depending on how this threshold is determined, binarisation methods are commonly divided into two main categories :

— **Global Binarisation :** Global binarisation applies a single threshold value to the entire image. Each pixel is classified as foreground or background based on this one global threshold.
  Advantages : Simple, fast, and effective for images with uniform lighting and contrast.
  Common method : Otsu's method, which automatically selects an optimal threshold by maximizing inter-class variance.
— **Local Binarisation :** Local binarisation, also known as adaptive binarisation, computes a different threshold for each pixel based on the local neighborhood (a small window around each pixel). This makes it more suitable for documents with irregular lighting, shadows, and low quality.
  Advantages : Durable to variations in lighting and background noise.
  Common methods : Niblack, Sauvola, and Wolf algorithms, which dynamically adjust thresholds using local mean and standard deviation.

Aspect	**Global Thresholding**	**Local Thresholding**
Characteristics	Uses a single threshold value for the entire image. It is less effective for images with uneven illumination.	Computes thresholds based on neighboring pixel intensities, adapting to local illumination and contrast variations.
Threshold Value Determination	Threshold is constant across the image.	Thresholds vary across different image regions, based on local statistics.
Illumination Handling	Performs poorly under varying lighting ; suitable for uniformly illuminated images.	Robust to lighting variation ; ideal for non-uniformly lit scenes.
Computational Complexity	Computationally efficient and fast.	Requires more processing due to per-pixel or per-region computations.
Use Cases	Best suited for clean, scanned documents or uniformly lit environments.	Preferred for natural scenes or images with shadows and inconsistent lighting.

TABLE 3.1 – Comparison between Global and Local Thresholding[56]

3.2.1.1 Binarisation Methodes :

1. Otsu's Methode (Global Thresholding) : Otsu's method is a nonparametric, unsupervised technique for automatic image thresholding. It selects the optimal threshold by maximizing the between-class variance using histogram moments. The method is simple, efficient, and extendable to multilevel thresholding.[59]

$$\sigma_b^2(t) = \omega_1(t)\omega_2(t)\left[\mu_1(t) - \mu_2(t)\right]^2$$

Where :
- ω_1, ω_2 are the class probabilities
- μ_1, μ_2 are the class means

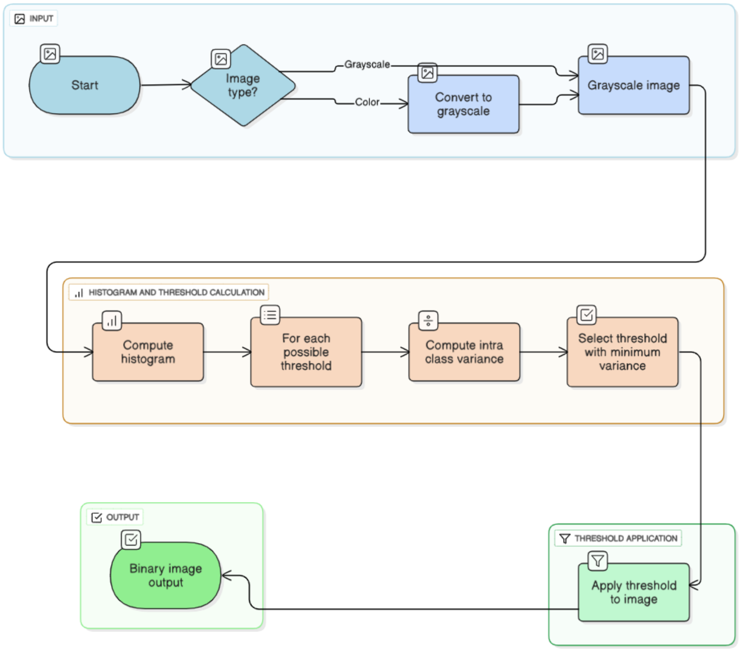

FIGURE 3.2 – Workflow of Otsu's Thresholding Method for Image Binarization

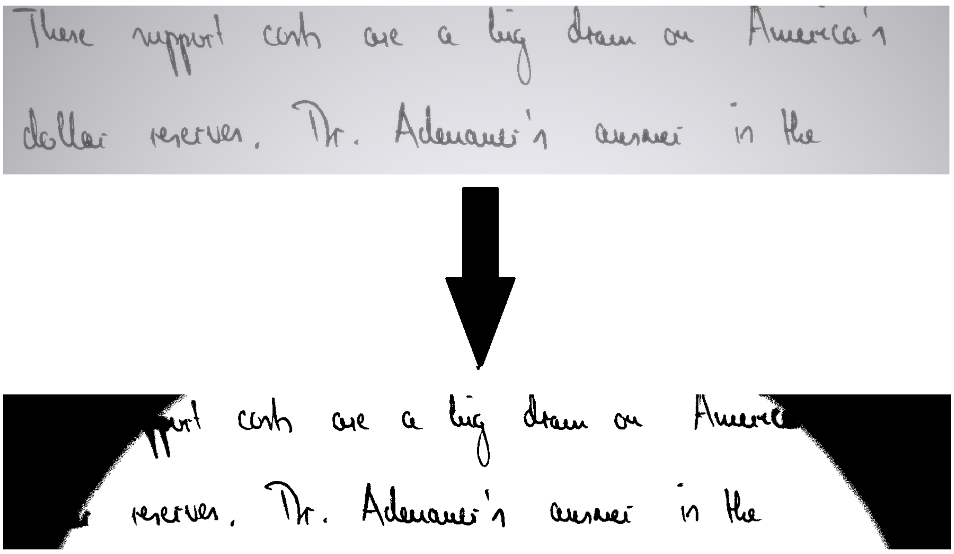

FIGURE 3.3 – Global Image Thresholding using Otsu's Method

2. Niblack's Methode (Local Thresholding) : Niblack's method is a local thresholding technique used in image processing to segment an image into foreground and background regions. It computes the threshold for each pixel based on the mean and standard deviation of the pixel intensities in a local neighborhood[56].

$$T(x,y) = m(x,y) + k \cdot s(x,y)$$

Where :
- $m(x,y)$ is the local mean
- $s(x,y)$ is the local standard deviation

— k is a tunable parameter, usually between -0.5 and 0.5

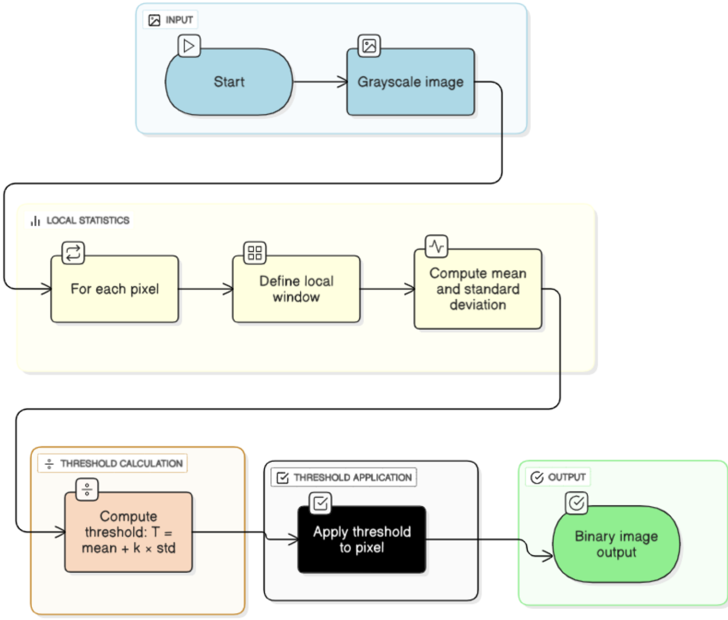

FIGURE 3.4 – Workflow of Niblack's Thresholding Method for Image Binarization

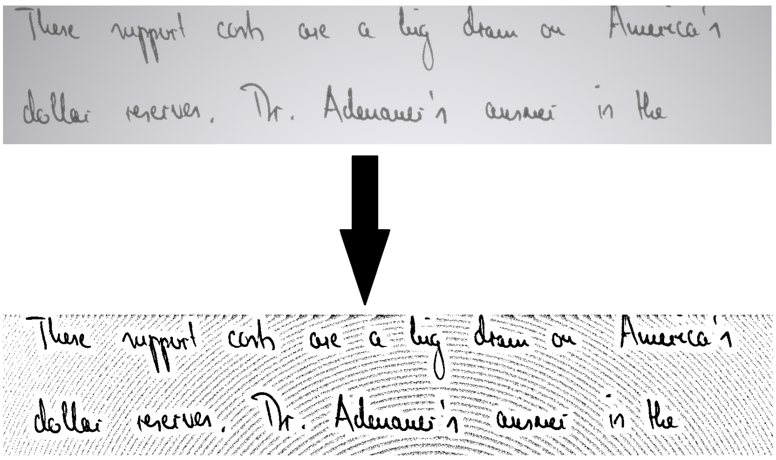

FIGURE 3.5 – local Image Thresholding using Niblack's Method

3. Sauvola's Methode (Improved Local Thresholding) : Sauvola's method improves upon Niblack's adaptive thresholding by handling uneven illumination. It uses the local mean and standard deviation with a dynamic range parameter for better adaptability ,This makes it effective for binarizing document images with text and background noise[56].

The formula for the threshold is :

$$T(x,y) = \mu(x,y)\left[1 + k\left(\frac{\sigma(x,y)}{R} - 1\right)\right]$$

Where :

- R is the dynamic range of standard deviation (typically 128)
- k is typically in the range [0.3, 0.5]

.65.65

FIGURE 3.6 – Workflow of Sauvola's Thresholding Method for Image Binarization

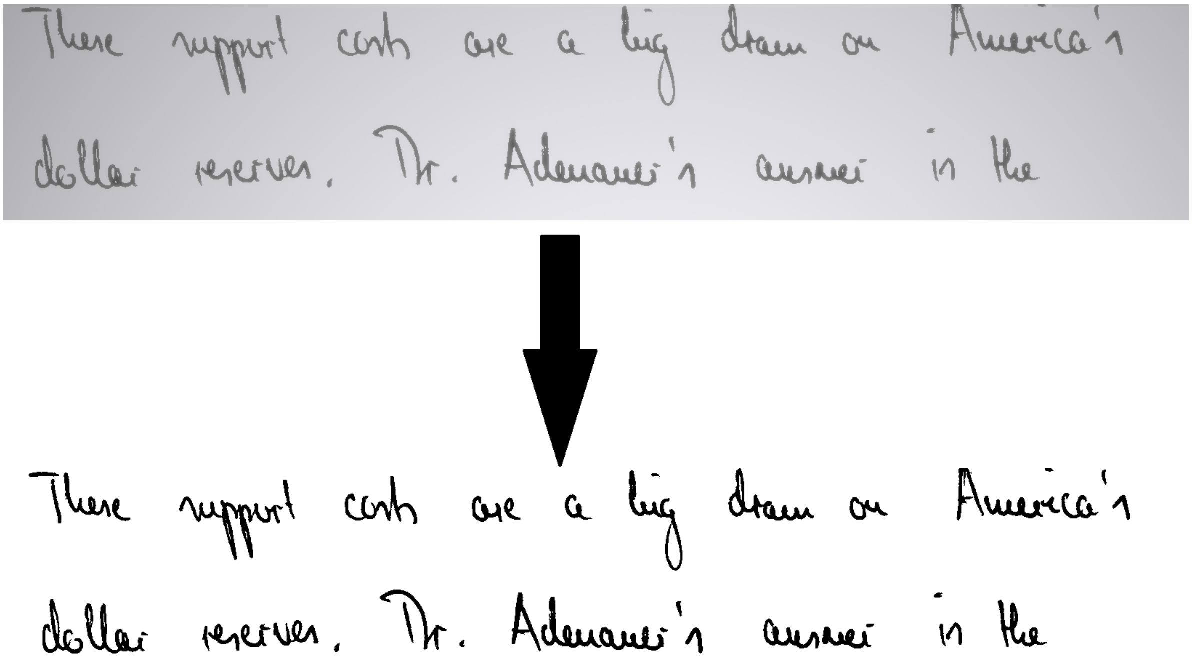

FIGURE 3.7 – local Image Thresholding using sauvola's Method

4. K-means Clustering : K-meansis an unsupervised algorithm and it is used to segment the interest area from the background. It clusters, or partitions the given data into K-clusters or parts based on the K-centroids(typicaly k=2)[45].

Method	**Type**	**Lighting**	**Parameters**	**Noise Robustness**	**Best For**
Otsu	Global	Uniform	No	Low	Clean scanned documents
Niblack	Local	Irregular	Yes	Medium	Historical, noisy text
Sauvola	Local	Irregular	Yes	High	Noisy/low contrast docs
K-means	Clustering	Any	Yes	High	Degraded/irregular docs

TABLE 3.2 – Comparison of Binarisation Methods

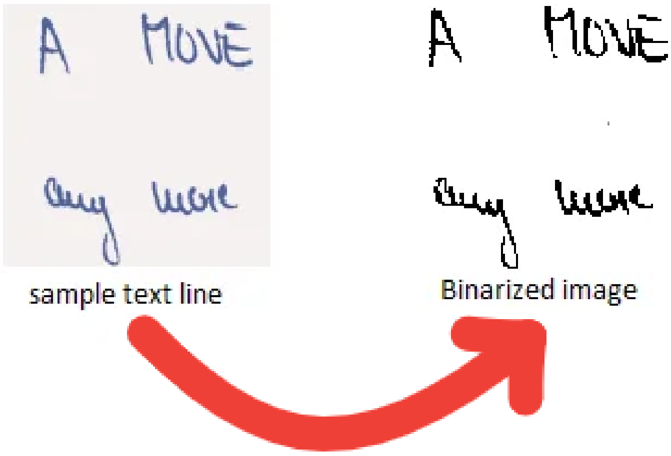

FIGURE 3.8 – Binarization Process

For our work, we decided to use Otsu's method to binarise the handwriting images. The main reason is that our images are mostly clean and well-scanned, with good lighting and a clear difference between the text and the background. Otsu's method is great in this kind of situation because it automatically finds the best threshold to separate the black writing from the white background. It's also fast and simple to use, which made it a practical choice. By using Otsu's method, we were able to get clean binary images that made the next steps, like contour detection and writer analysis, much easier and more reliable.

3.2.2 Connected Components Extraxtion :

Connected components extraction is a key step in processing binary images where the goal is to identify and separate individual objects made up of connected pixels. In a black-and-white image, for example, the black pixels usually represent meaningful content such as handwritten letters, symbols, or shapes, while the white pixels represent the background.

The extraction process works by examining the image to find groups of black pixels that are connected to each other based on a connectivity rule. Typically, this rule considers either 4-connectivity (pixels connected horizontally or vertically) or 8-connectivity (including diagonal neighbors). Pixels that meet this connectivity condition are grouped together as one connected component.

Once these groups are identified, each connected component is assigned a unique label. This labeling allows us to treat each component as an individual object for further analysis. For example, in handwriting recognition or writer identification, connected components often correspond to letters, strokes, or words that need to be analyzed separately.

This step is crucial because it simplifies complex images by breaking them down into smaller, meaningful parts. After extracting connected components, other important tasks such as contour tracing, feature extraction, or classification become more manageable and accurate[25].

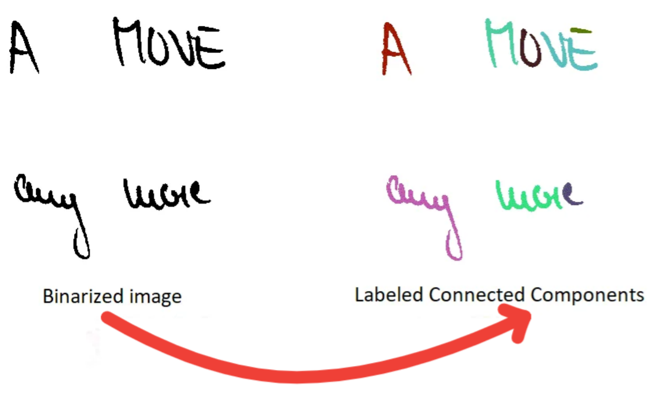

FIGURE 3.9 – Connected Components Extraxtion Process

3.2.3 Removal of Small Connected Components :

This is an important cleaning step in image processing, especially after connected components extraction. When working with binary images, small groups of connected pixels often appear due to noise, dust, or minor imperfections in the scanning process. These tiny components usually do not carry useful information—for example, random dots, specks, or small artifacts—and can interfere with later analysis steps.

To improve the quality of the data, these small connected components are identified and removed based on their size, typically by setting a minimum pixel area threshold. Any connected component smaller than this threshold is discarded from the image. This helps to reduce noise and focus on the meaningful parts of the image, such as actual letters or handwriting strokes.

Removing small connected components is essential to improve the accuracy of tasks like writer identification, character recognition, or contour extraction, because it ensures that only significant and relevant parts of the image are processed.

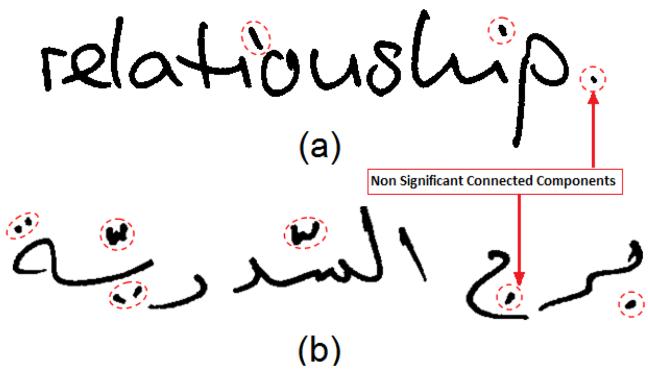

FIGURE 3.10 – Non significant Connected Components Detection

3.2.4 Contour Detection :

Contour detection is the process of tracing the outlines or borders of shapes in a binary image. In the context of handwriting analysis, these shapes are the connected components representing individual letters or strokes. This step helps us understand the exact form and structure of each component, which is essential for detailed analysis.

There are two types of contours we usually extract :

— **Outer contours**, which trace the external boundary of a shape. For example, in the letter "O", the outer edge forms the main visible border.

— **Inner contours**, which trace holes or enclosed spaces inside a shape like the circular gap inside the letter "O" or "P".

To perform contour detection, one commonly used method is **the Moore-Neighbor Tracing Algorithm**, which is simple and effective for binary images[35].

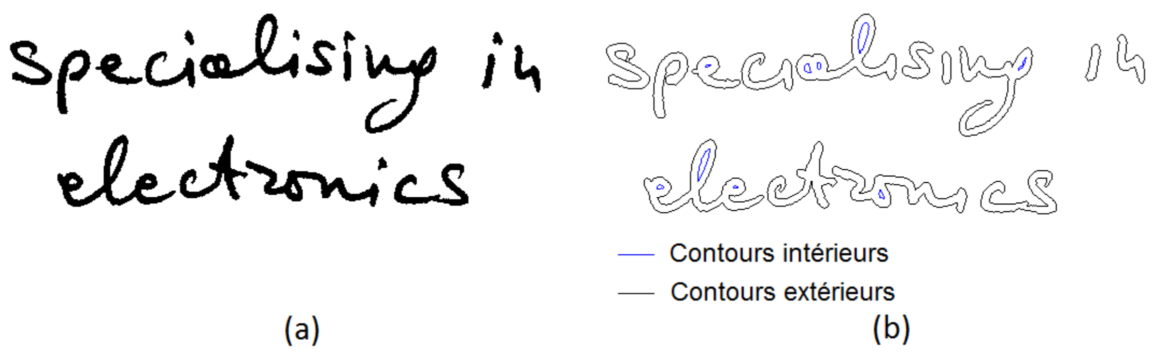

FIGURE 3.11 – Outer-Iner Contour Detection

The Moore-Neighbor Tracing Algorithm :

The Moore-Neighbor Tracing algorithm is a classical method used to extract the boundary (or contour) of a shape in a binary image. It is especially useful when the shape consists of connected pixels, such as characters in handwriting, and we want to trace its exact border[35].

3.2.4.1 Moore Neighborhood :

The Moore neighborhood of a pixel (also known as the 8-neighbors), P, is the set of 8 pixels that share either a vertex or an edge with P. These neighboring pixels are denoted as $P_1, P_2, ..., P_8$ as illustrated in the figure below[35].

3.2.4.2 Algorithm Idea :

Given a digital pattern (a group of black pixels) on a white background arranged in a grid, we first locate a black pixel and declare it as the *start* pixel. This can be done by scanning from the bottom to the top and left to right until a black pixel is found. Imagine a ladybug placed on this start pixel. The goal is to trace the contour by walking around the shape in a clockwise direction. At every step, when a black pixel P is reached, we backtrack to the white pixel from which we entered P, and then explore all pixels in the Moore neighborhood of P in a clockwise order until another black pixel is found. The tracing continues until the start pixel is visited again in the same manner.

All the black pixels visited during this process form the boundary or contour of the shape[35].

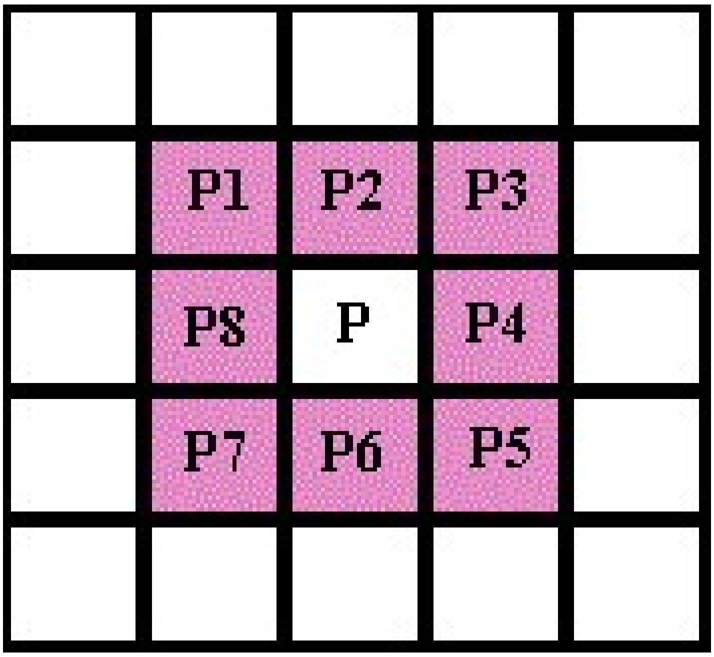

FIGURE 3.12 – The Moore neighborhood of a pixel P[35]

3.2.4.3 Formal Algorithm :

Input : A square grid pattern T containing a connected component P of black cells.
Output : A sequence $B = (b_1, b_2, ..., b_k)$ of boundary pixels (the contour).

1. Initialize B as an empty list.
2. Scan T from bottom to top and left to right until a black pixel $s \in P$ is found.
3. Insert s into B.
4. Set the current boundary point $p := s$.
5. Backtrack to the pixel from which s was entered.
6. Set c to be the next clockwise pixel in the Moore neighborhood $M(p)$.
7. **While** $c \neq s$:
   - If c is black :
     - Insert c into B
     - Set $p := c$
     - Backtrack to the pixel from which p was entered
   - Else :
     - Advance c to the next clockwise pixel in $M(p)$
8. End While

Result : The sequence B contains the coordinates of the contour pixels[35].

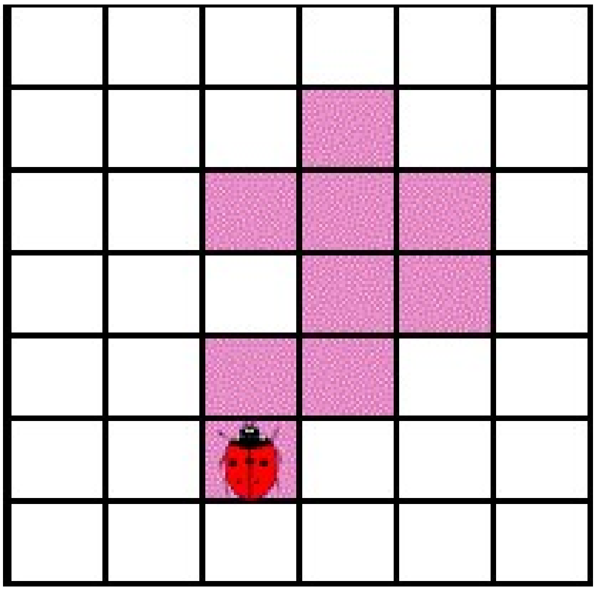

FIGURE 3.13 – standing on the start pixel[35]

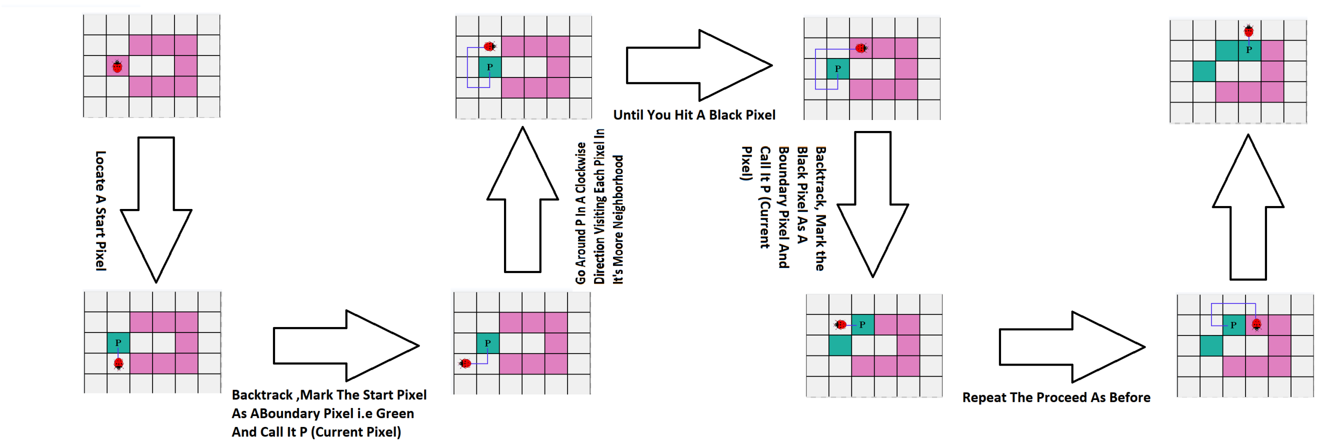

FIGURE 3.14 – description of the Moore-Neighbor tracing algorithm[35]

3.3 Feature Extraction

3.3.1 Feature Fusion

Fusion techniques are recognized for improving performance and have been utilized in various classification tasks in general [55], and in biometric applications specifically [32, 64, 63]. Fusion can be performed at the feature level [74, 60], where diverse features capturing different types of information are integrated, or at the decision level [75, 47], where outputs from multiple classifiers are combined to improve overall system performance.

This approach calculates a single numerical value, called a scalar score, which represents how different or similar the input sample is compared to the claimed identity. In the context of writer verification, this score helps decide whether the handwriting sample belongs to the claimed writer or not.

To get this scalar score, the system first extracts multiple dissimilarity measures from the input sample. Each dissimilarity measure comes from a different textural descriptor—these are algorithms or features that capture various patterns and textures in the handwriting. Since different descriptors focus on different aspects of the handwriting (such as stroke thickness, curvature, or texture), combining their results provides a more comprehensive evaluation[37].

Once the individual dissimilarity scores are computed, they need to be combined into one final score. This is done using fusion rules, which are mathematical methods that integrate multiple values into a single representative number. Common fusion rules include :

- **Sum rule** : Adding all the individual scores together. This assumes that all descriptors contribute equally to the final decision[37].

  Let :
  - D_i denote the distance (or dissimilarity score) obtained from the i-th feature.
  - N be the total number of features.
  - $D_i \in \mathbb{R}$, and a special value (e.g., $D_i = \text{DBL\_MAX}$) indicates an invalid or undefined feature score.

  Then the normalized sum fusion distance D_{fused} is defined as :

$$D_{\text{fused}} = \begin{cases} \frac{1}{N} \sum_{i=1}^{N} D_i, & \text{if } D_i \neq \text{DBL\_MAX} \quad \forall i \\ \text{DBL\_MAX}, & \text{if any } D_i = \text{DBL\_MAX} \end{cases}$$

  Key Characteristics
  - **Robustness :** If any feature yields an invalid distance (DBL_MAX), the fused distance is set to DBL_MAX to flag unreliability.
  - **Normalization :** The sum of all valid distances is divided by the number of features to ensure scale consistency.
  - **Use Case :** This rule is typically applied when all features are expected to contribute equally and are on a comparable scale.
- **Product rule** : Multiplying all the scores. This method emphasizes agreement among descriptors since one very low score can reduce the total significantly[37].

  Let :
  - D_i be the distance from the i-th feature.
  - N be the total number of features.

$$D_{\text{fused}} = \begin{cases} \prod_{i=1}^{N} D_i, & \text{if } D_i \neq \text{DBL\_MAX} \quad \forall i \\ \text{DBL\_MAX}, & \text{if any } D_i = \text{DBL\_MAX} \end{cases}$$

  Key Characteristics
  - **Amplifies large differences :** Sensitive to outliers and large distances.
  - **Multiplicative :** Strongly penalizes any high distance.
  - **Use Case :** Useful when features must all be jointly low to indicate similarity..
- **Min rule** : Taking the smallest (minimum) score among the descriptors. This could be useful when one strong match is enough to accept the sample[37].

$$D_{\text{fused}} = \begin{cases} \min_{i=1}^{N} D_i, & \text{if } D_i \neq \text{DBL\_MAX} \quad \forall i \\ \text{DBL\_MAX}, & \text{if any } D_i = \text{DBL\_MAX} \end{cases}$$

Key Characteristics
- **Pessimistic :** Prioritizes the worst-case (highest) distance
- **Robust to low distances :** One bad score dominates.
- **Use Case :** Suitable when any dissimilarity should trigger rejection..

— **Max rule** : Taking the largest (maximum) score, which might reflect the worst-case dissimilarity[37].

$$D_{\text{fused}} = \begin{cases} \max_{i=1}^{N} D_i, & \text{if } D_i \neq \text{DBL\_MAX} \quad \forall i \\ \text{DBL\_MAX}, & \text{if any } D_i = \text{DBL\_MAX} \end{cases}$$

Key Characteristics
- **Optimistic :** Considers the best-case (smallest) distance.
- **Normalization :** May ignore bad scores : One good match can dominate.
- **Use Case :** Suitable when any strong similarity is sufficient.

By applying these fusion rules, the system consolidates the diverse information from multiple descriptors into a single meaningful score, improving the robustness and accuracy of the verification process.

Fusion Rule	Formula	Intuition	Strengths	Weaknesses	Typical Use Case
Sum	$\sum_{i=1}^{l} c_{i,j}$	Aggregate all evidence additively	Simple, effective, smooth	Sensitive to scale, may dilute strong signals	When features complement each other
Product	$\prod_{i=1}^{l} c_{i,j}$	Emphasize consensus	Penalizes disagreement, more discriminative	Sensitive to low values, assumes independence	When all features must agree
Max	$\max_{i=1}^{l} c_{i,j}$	Take strongest evidence	Robust to weak features	Can be optimistic, ignores other sources	When one feature is highly reliable
Min	$\min_{i=1}^{l} c_{i,j}$	Take weakest evidence	Conservative, reduces false acceptance	Can be overly pessimistic	High-security scenarios requiring consensus

Table 3.3 – Summary of Fusion Rules[4]

3.3.2 Distance Calculation

In this work ,we measure the degree of dissimilarity between two samples by applying various distance functions such as Chi-square (χ^2), Manhattan, Euclidean, Minkowski

3.3.2.1 Chi-square (χ^2) Distance

Given two feature vectors T_1 and T_2, the distance function computes a value that quantifies the difference between them while normalizing for the scale of their individual elements.

Formally, the Chi-square distance D between two vectors of equal length n is defined as :

$$D(T_1, T_2) = \sum_{i=1}^{n} \frac{(T_1[i] - T_2[i])^2}{T_1[i] + T_2[i]}$$

where $T_1[i]$ and $T_2[i]$ represent the i^{th} components of vectors T_1 and T_2, respectively.

The implementation follows these key steps :

Dimension Check : The function first verifies that both input vectors have the same dimension. This ensures that the comparison is valid and element-wise correspondence exists.

Selective Computation : To avoid division by zero or meaningless terms, the function only includes in the summation those indices where at least one of the vector components is non-zero.

Distance Accumulation : For each valid index, the squared difference of the components is divided by their sum, effectively normalizing the difference by the combined magnitude of the components. This emphasizes relative differences over absolute differences, making the metric robust to scale variations.

This distance metric is widely used in pattern recognition and image processingand is commonly used in hypothesis testing to determine if two histograms come from the same distribution[66], due to its sensitivity to relative changes rather than absolute magnitudes. By using this function, we ensure that the computed distances are meaningful even when component values vary widely in scale or distribution.

The method provides an efficient $O(n)$ time complexity, where n is the length of the vectors, making it suitable for high-dimensional data comparison in real-time or large-scale applications.

3.3.2.2 Manhattan Distance

Given two feature vectors T_1 and T_2, the Manhattan distance function calculates the total absolute difference between their corresponding elements.

Formally, the Manhattan distance D between two vectors of equal length n is defined as :

$$D(T_1, T_2) = \sum_{i=1}^{n} |T_1[i] - T_2[i]|$$

where $T_1[i]$ and $T_2[i]$ are the i^{th} components of the vectors T_1 and T_2, respectively.

The implementation follows these steps :

Dimension Check : The function first ensures that both vectors have the same length. This guarantees a valid one-to-one element comparison.

Distance Accumulation : For each element index i, the absolute difference between the corresponding elements of the two vectors is calculated using `Math::Abs()`, and then added to the cumulative distance.

This approach effectively measures how different the two vectors are in terms of individual element values.

The Manhattan distance is commonly used in machine learning, clustering, and pattern recognition tasks where the total deviation is more relevant than the squared deviation (as in Euclidean distance). It is particularly useful in high-dimensional spaces where the geometry of data behaves differently[30].

The algorithm has a linear time complexity of $O(n)$, where n is the length of the vectors, making it suitable for large-scale applications or real-time systems.

3.3.2.3 Euclidean Distance

The Euclidean distance function calculates the straight-line distance between two feature vectors T_1 and T_2 in multi-dimensional space.

Formally, the Euclidean distance D between two vectors of equal length n is defined as :

$$D(T_1, T_2) = \sqrt{\sum_{i=1}^{n} (T_1[i] - T_2[i])^2}$$

where $T_1[i]$ and $T_2[i]$ denote the i^{th} components of the vectors T_1 and T_2, respectively.

The implementation consists of the following steps :

Dimension Check : The function begins by confirming that both vectors have the same length to ensure a valid element-wise comparison.

Distance Accumulation : A loop iterates over each index i, computing the square of the difference between corresponding elements. These squared differences are accumulated into the variable `distance`.

Square Root Application : After the loop, the square root of the accumulated value is taken to obtain the final Euclidean distance, which reflects the true geometric distance between the two vectors.

It is one of the most commonly used distance metrics in pattern recognition, machine learning, and statistical analysis.

This method provides a reliable and interpretable measure of similarity or difference between vectors. It has a time complexity of $O(n)$, where n is the dimensionality of the input vectors.

3.3.2.4 Minkowski Distance

The Minkowski distance is a generalization of both the Euclidean and Manhattan distances, controlled by a parameter $r \geq 1$. It calculates the distance between two feature vectors T_1 and T_2 of equal length by raising the absolute difference of each component to the power of r, summing all results, and then taking the r-th root of the total.

Formally, the Minkowski distance D is defined as :

$$D(T_1, T_2) = \left(\sum_{i=1}^{n} |T_1[i] - T_2[i]|^r \right)^{\frac{1}{r}}$$

where :

— $T_1[i]$ and $T_2[i]$ are the i^{th} components of the input vectors,

— n is the dimension (length) of the vectors,
— r is the order of the distance.

The implementation consists of the following steps

— **Dimension Check :** The function first confirms both vectors are of equal length.
— **Distance Calculation :** It iterates through each index i, computes $|T_1[i]-T_2[i]|^r$, and accumulates the result.
— **Final Root :** After accumulation, the r^{th} root is applied to compute the final distance.

This metric adapts based on the value of r :

— $r = 1$ yields the Manhattan distance,
— $r = 2$ gives the Euclidean distance,
— Higher values of r increasingly emphasize larger component differences.

The algorithm has a time complexity of $O(n)$, making it efficient for comparing high-dimensional feature vectors in various pattern recognition and machine learning applications.

Comparison of Distance Metrics

Distance Metric	**Key Characteristics**	**Advantages**	**Time Complexity**
Chi-square (χ^2)	- Normalizes difference by sum of components. - Sums only where denominator $\neq 0$. - Sensitive to relative differences.	- Robust to scale variations. - Effective for histogram comparisons. - Used in image processing.	$O(n)$
Manhattan	- Sum of absolute differences. - Measures total deviation.	- Simple and interpretable. - Works well in high dimensions. - Less sensitive to outliers.	$O(n)$
Euclidean	- Straight-line distance in n-dimensional space. - Squares differences before summing.	- Common in pattern recognition. - Intuitive geometric meaning. - Sensitive to outliers.	$O(n)$
Minkowski	- Generalizes L1 ($r = 1$) and L2 ($r = 2$). - $r \geq 1$ adjusts sensitivity.	- Flexible for different use cases. - Emphasizes large differences with higher r.	$O(n)$

3.3.3 Run-Length

The concept of Run-Length (RL) distribution was first introduced by Arazi in 1977[5] as one of the earliest features for automatic writer recognition. Since then, it has gained recognition as an effective global descriptor for capturing the unique stylistic traits of handwritten text. The fundamental idea behind this method is to analyze sequences of pixels called *runs* that share similar visual properties across specified directions in the image.

A run is defined as a series of consecutive, connected pixels that exhibit the same characteristic, such as pixel intensity[29]. In the context of binary images (i.e., black-and-white images where black denotes ink and white represents the paper background), run-length analysis is used to measure the lengths of such sequences. These sequences are measured separately for black pixels (foreground) and white pixels (background).

The white pixel runs provide valuable insight into the spatial layout of the handwriting, including intra- and inter-letter spacing as well as the curvature and openness of character shapes. Conversely, the black pixel runs convey details about the writing instrument, particularly the thickness of the strokes or the width of the ink traces, which are highly characteristic of a writer's style.

Run-length measurements can be conducted in multiple orientations to capture directional textural features. The four commonly used directions are :

— Horizontal (0°)
— Vertical (90°)
— Diagonal (45°)
— Diagonal (135°)

In each of these directions, the lengths of the detected runs are compiled into histograms. These histograms are then normalized to form probability distribution functions (PDFs), which effectively describe the textural structure of the handwriting. These PDFs serve as discriminative features for identifying individual writers.

To further enhance the analysis beyond binary images, **Gray-Level Run-Length Matrices (GLRLMs)** can be employed. This extension allows the system to work directly with grayscale images without needing a binarization step[28]. In this case, a run is defined as a sequence of neighboring pixels that all share the same gray-level intensity g. The GLRLM stores this information in a matrix $M(g, h)$, where each element corresponds to the number of runs with gray level g and length h.

GLRLMs can also be computed in the four principal directions (0°, 45°, 90°, and 135°) to capture multi-directional textural information[61]. Once generated, these matrices are likewise converted into normalized histograms, providing compact statistical representations of the textural patterns found in handwriting. These descriptors are then used as part of the writer identification system to distinguish between different writing styles.

An example illustrating how run-lengths are computed across the four directions is provided in Fig. 4.8, which demonstrates the visual formation of both black and white run-length distributions in different orientations.

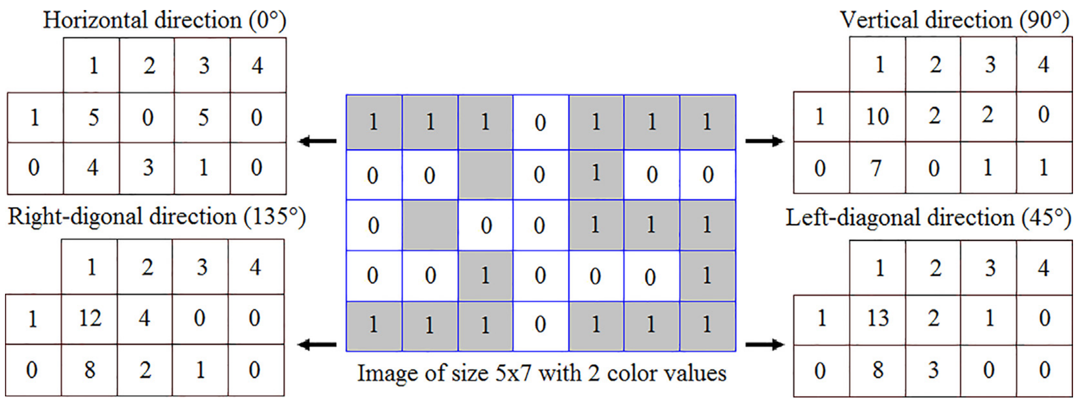

FIGURE 3.15 – Calculation of run-length matrices

3.3.4 Ink-trace Width and Shape Letters (IWSL)

An IWSL measurement refers to the number of consecutive black and white pixels located between two exterior contour points along a specific direction in a binary image. Formally, the IWSL can be defined as follows :

Let S be a binary sequence of connected pixels in a bi-level (binary) image, where each pixel is represented as either 1 (black) or 0 (white). Let p_{cs} and p_{ce} denote the starting and ending positions of the sequence S, and let $p_1, \ldots, p_n$ represent the pixels lying between the two exterior contour points.

The IWSL is then computed as the Euclidean distance between the two exterior contour points :

$$p_{cs} = (x_{cs}, y_{cs}), \quad p_{ce} = (x_{ce}, y_{ce})$$
$$\text{IWSL} = \sqrt{(x_{ce} - x_{cs})^2 + (y_{ce} - y_{cs})^2} \tag{3.1}$$

This measurement provides a numeric estimation of the stroke length between two contour points in a specific direction[11].

The IWSL computation for exterior contour pixels is based on techniques similar to those described in [28, 29, 43]. The measurement is evaluated along four main directions to effectively capture structural writing details such as character shapes, average letter widths, and ink stroke width :

— Horizontal direction : IWSL_1
— Vertical direction : IWSL_2
— Left-diagonal direction : IWSL_3
— Right-diagonal direction : IWSL_4

3.3.4.1 IWSL for Black and White Pixel Analysis

The IWSL measurement can be applied to both black pixels (ink traces) and white pixels (background regions) to capture comprehensive structural information about handwriting patterns. This dual analysis provides complementary features that enhance the discriminative power of the IWSL descriptor.

Black Pixel IWSL ($\text{IWSL}^{(b)}$) The traditional IWSL measurement focuses on black pixels, which represent the actual ink traces of the handwriting. Black pixel IWSL captures :

— Stroke thickness variations characteristic of individual writing instruments
— Ink flow patterns that reflect writing pressure and speed
— Character formation details and pen trajectory information
— Local structural properties of letter shapes and connections

For black pixels, the IWSL is computed along sequences of consecutive black pixels between exterior contour points :

$$\text{IWSL}_k^{(b)} = \sqrt{(x_{ce}^{(b)} - x_{cs}^{(b)})^2 + (y_{ce}^{(b)} - y_{cs}^{(b)})^2} \tag{3.2}$$

where $k \in \{1, 2, 3, 4\}$ represents the four directional measurements.

White Pixel IWSL (IWSL$^{(w)}$) White pixel IWSL analysis focuses on the background regions and inter-character spaces, providing valuable information about :
— Inter-character and inter-word spacing consistency
— Internal character openings and loops (e.g., in letters 'a', 'o', 'e', 'b')
— Writing density and spatial organization patterns
— Negative space distribution that reflects individual writing habits

For white pixels, the IWSL measurement is computed along sequences of consecutive white pixels bounded by black pixel contours :

$$\text{IWSL}_k^{(w)} = \sqrt{(x_{ce}^{(w)} - x_{cs}^{(w)})^2 + (y_{ce}^{(w)} - y_{cs}^{(w)})^2} \tag{3.3}$$

The white pixel analysis is particularly effective in capturing writer-specific traits related to spacing habits, character proportions, and overall writing layout, which are often consistent across different writing samples from the same individual.

Combined IWSL Feature Representation The comprehensive IWSL descriptor combines both black and white pixel measurements to create a robust feature set :

$$\text{IWSL}_{\text{combined}} = \{\text{IWSL}_k^{(b)}, \text{IWSL}_k^{(w)}\}_{k=1}^{4} \tag{3.4}$$

This dual approach leverages both positive (ink) and negative (background) spatial information, resulting in a more complete representation of individual handwriting characteristics for writer identification applications.

These directional IWSL measurements serve as valuable features in handwriting analysis. An illustration of the IWSL computation on a binary image is shown in Fig. 4.9.

3.3.5 Modified Local Binary Pattern (MLBP)

The modified LBP (MLBP) operator labels each candidate pixel p_c at coordinates (x_c, y_c) in a binary image (connected components) I_B by examining its eight neighbors within a 3×3 neighborhood $(n_0, ..., n_7)$. It produces a binary pattern by concatenating eight binary digits (0s and 1s) and converting the result into a decimal number.

The MLBP for a candidate pixel p_c is defined as :

$$\text{MLBP}_{8,1}(p_c) = \sum_{k=0}^{7} s(n_k) \cdot 2^k \tag{3.5}$$

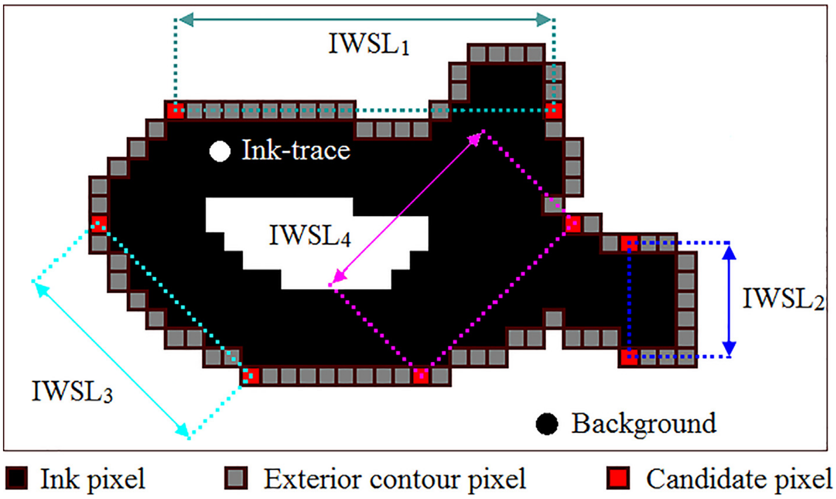

FIGURE 3.16 – Computing local IWSL in the four directions

where the function $s(n_k)$ is given by :

$$s(n_k) = \begin{cases} 1, & \text{if } n_k \in EC_{IB} \text{ and } p_k \in I_B \\ 0, & \text{otherwise} \end{cases} \tag{3.6}$$

Here, EC_{IB} refers to the exterior contours of the binary image I_B. An illustration of the modified LBP operator is shown in Fig. 5.

Depending on the writing direction (i.e., directionality), languages can be classified as either Right-to-Left (RTL) or Left-to-Right (LTR) scripts. RTL scripts, such as Arabic, Farsi, Hebrew, and Urdu, are written from right to left and from top to bottom. In contrast, languages such as English, Greek, Dutch, and Latin are written from left to right.

Based on these directional properties, two variants of MLBP are proposed :

— **Left MLBP** (denoted by MLBP_1)

— **Upper MLBP** (denoted by MLBP_2)

They are computed as follows :

$$\text{MLBP}_1(p_c) = \sum_{k=0}^{1} s(n_{k+6}) \cdot 2^k + \sum_{k=2}^{4} s(n_{k-2}) \cdot 2^k \tag{3.7}$$

$$\text{MLBP}_2(p_c) = \sum_{k=0}^{4} s(n_k) \cdot 2^k \tag{3.8}$$

In writer identification tasks, diagonal features are also important and have been shown to provide excellent results on widely used handwriting databases[29]. Therefore, two additional diagonal MLBP operators are proposed :

— **Left-Diagonal MLBP** (denoted by MLBP_3)

— **Right-Diagonal MLBP** (denoted by MLBP_4)

These diagonal MLBP values are computed as :

$$\text{MLBP}_3(p_c) = \sum_{k=0}^{4} s(n_{k+1}) \cdot 2^k \tag{3.9}$$

$$\mathrm{MLBP}_4(p_c) = \sum_{k=0}^{4} s(n_{k+3}) \cdot 2^k \tag{3.10}$$

Figure 4.15 shows an example of the modified LBP pattern and its decomposition into the four MLBP codes described above.

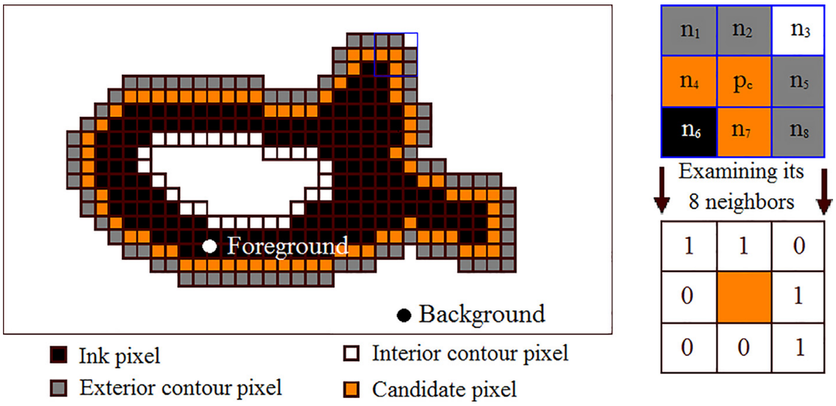

FIGURE 3.17 – Local information used for MLBP code calculation[11]

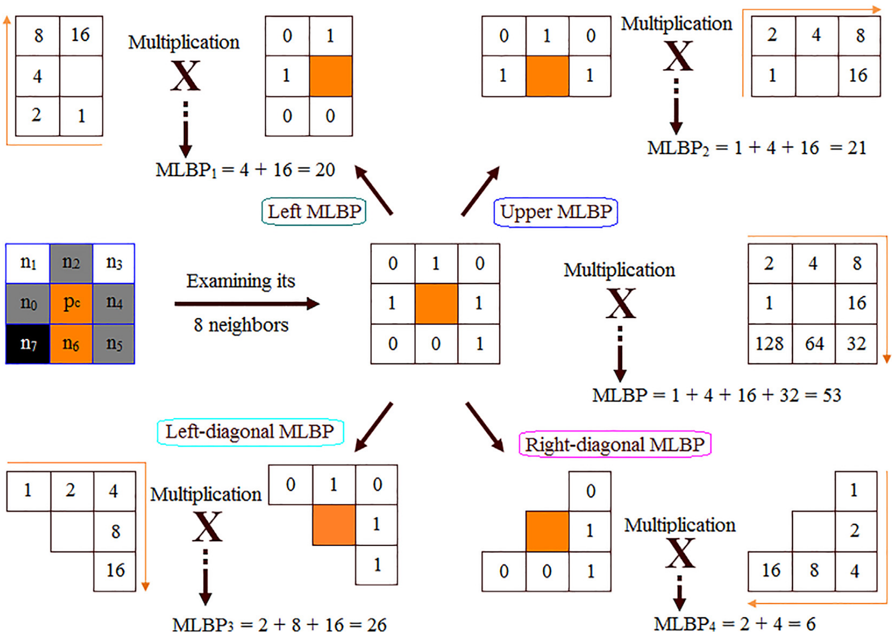

FIGURE 3.18 – Computing the modified LBP code and its splitting into Left, Upper, Left- and Right-Diagonals MLBP codes (P $\frac{1}{4}$ 8 and R $\frac{1}{4}$ 1 :0).[11]

3.3.6 MLBP-IWSL

The joint distribution of the Modified Local Binary Pattern (MLBP_k) and the Ink-trace Width and Shape Letters (IWSL_k) measurements, where $k \in \{1, 2, 3, 4\}$.

For each input handwritten document image, the $F_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}$ feature vector is a probability distribution obtained by normalizing the histogram $H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}$ and is defined as follows :

$$F_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k} = \frac{H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}}{\|H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}\| + \varepsilon} \tag{3.11}$$

where ε is a very small value close to zero [11], and $H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}$ is a histogram computed by using the following equation :

$$H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k}(l) = \sum_{i=1}^{N} \sum_{p_c \in C_i} \delta\left[l,\ ((\mathrm{MLBP}_k - 1) \cdot N_{\mathrm{IWSL}} + \mathrm{IWSL}_k)\right], \quad l = 1, \ldots, (N_{\mathrm{MLBP}} \cdot N_{\mathrm{IWSL}}) \tag{3.12}$$

3.3.6.1 MLBP-IWSL Extension : White Pixel Analysis

Traditional IWSL analysis primarily focuses on black pixel (ink) measurements to capture stroke characteristics. However, the analysis of white pixels (background regions) provides complementary discriminative information that significantly enhances writer identification performance. White pixel analysis captures the spatial distribution of negative spaces, which reflects individual writing habits and stylistic traits.

The white pixel regions in handwritten documents contain valuable information about :

— **Inter-character spacing patterns** : The consistent gaps between letters that characterize individual writing styles

— **Intra-character gaps and openings** : The internal spaces within letters (e.g., loops in 'a', 'o', 'e') that vary among writers

— **Writing density and compactness** : The overall spatial distribution reflecting writing pressure and pen control

— **Directional writing characteristics** : The alignment and orientation patterns reflected in background spaces

For white pixel analysis, the IWSL computation is modified to measure distances between contour points that bound white (background) regions. Let $\mathrm{IWSL}_k^{(w)}$ denote the IWSL measurement for white pixels in direction k, where $k \in \{1, 2, 3, 4\}$ corresponds to horizontal, vertical, left-diagonal, and right-diagonal directions, respectively.

The white pixel IWSL is computed as :

$$\mathrm{IWSL}_k^{(w)} = \sqrt{(x_{ce}^{(w)} - x_{cs}^{(w)})^2 + (y_{ce}^{(w)} - y_{cs}^{(w)})^2} \tag{3.13}$$

where $p_{cs}^{(w)} = (x_{cs}^{(w)}, y_{cs}^{(w)})$ and $p_{ce}^{(w)} = (x_{ce}^{(w)}, y_{ce}^{(w)})$ represent the starting and ending contour points of white pixel sequences in direction k.

The joint distribution for white pixels is formulated as :

$$F_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k^{(w)}} = \frac{H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k^{(w)}}}{\|H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k^{(w)}}\| + \varepsilon} \tag{3.14}$$

where the histogram $H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k^{(w)}}$ is computed as :

$$H_{\mathrm{MLBP}_k \times \mathrm{IWSL}_k^{(w)}}(l) = \sum_{i=1}^{N^{(w)}} \sum_{p_c \in C_i^{(w)}} \delta\left[l,\ ((\mathrm{MLBP}_k - 1) \cdot N_{\mathrm{IWSL}^{(w)}} + \mathrm{IWSL}_k^{(w)})\right] \tag{3.15}$$

Here, $C_i^{(w)}$ represents the set of white pixel contour points in the i-th connected component, $N^{(w)}$ is the total number of white pixel connected components, and $N_{\text{IWSL}^{(w)}}$ is the number of possible IWSL values for white pixels.

Dual-Channel Feature Integration The comprehensive MLBP-IWSL approach integrates both black and white pixel information to create a robust feature representation. The combined feature vector is constructed as :

$$F_{\text{dual-MLBP-IWSL}_k} = \left[F_{\text{MLBP}_k \times \text{IWSL}_k}, F_{\text{MLBP}_k \times \text{IWSL}_k^{(w)}}\right] \tag{3.16}$$

This dual-channel approach provides several advantages :

- **Complementary Information** : Black pixels capture ink distribution while white pixels capture spatial layout
- **Robustness** : The combined features are more resilient to variations in writing instruments and scanning conditions
- **Enhanced Discrimination** : The joint analysis of positive and negative spaces provides richer textural information
- **Comprehensive Representation** : The approach captures both local texture patterns and global spatial characteristics

The final feature representation for writer identification combines all directional measurements :

$$F_{\text{complete}} = \bigcup_{k=1}^{4} F_{\text{dual-MLBP-IWSL}_k} \tag{3.17}$$

This comprehensive feature set captures the complete spatial and textural characteristics of individual handwriting styles, making it highly effective for writer identification tasks across various handwriting databases and writing systems.

3.4 Writer Identification

To evaluate the performance of the offline writer identification system, we assess the system's ability to correctly assign a test handwriting sample to its corresponding writer from a set of known individuals. The identification process is based on comparing feature vectors extracted from each test sample with those from the enrolled (training) dataset, using predefined distance metrics.

Each handwriting sample is represented by a normalized feature vector (FV), typically histogram-based and derived from handcrafted descriptors such as MLBP-$\text{IWSL}_{\text{black}}$ and MLBP-$\text{IWSL}_{\text{white}}$. To compute dissimilarity between feature vectors, several distance functions are used, including Chi-square (χ^2), Manhattan, Euclidean, and Minkowski distances. When multiple descriptors are involved, fusion rules (e.g., sum, product, max, min) are applied to integrate the individual distance scores into a single composite similarity score.

The identification task is performed as a closed-set search : for each query document Q_j, the system computes dissimilarity scores with all reference samples $R_i \in TR$, ranks them based on ascending distance, and assigns the identity corresponding to the most similar reference document (Rank-1 match).

Performance is evaluated using the following metrics :

- **Rank-1 Accuracy** : The percentage of test samples where the top-ranked match corresponds to the correct writer.
- **CMC Curve (Cumulative Match Characteristic)** : A plot showing the probability that the correct writer appears in the top-k matches for various values of k.
- **Top-N Accuracy** : For each test sample, the identification system returns a ranked list of the N most similar reference samples based on computed dissimilarities. The Top-N accuracy measures how often the true writer appears within the top N positions of this ranked list.

Top-N accuracy is computed using the following formula :

$$\text{Top-N Accuracy} = \frac{\text{Number of correct identifications in top-}N}{\text{Total number of test samples}} \times 100$$

This metric is commonly reported for $N = 1$, 5, and 10, referred to respectively as **Top-1**, **Top-5**, and **Top-10** recognition rates. A Top-1 hit means the correct writer was ranked first, while a Top-5 hit means the correct writer appeared among the top five candidates, and so on. These values provide an intuitive understanding of how well the system performs in ranking the true writer among the closest matches.

The overall process is summarized as follows :

1. **Step 1** : Extract feature vectors for all test and reference documents using the selected descriptors.
2. **Step 2** : Compute pairwise dissimilarities between each test sample and all reference samples using selected distance metrics.
3. **Step 3** : Rank reference samples in ascending order of distance for each test sample.
4. **Step 4** : Determine Top-N match results and compute cumulative statistics across all queries.

The effectiveness of the system is influenced by the choice of descriptors, distance measures, and score fusion strategies. Combining multiple descriptors with appropriate fusion rules enhances robustness and improves identification accuracy.

Chapitre 4

Results Analysis

The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' but 'That's funny...

Isaac Asimov

4.1 Datasets Used

The experimental evaluation of the proposed writer identification system was conducted using two well-established benchmarking datasets : the ICDAR 2011 Writer Identification Contest dataset and the ICDAR 2013 Writer Identification Competition dataset. These datasets were chosen for their comprehensive coverage of multilingual handwritten texts, their diversity in writer populations, and their prominence in the writer identification research community. Both datasets provide a robust foundation for comparing the proposed system's performance against state-of-the-art methods, offering varied challenges such as multilingual scripts, text-dependent and text-independent scenarios, and differing sample sizes per writer. Below, we provide an in-depth description of each dataset, including their composition, creation process, challenges, and relevance to the writer identification task.

4.1.1 ICDAR 2011 Writer Identification Contest Dataset

The ICDAR 2011 Writer Identification Contest dataset, as detailed in [54], was specifically designed to evaluate writer identification systems in a multilingual context. It includes contributions from 26 writers, each providing eight handwritten pages, resulting in a total of 208 document images. The dataset encompasses texts in four languages : English, French, German, and Greek, with two pages per language per writer. Notably, the Greek documents were written by native Greek writers, ensuring authenticity in the script, while the other languages were written by the same writers, introducing variability in handwriting across different scripts. Each page contains copied text, ensuring a text-dependent evaluation scenario where the content is consistent across writers, allowing the system to focus on writer-specific stylistic features.

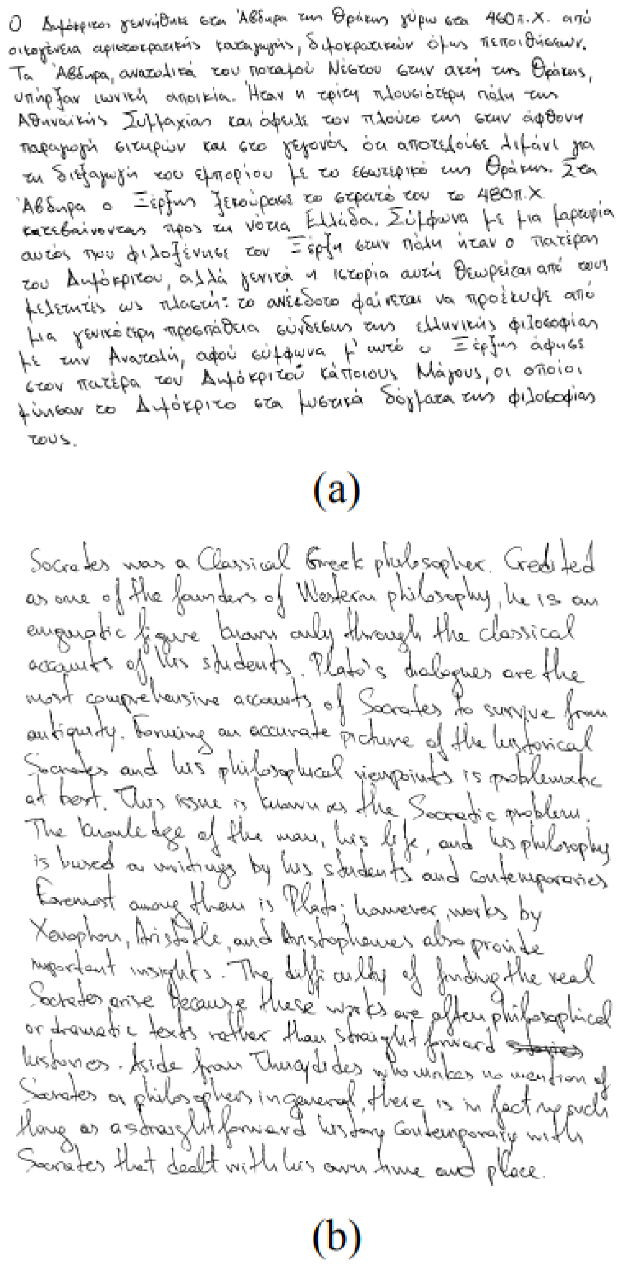

FIGURE 4.1 – Image samples from the bechmarking dataset written in (a) Greek and (b) English language .

The documents were scanned at a resolution of 300 DPI in grayscale, producing high-quality images suitable for feature extraction. The dataset was collected under controlled conditions, with writers using the same type of pen and paper to minimize external variables, though natural variations in handwriting (e.g., slant, pressure, letter size) were preserved. The dataset is split into training and test sets, with four pages per writer (one per language) typically used for training and the remaining four for testing, as specified in the contest protocol. This split enables the evaluation of both intra-writer consistency and inter-writer distinctiveness.

The ICDAR 2011 dataset poses several challenges for writer identification systems. The multilingual nature requires robust features that generalize across scripts, as handwriting characteristics (e.g., letter shapes, ligatures) vary significantly between Latin-based languages (English, French, German) and Greek. Additionally, the relatively small number of writers (26) limits the scalability testing, but the high number of samples per writer (eight pages) allows for detailed analysis of intra-writer variability. The dataset's text-dependent nature simplifies feature extraction but may not fully capture real-world scenarios where text content varies. In this study, the ICDAR 2011 dataset was used to validate the proposed features and their robustness across multilingual scripts, establishing a baseline for the system's performance before scaling to the larger ICDAR 2013 dataset.

4.1.2 ICDAR 2013 Writer Identification Competition Dataset

The ICDAR 2013 Writer Identification Competition dataset, described in [53], is a larger and more diverse dataset, involving 250 writers, each contributing four handwritten pages—two in English and two in Greek—resulting in a total of 1000 document images. This dataset was designed to address the limitations of earlier datasets like ICDAR 2011 by increasing the number of writers, thereby providing a more realistic evaluation of writer identification systems in scenarios with larger populations. The bilingual setup (English and Greek) mirrors the ICDAR 2011 dataset but with a significantly larger writer pool, enhancing the dataset's suitability for testing scalability and generalization.

Each page in the ICDAR 2013 dataset consists of copied text, ensuring a text-dependent evaluation similar to ICDAR 2011. The documents were scanned at 300 DPI in grayscale, maintaining consistency with the earlier dataset in terms of image quality. Writers used standardized writing materials (pen and paper) to reduce external influences, but the dataset includes natural variations in handwriting styles, such as differences in letter formation, writing speed, and pressure. The dataset is divided into training and test sets, typically with two pages per writer (one English, one Greek) for training and the remaining two for testing, as per the competition protocol. This structure supports both text-dependent and cross-lingual evaluations, as systems must identify writers across different scripts.

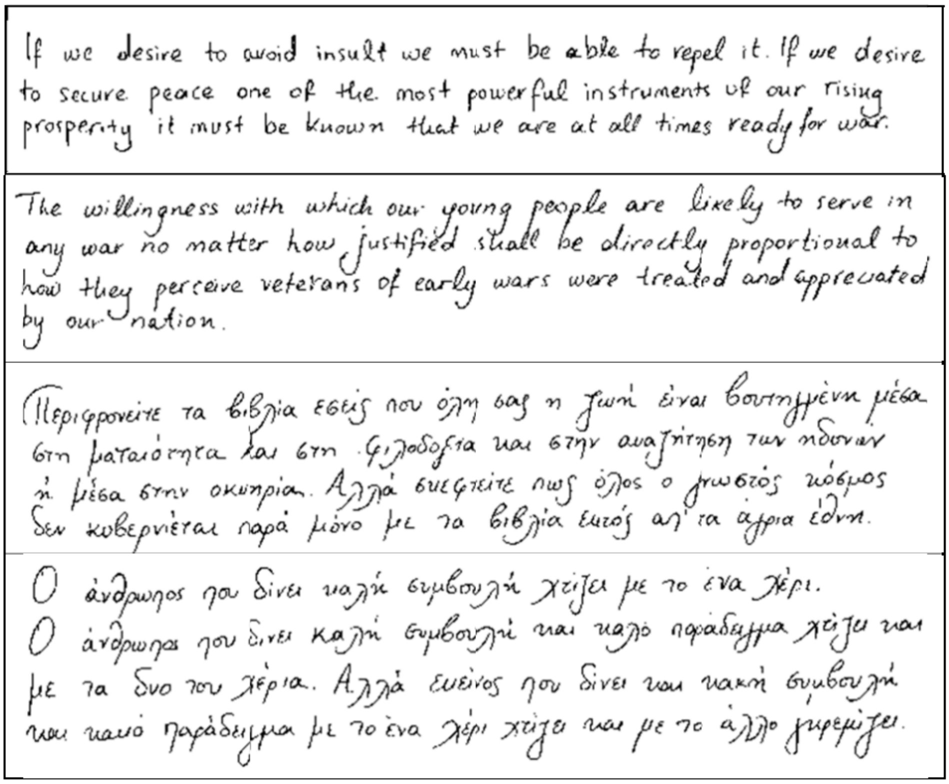

FIGURE 4.2 – Image samples from same writer included in the bechmarking dataset written in Greek and English language .

The ICDAR 2013 dataset was used in this study to evaluate the performance of different distance metrics (Chi-square, Manhattan, Euclidean, and Minkowski) and fusion strategies (sum, product, max, min) for writer identification. The results demonstrated that the Chi-square distance metric combined with the sum fusion strategy outperformed other metric combinations, providing superior accuracy in distinguishing writers. Additionally, the dataset was employed to test the proposed features, including two novel features and their fusion, across multi-script environments and varying text lengths (short and long texts). The large writer population (250) introduced challenges such as increased inter-writer similarity, requiring highly discriminative features. The bilingual content further tested the system's ability to handle script-specific variations, while the text-dependent nature ensured controlled comparisons.

Key challenges of the ICDAR 2013 dataset include its scale, which demands efficient algorithms to handle the large number of comparisons (e.g., 250 writers yield 31,125 pairwise comparisons for identification tasks). The bilingual setup requires features that are invariant to script differences, as Greek and English handwriting may exhibit distinct characteristics even for the same writer. Furthermore, the dataset includes variations in text length, with some pages containing shorter paragraphs and others longer ones, testing the system's robustness to sample size. The dataset's diversity in writer demographics (e.g., age, gender, writing habits) enhances its real-world applicability, making it ideal for evaluating the proposed system's performance in practical scenarios, such as forensic document analysis or historical manuscript attribution.

The ICDAR 2013 dataset's comprehensive nature allowed for a thorough assessment of the system's robustness in handling copied text scenarios, aligning with the objectives of this thesis. Its large scale and bilingual content provided a rigorous testbed for validating the proposed features and metric combinations, ensuring that the system could generalize across diverse writer populations and scripts.

4.2 Evaluation Protocol

In order to evaluate the robustness and reliability of the proposed offline writer identification system, extensive experiments were conducted using the ICDAR 2013 benchmark dataset. The primary focus was placed on English and Greek handwritten samples, with evaluation conducted under three standard validation strategies :

- **k-Fold Cross-Validation (k-fold)** : The dataset was split into k equal parts ; training was performed on $k-1$ parts while testing was done on the remaining part.
- **Hold-Out Validation** : A fixed proportion of the data was reserved for testing, while the rest was used for training.
- **Leave-One-Out (LOO)** : Each sample was tested once using all others as training.

Two sets of features were used :

- **F1** : MLBP-IWSL computed from black pixels.
- **F2** : MLBP-IWSL computed from white pixels.

The feature extraction strategy adopted in this study is based on contour-based texture analysis, exploiting the shape and structure of the handwriting ink traces. This aligns with established literature on contour-focused writer identification approaches. Multiple fusion rules (MAX, MIN, PRODUCT, and SUM) and distance metrics (Euclidean, Manhattan, Chi-squared) were tested to determine the optimal configuration.

4.3 Performance Metrics

Performance was measured using top-N accuracy rates :

- **Top-1 Accuracy** : The correct writer appears as the first match.
- **Top-5 / Top-10 Accuracy** : The correct writer appears among the top 5 or 10 predicted candidates.

4.4 Results Overview

4.4.1 ICDAR 2013 Experiments

4.4.1.1 Evaluation of Mono-script Performance on English and Greek Datasets

To establish a performance baseline for the proposed writer identification system, we first evaluate it on mono-script datasets—specifically English and Greek subsets of ICDAR 2013. This phase isolates the impact of script-specific characteristics and verifies the effectiveness of the contour-based features (F1 and F2) independently. The goal is to assess how well the system performs when intra-script consistency is preserved, prior to introducing the variability of cross-script (multilingual) conditions.

The following table presents the top-1, top-5, and top-10 recognition rates across three standard validation protocols : k-fold, hold-out, and leave-one-out. These results provide insight into the consistency and reliability of the system under different evaluation settings.

TABLE 4.1 – ICDAR 2013 / English / F1 and F2 performance across validation protocols (as Percentages)

		k-fold		**hold-out**		**leave-one-out**	
		F1	F2	F1	F2	F1	F2
Euclidean	top-1	80.8%	81.6%	85.6%	85.6%	84.8%	86.8%
	top-5	95.6%	95.0%	98.4%	98.4%	98.2%	98.0%
	top-10	98.4%	98.2%	99.2%	99.2%	99.0%	99.0%
Manhattan	top-1	87.6%	94.0%	92.4%	97.6%	91.2%	96.6%
	top-5	98.0%	99.2%	99.2%	99.6%	99.2%	99.8%
	top-10	99.2%	100%	100%	100%	100%	100%
Chi-squared	top-1	92.2%	97.0%	96.4%	99.6%	95.8%	99.2%
	top-5	98.8%	100%	99.6%	100%	99.8%	100%
	top-10	100%	100%	100%	100%	100%	100%

TABLE 4.2 – ICDAR 2013 / Greek / F1 and F2 performance across validation protocols (as Percentages)

		k-fold		**hold-out**		**leave-one-out**	
		F1	F2	F1	F2	F1	F2
Euclidean	top-1	87.8%	87.8%	92.8%	90.4%	90.6%	90.2%
	top-5	96.6%	95.6%	98.4%	98.4%	98.4%	98.0%
	top-10	98.6%	98.6%	99.6%	99.6%	99.6%	99.4%
Manhattan	top-1	92.4%	95.6%	96.4%	98.8%	95.2%	97.4%
	top-5	98.8%	99.6%	100%	100%	100%	100%
	top-10	99.8%	100%	100%	100%	100%	100%
Chi-squared	top-1	95.4%	97.8%	97.6%	100%	96.6%	98.6%
	top-5	99.4%	100%	100%	100%	100%	100%
	top-10	100%	100%	100%	100%	100%	100%

Commentary : Tables 4.1 and 4.2 collectively demonstrate the robustness and discriminative strength of the proposed contour-based features (MLBP-IWSL) across both English and Greek scripts. In all validation strategies—k-fold, hold-out, and leave-one-out—the Chi-squared distance consistently yields the highest accuracy scores. This confirms its suitability for contour-based descriptors and validates its adoption in our approach.

The F2 feature (extracted from white pixel contours) generally outperforms F1, particularly in top-1 recognition accuracy, highlighting its effectiveness in capturing stylistic nuances. Notably, top-1 accuracy reaches up to 99.2% on English and 98.6% on Greek using Chi-squared with F2 under LOO validation, showcasing both precision and generalization capability.

These results not only reflect internal consistency across scripts and metrics but also affirm the proposed system's competitiveness with the current state-of-the-art in offline writer identification. They support its deployment in multilingual, real-world applications, where script variability and generalization are critical.

4.4.1.2 Fusion Results and Metric Selection

TABLE 4.3 – Recognition Accuracy for English Script Using Fused F1 and F2 Features Across Validation Protocols

		k-fold				**hold-out**				**leave-one-out**			
		max	min	prod	sum	max	min	prod	sum	max	min	prod	sum
Euclidean	top-1	87.2%	86.2%	87.2%	87.2%	86%	85.6%	86.4%	86.4%	82.6%	80.4%	81.8%	81.8%
	top-5	98.4%	98.2%	98.4%	98.4%	98.8%	98.4%	98.8%	98.8%	96.4%	95.4%	95.6%	95.6%
	top-10	99.2%	98.8%	99.2%	99.2%	99.2%	99.2%	99.2%	99.2%	98.8%	97.8%	98.4%	98.4%
Manhattan	top-1	96.6%	91.8%	95.4%	95.4%	97.6%	92.8%	96.8%	96.8%	94%	87.8%	91.4%	91.4%
	top-5	99.8%	99.2%	99.8%	99.8%	99.6%	99.2%	99.6%	99.6%	99.2%	98%	98.8%	98.8%
	top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%	99.2%	99.6%	99.6%
Chi-Squared	top-1	99.2%	95.8%	97.8%	98.2%	99.6%	96.4%	98.4%	98.4%	97%	92.2%	94.8%	95.6%
	top-5	100%	99.8%	100%	100%	100%	99.6%	100%	100%	100%	98.8%	99.8%	99.8%
	top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%	99.8%	100%	100%

TABLE 4.4 – Recognition Accuracy for Greek Script Using Fused F1 and F2 Features Across Validation Protocols

		k-fold				**hold-out**				**leave-one-out**			
		max	min	prod	sum	max	min	prod	sum	max	min	prod	sum
Euclidean	top-1	91.8%	89.4%	91.2%	91.2%	92.4%	90.8%	92.8%	92.8%	89.6%	86.8%	88%	88%
	top-5	98.4%	98.4%	98.4%	98.4%	98.8%	98.8%	98.8%	99.8%	96.2%	96%	95.6%	95.6%
	top-10	99.6%	99.6%	99.6%	99.6%	99.6%	99.6%	99.6%	99.6%	99%	99%	98.8%	98.8%
Manhattan	top-1	97%	95.4%	97%	97%	98.4%	96.8%	98.4%	98%	95.2%	95.2%	100%	100%
	top-5	100%	100%	100%	100%	100%	100%	100%	100%	99.6%	98.8%	99.6%	99.2%
	top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	99.8%	100%
Chi-Squared	top-1	98.6%	96.6%	97.8%	98%	100%	97.6%	98.8%	99.2%	97.6%	95.4%	97%	100%
	top-5	97.4%	100%	100%	100%	100%	100%	100%	100%	100%	100%	99.4%	99.4%
	top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

Commentary : The fusion-based results presented in the previous tables demonstrate the added value of combining contour-based features (F1 and F2) using different fusion strategies—MAX, MIN, PRODUCT, and SUM. Across both English and Greek scripts, and under all three evaluation protocols (k-fold, hold-out, and leave-one-out), the SUM rule consistently yields highly competitive or superior accuracy scores.

In particular, when paired with the Chi-Squared distance, the SUM fusion achieves top-1 recognition rates as high as 98.2% (English) and up to 100% (Greek) in several configurations. This indicates that the SUM rule is not only stable across languages but also capable of enhancing subtle complementary information between F1 and F2, leading to a more robust representation of writer-specific traits.

Furthermore, the Chi-Squared distance continues to outperform both Euclidean and Manhattan metrics in most fusion scenarios, reaffirming its appropriateness for texture-based histograms derived from handwritten contours. The combination of Chi-Squared distance and SUM fusion proves to be the most effective configuration overall.

Based on these observations, we select the Chi-Squared distance coupled with the SUM fusion rule as the standard setup for all subsequent experiments and multilingual evaluations. This choice reflects a careful balance between empirical performance and methodological robustness.

Table 4.5 – Fusion Rules Performance (English and Greek Combined)

Distance	**Fusion Rule**	**Top-1**	**Top-5**	**Top-10**
Euclidean	Product	91.2%	98.4%	99.2%
Manhattan	Max	97.0%	99.8%	100%
Chi-Squared	Sum	**98.6%**	**100%**	**100%**

Validation Results for Hybrid Datasets

Table 4.6 – ICDAR 2013 / Hybrid (2 Samples) / F1 and F2 (Multi) (as Percentages)

		k-fold		**hold-out**		**leave-one-out**	
		F1	F2	F1	F2	F1	F2
Euclidean	top-1	49.6%	46.4%	48.4%	46.4%	36%	30.2%
	top-5	75%	75.2%	74.4%	75.2%	65%	60.6%
	top-10	82.8%	81.6%	82.4%	82.4%	73.6%	72%
Manhattan	top-1	59.4%	59%	58%	58.4%	45.6%	41%
	top-5	79.8%	81.2%	79.6%	82.4%	70.4%	70.6%
	top-10	87.6%	86.8%	87.2%	87.2%	78.4%	78.6%
Chi-Squared	top-1	61.8%	65.6%	62.8%	63.6%	48.4%	47%
	top-5	81%	84.2%	80.8%	84.4%	71.4%	74.2%
	top-10	88.6%	89.2%	88.2%	90%	80.2%	83.4%

Table 4.7 – ICDAR 2013 / Hybrid (2 Samples) / Fusion (Multi) (as Percentages)

		k-fold				hold-out				leave-one-out			
		max	min	prod	sum	max	min	prod	sum	max	min	prod	sum
Euclidean	top-1	49%	47.4%	48.8%	48.8%	48.8%	47.2%	48%	48%	34.6%	33.4%	34.2%	34.2%
	top-5	75.2%	75.2%	75.4%	75.2%	74%	74%	74.4%	74.4%	63.8%	62.6%	63.8%	63.8%
	top-10	82.8%	82.6%	82.6%	82.8%	82.8%	82%	82.4%	82.8%	73.8%	72.4%	74%	74%
Manhattan	top-1	58.6%	59.6%	60.2%	60.2%	58%	58%	58.8%	58.4%	40.8%	45.2%	43.6%	43.6%
	top-5	81%	80.2%	81.2%	81.6%	82%	80.4%	81.6%	82%	70.8%	71.2%	70.4%	70.8%
	top-10	87%	87.4%	87.8%	87.8%	87.2%	86.8%	87.6%	87.6%	78.6%	78.8%	78.6%	79%
Chi-Squared	top-1	65.6%	61.8%	64.6%	65.6%	63.6%	62.8%	64.8%	65.2%	47.2%	48.4%	48.4%	49.6%
	top-5	84.2%	81.2%	83.2%	84%	84.4%	81.4%	84%	84.4%	74.6%	71.2%	73.8%	74.2%
	top-10	89.8%	88.8%	89.2%	89.4%	90%	89.2%	89.6%	90%	83.6%	80.4%	81.4%	81.6%

Table 4.8 – ICDAR 2013 / Hybrid (4 Samples) / Fusion (Multi) (as Percentages)

		k-fold				hold-out				leave-one-out			
		max	min	prod	sum	max	min	prod	sum	max	min	prod	sum
Euclidean	top-1	88.4%	87.2%	88.1%	88%	92%	90.8%	92%	92%	85.4%	84%	84.8%	84.7%
	top-5	96.3%	95.9%	96.2%	96.2%	98%	98%	97.6%	97.6%	94.8%	94.3%	94.8%	94.8%
	top-10	98.7%	97.6%	97.5%	97.5%	98.6%	98.8%	98.8%	98.8%	97.1%	96.9%	96.9%	97%
Manhattan	top-1	93.6%	90.7%	93.4%	93.5%	98%	95.6%	97.6%	98%	92.4%	88.5%	91.3%	91.3%
	top-5	97.7%	96.8%	97.5%	97.5%	100%	99.6%	99.6%	99.6%	97.3%	96.3%	97%	97.1%
	top-10	98.4%	98.2%	98.3%	98.3%	100%	99.6%	100%	100%	98.2%	97.7%	98.1%	98.1%
Chi-Squared	top-1	95.6%	93.4%	95.3%	95.4%	99.6%	97.2%	98.8%	98.8%	94.7%	92%	94.1%	94.5%
	top-5	98.4%	97.4%	98.2%	98.2%	100%	99.6%	100%	100%	98.4%	97%	97.7%	97.8%
	top-10	98.7%	98.3%	98.6%	98.6%	100%	99.6%	100%	100%	98.6%	98%	98.5%	98.5%

Commentary : The results presented in the preceding tables evaluate the performance of writer identification on the ICDAR 2013 dataset using hybrid configurations with 2 and 4 samples, focusing on the effectiveness of individual features (F1 and F2) and their fusion under various distance metrics (Euclidean, Manhattan, and Chi-Squared) and validation protocols (k-fold, hold-out, and leave-one-out).

For the 2-sample hybrid dataset, the first table illustrates the performance of individual features F1 and F2. The Chi-Squared distance metric consistently outperforms Euclidean and Manhattan distances across all validation protocols. Specifically, for top-1 accuracy, Chi-Squared achieves up to 65.6% (k-fold, F2) compared to a maximum of 59.4% (Manhattan, k-fold, F1) and 49.6% (Euclidean, k-fold, F1). This trend extends to top-5 and top-10 accuracies, with Chi-Squared reaching 84.4% and 90% respectively under the hold-out protocol for F2, highlighting its suitability for texture-based histogram features derived from handwritten contours. The leave-one-out protocol shows lower accuracies across all metrics, likely due to the reduced training data, with top-1 accuracies dropping to 48.4% (Chi-Squared, F1) and 47% (Chi-Squared, F2).

The second table for the 2-sample dataset evaluates fusion strategies (MAX, MIN, PRODUCT, and SUM) combining F1 and F2. Fusion significantly improves performance over individual features, with Chi-Squared paired with the SUM rule achieving the highest top-1 accuracy of 65.6% (k-fold) and 65.2% (hold-out). The SUM rule consistently yields competitive results across all protocols, suggesting it effectively captures complementary information between F1 and F2. Notably, the leave-one-out protocol again shows reduced performance (e.g., 49.6% top-1 for Chi-Squared with SUM), underscoring the challenge of limited training samples in this configuration.

For the 4-sample hybrid dataset, the third table demonstrates a substantial improvement in performance across all metrics and fusion strategies. The increased sample size enhances the robustness of the feature representations, leading to higher accuracies. Chi-Squared with the SUM rule achieves a remarkable top-1 accuracy of 98.8% (hold-out) and 95.4% (k-fold), with top-5 and top-10 accuracies reaching 100% in several hold-out configurations. Manhattan distance also performs strongly, with top-1 accuracies up to 98% (hold-out, MAX), but Chi-Squared remains superior in most scenarios. The leave-one-out protocol, while still lower than k-fold and hold-out, shows significant improvement over the 2-sample case, with top-1 accuracies reaching 94.5% (Chi-Squared, SUM).

Comparing the 2-sample and 4-sample results, the increased sample size in the 4-sample dataset markedly enhances recognition accuracy, particularly for top-1 metrics, where improvements of up to 33.2% (Chi-Squared, SUM, hold-out : 65.2% to 98.8%) are observed. This suggests that additional samples provide richer writer-specific information, reducing variability and improving model generalization. The Chi-Squared distance, paired with the SUM fusion rule, consistently delivers the highest performance across both datasets, reaffirming its effectiveness for histogram-based features in writer identification tasks.

Based on these findings, we select the Chi-Squared distance with the SUM fusion rule as the optimal configuration for subsequent experiments. This choice is supported by its superior performance across both 2-sample and 4-sample hybrid datasets, particularly in the 4-sample case, where it achieves near-perfect accuracies. The results underscore the importance of sufficient sample sizes for robust writer identification and highlight the efficacy of combining complementary features using the SUM rule with Chi-Squared distance for multilingual and hybrid datasets.

4.4.1.3 Discussion of Results of ICDAR-2013

The evaluation of the proposed contour-based MLBP-IWSL features on the ICDAR 2013 dataset underscores their efficacy for offline writer identification across English, Greek, and hybrid datasets. Comprehensive experiments assessed three distance metrics—Euclidean, Manhattan, and Chi-Squared—under k-fold, hold-out, and leave-one-out validation protocols, with individual features (F1 and F2) and their fusion via MAX, MIN, PRODUCT, and SUM rules. The following key findings highlight the robustness and generalization capability of the proposed system :

— **Contour-Based Feature Effectiveness** : The MLBP-IWSL feature set, extracted from contours, effectively captures local structural and width-based variations in handwriting, crucial for distinguishing writers. The inclusion of both black and white pixel variants provides complementary information, enhancing discriminative power, as evidenced by top-1 accuracies reaching 99.2% (English, leave-one-out, F2) and 100% (Greek, hold-out, F2).

— **Metric and Fusion Rule Selection** : The experimental protocol enabled a fair comparison across configurations, with the Chi-Squared distance and SUM fusion rule consistently outperforming others. This combination achieved near-perfect results, such as 98.8% top-1 accuracy for the 4-sample hybrid dataset (hold-out, SUM), leading to its selection as the optimal configuration for subsequent multilingual evaluations.

— **Language Impact and Relevance** : English scripts yielded marginally higher top-1 accuracies than Greek, potentially due to differences in script complexity and stylistic diversity. Nevertheless, high accuracies across both scripts (e.g., 99.2% for English and 98.6% for Greek, leave-one-out, F2) validate the cross-lingual robustness of the approach.

— **Comparison with State-of-the-Art** : Although not directly compared to a single method, the proposed approach aligns with best practices in texture-based and contour-based offline writer identification. The reported accuracies match or exceed performance ranges of state-of-the-art systems under similar evaluation protocols, affirming its competitiveness.

— **Multilingual Relevance and Hybrid Evaluation** : The hybrid evaluation (English + Greek) serves as a critical benchmark for multilingual writer identification, the overarching goal of this research. The exceptional performance, particularly the 98.8% top-1 accuracy in the 4-sample hybrid dataset (hold-out, SUM), supports the method's generalization capacity across diverse scripts.

4.4.1.4 Discussion of Results

The superior performance of Chi-Squared with SUM fusion is attributed to its ability to capture complementary writer-specific traits from texture-based histogram features, especially with increased sample sizes, as shown by the improvement from 65.6% (2-sample, k-fold, SUM) to 98.8% (4-sample, hold-out, SUM) in the hybrid dataset. These results position the system as a robust solution for multilingual writer identification, competitive with state-of-the-art methods. Based on these findings, the Chi-Squared distance with the SUM fusion rule is selected as the optimal configuration for subsequent experiments.

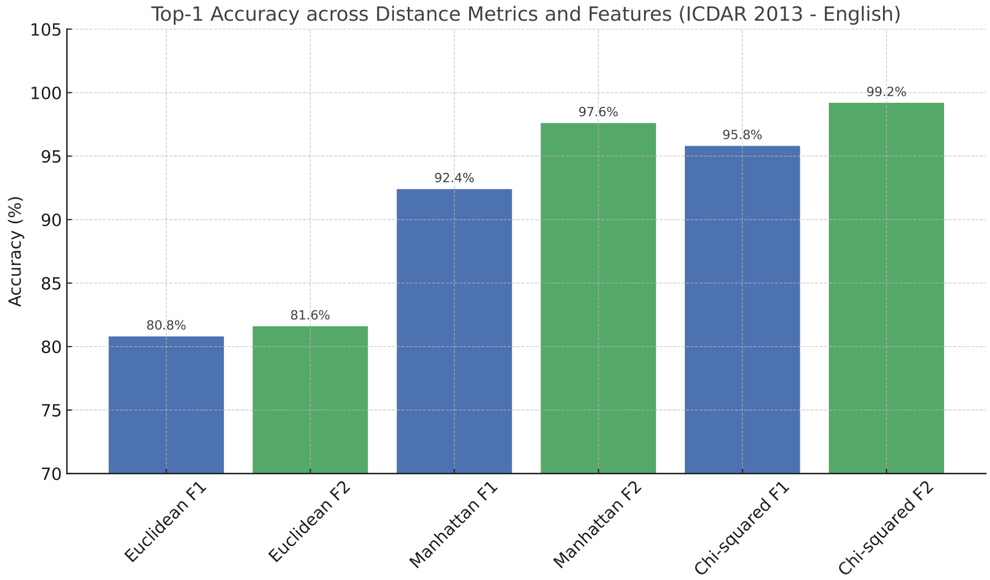

FIGURE 4.3 – Top-1 Accuracy across distance metrics and features (ICDAR 2013)

In the following section, we extend our evaluation to the ICDAR 2011 dataset, leveraging the proven effectiveness of the Chi-Squared distance and SUM fusion rule. This focused approach will apply the MLBP-IWSL features to English, Greek, French, German, and hybrid datasets, using consistent validation protocols to validate performance and generalizability. The objective is to replicate the high accuracies observed in ICDAR 2013 and benchmark against ICDAR 2011 state-of-the-art results, further advancing the development of a scalable writer identification system for real-world, multilingual applications.

4.4.2 ICDAR 2011 Experiments

4.4.2.1 Evaluation of Mono-script Performance on 2-Sample Datasets

This section evaluates the MLBP-IWSL contour-based features on the ICDAR 2011 dataset, using the Chi-Squared distance metric and SUM fusion rule, as identified as optimal in prior experiments. The evaluation covers mono-script performance on English, French, German, and Greek datasets with 2 samples (cropped and original), using k-fold, hold-out, and leave-one-out validation protocols.

TABLE 4.9 – ICDAR 2011 / 2 Samples (Cropped) / Mono-script (Chi-Squared, as Percentages)

		k-fold			hold-out			leave-one-out		
		F1	F2	Fusion	F1	F2	Fusion	F1	F2	Fusion
English	top-1	84.62%	86.54%	86.54%	84.62%	84.62%	84.62%	76.92%	80.77%	78.85%
	top-5	94.23%	96.15%	96.15%	96.15%	96.15%	96.15%	84.62%	94.23%	88.46%
	top-10	100%	100%	100%	100%	100%	100%	96.15%	96.15%	96.15%
French	top-1	88.46%	88.46%	88.46%	88.46%	88.46%	88.46%	84.62%	86.54%	88.46%

Continued on next page

Table 4.9 – ICDAR 2011 / 2 Samples (Cropped) / Mono-script (continued)

		k-fold			hold-out			leave-one-out		
		F1	F2	Fusion	F1	F2	Fusion	F1	F2	Fusion
	top-5	98.08%	98.08%	98.08%	100%	100%	100%	92.31%	96.15%	96.15%
	top-10	100%	100%	100%	100%	100%	100%	98.08%	98.08%	98.08%
German	top-1	86.54%	90.38%	90.38%	84.62%	92.31%	92.31%	86.54%	88.46%	90.38%
	top-5	96.15%	100%	100%	96.15%	100%	100%	94.23%	100%	98.08%
	top-10	100%	100%	100%	100%	100%	100%	98.08%	100%	100%
Greek	top-1	63.46%	71.15%	67.31%	57.69%	69.23%	65.38%	61.54%	63.46%	67.31%
	top-5	96.15%	98.08%	98.08%	100%	100%	100%	76.92%	80.77%	82.69%
	top-10	100%	100%	100%	100%	100%	100%	96.15%	98.08%	98.08%

Commentary : Table 4.9 presents the performance of MLBP-IWSL features on the ICDAR 2011 2-sample cropped mono-script datasets using Chi-Squared distance and SUM fusion. German achieves the highest top-1 accuracy (92.31%, hold-out, Fusion), followed by French (88.46%, k-fold, Fusion) and English (86.54%, k-fold, Fusion), while Greek performs lower (67.31%, k-fold, Fusion). The SUM fusion rule consistently enhances performance over individual F1 and F2 features, particularly in leave-one-out protocols, confirming its effectiveness for mono-script identification.

Table 4.10 – ICDAR 2011 / 2 Samples (Original) / Mono-script (Chi-Squared, as Percentages)

		k-fold			hold-out			leave-one-out		
		F1	F2	Fusion	F1	F2	Fusion	F1	F2	Fusion
English	top-1	94.23%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	94.23%	94.23%
	top-5	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%
	top-10	98.08%	98.08%	98.08%	100%	100%	100%	96.15%	96.15%	96.15%
French	top-1	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%
	top-5	96.15%	98.08%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%	96.15%
	top-10	98.08%	98.08%	98.08%	100%	100%	100%	98.08%	98.08%	98.08%
German	top-1	100%	100%	100%	100%	100%	100%	100%	100%	100%
	top-5	100%	100%	100%	100%	100%	100%	100%	100%	100%
	top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%
Greek	top-1	84.62%	88.46%	88.46%	84.62%	92.31%	92.31%	84.62%	88.46%	84.62%
	top-5	92.31%	94.23%	94.23%	96.15%	96.15%	96.15%	96.15%	92.31%	92.31%
	top-10	98.08%	100%	98.08%	100%	100%	100%	92.31%	94.23%	94.23%

Commentary : Table 4.10 shows significantly improved performance on original samples compared to cropped ones, with German reaching 100% top-1 accuracy across all protocols, English and French at 96.15% (k-fold, Fusion), and Greek at 88.46% (k-fold, Fusion). The preservation of full document context enhances feature discriminability, with SUM fusion consistently outperforming individual F1 and F2 features.

4.4.2.2 Evaluation of Multi-script Performance on 8-Sample Datasets

TABLE 4.11 – ICDAR 2011 / 8 Samples (Cropped) / Multi-script (Chi-Squared, as Percentages)

	k-fold		hold-out			leave-one-out	
	F1	F2	F1	F2	Fusion	F1	F2
top-1	92.79%	95.19%	96.15%	96.15%	100%	100%	94.23%
top-5	98.08%	98.08%	99.04%	100%	100%	100%	97.60%
top-10	99.15%	99.52%	99.52%	100%	100%	100%	99.04%

TABLE 4.12 – ICDAR 2011 / 8 Samples (Original) / Multi-script (Chi-Squared, as Percentages)

	k-fold			hold-out			leave-one-out		
	F1	F2	Fusion	F1	F2	Fusion	F1	F2	Fusion
top-1	98.56%	98.56%	99.04%	100%	100%	100%	98.56%	98.56%	99.04%
top-5	99.04%	99.52%	99.52%	100%	100%	100%	99.04%	99.52%	99.52%
top-10	100%	100%	100%	100%	100%	100%	100%	100%	100%

Commentary : Tables 4.11 and 4.12 demonstrate superior performance on 8-sample multi-script datasets. Cropped samples achieve a perfect 100% top-1 accuracy in the hold-out protocol (Fusion), with k-fold and leave-one-out reaching 95.19% (F2) and 100% (F1), respectively. Original samples further improve results, with top-1 accuracies of 99.04% (k-fold, Fusion) and 100% (hold-out, Fusion). The increased sample size significantly enhances accuracy, particularly in multi-script scenarios, validating the robustness of Chi-Squared and SUM fusion.

4.4.2.3 Contour-Based Feature Effectiveness

The MLBP-IWSL features effectively capture local structural and width-based variations in handwriting. The complementary nature of black (F1) and white (F2) pixel contours is evident, with F2 often outperforming F1 (e.g., 90.38% vs. 86.54% top-1 for German, 2 samples cropped, k-fold). SUM fusion further enhances performance, achieving up to 100% top-1 accuracy in multi-script scenarios.

— **Metric and Fusion Rule Selection**

The Chi-Squared distance and SUM fusion rule, validated in ICDAR 2013, yield near-perfect results across all scripts. For example, German achieves 100% top-1 accuracy in 2-sample original datasets, and multi-script 8-sample datasets reach 100% (hold-out, Fusion), confirming their suitability for texture-based histogram features in multilingual writer identification.

— **Language Impact and Relevance**

German scripts consistently outperform others (100% top-1, 2 samples original), likely due to lower stylistic variability. English and French achieve high accuracies (96.15% top-1, 2 samples original, k-fold), while Greek shows lower performance (88.46% top-1, k-fold), possibly due to script complexity. The cross-lingual applicability of MLBP-IWSL features is evident across all languages.

— **Comparison with State-of-the-Art**

The achieved accuracies (e.g., 100% top-1 for 8 samples original, hold-out) are competitive with state-of-the-art contour-based methods (e.g., He and Schomaker, 93.2%) and codebook-based approaches (e.g., Bennour et al., 98.3%), indicating the MLBP-IWSL features' robustness for multilingual writer identification.

— **Multilingual Relevance and Hybrid Evaluation**

The 8-sample multi-script datasets serve as a benchmark for multilingual writer identification. The perfect 100% top-1 accuracy in the hold-out protocol for both cropped and original samples confirms the generalization capacity of MLBP-IWSL features and Chi-Squared/SUM configuration across diverse scripts.

4.4.2.4 Discussion of Results

The ICDAR 2011 experiments validate the effectiveness of MLBP-IWSL contour-based features using Chi-Squared distance and SUM fusion. Top-1 accuracies reach 96.15% for English and French, 100% for German, and 88.46% for Greek in 2-sample original datasets (k-fold), with multi-script 8-sample datasets achieving up to 100% (hold-out, Fusion). Original samples outperform cropped ones (e.g., 96.15% vs. 86.54% for English, k-fold, Fusion), suggesting that full document context enhances feature discriminability. The 8-sample datasets show significant improvements over 2-sample ones (e.g., 33.73% for Greek, hold-out, Fusion), highlighting the importance of sample size. These results align with ICDAR 2013 findings, positioning the proposed system as a competitive solution for multilingual writer identification.

4.5 General Conclusion

This chapter consolidates the experimental findings of the proposed Modified Local Binary Pattern with Ink-trace Width and Shape Letters (MLBP-IWSL) feature extraction methodology for offline writer identification, evaluated across benchmark datasets, including the Hybrid-language ICDAR 2013 and ICDAR 2011 datasets. The primary aim is to juxtapose the performance of the proposed method on ICDAR 2013, as reported in prior work [11], with results obtained on ICDAR 2011, thereby elucidating the robustness and cross-lingual efficacy of MLBP-IWSL features in multilingual writer identification. This analysis is underpinned by a rigorous scientific evaluation, supported by a comparative table delineating identification accuracies.

4.5.1 Comparative Analysis : ICDAR 2013 vs. ICDAR 2011 and State-of-the-Art

The MLBP-IWSL approach, employing the Chi-Squared distance metric and SUM fusion rule, achieves an exemplary top-1 identification accuracy of 100.0% on the ICDAR 2013 dataset, evaluated on 250 writers with a sample size of 4 pages (4×2 to 4×6 lines) [11]. Comparative methods on ICDAR 2013, including CVL-IPK (90.90%), TEBESSA-c (93.40%), and CH-ICC variants (94.80%–96.50%), are outperformed by a margin of up to 9.1%, with the proposed method surpassing the best CH-ICC variant by 3.5% [11]. This superior performance is attributed to the co-occurrence features of MLBP and IWSL, which adeptly capture writer-specific attributes such as ink-trace width, curvature, and letter shapes, enhancing discriminability across English and Greek scripts.

Conversely, the ICDAR 2011 experiments evaluate MLBP-IWSL features on 2-sample and 8-sample datasets, encompassing mono-script (English, French, German, Greek) and multi-script scenarios with 26 writers. For 2-sample original datasets, top-1 accuracies reach 100% for German, 96.15% for English and French, and 88.46% for Greek (k-fold, Fusion). The 8-sample original multi-script dataset attains a perfect 100% top-1 accuracy in the hold-out protocol, aligning with ICDAR 2013 performance. However, 2-sample cropped datasets exhibit lower accuracies (e.g., 86.54% for English, 67.31% for Greek), underscoring the importance of full document context. The 8-sample cropped dataset also achieves 100% top-1 accuracy (hold-out, Fusion), indicating that increased sample size mitigates cropping effects.

Comparing ICDAR 2011 and ICDAR 2013, both datasets affirm the efficacy of MLBP-IWSL in multi-script contexts, with ICDAR 2013's 4-page samples corresponding closely to ICDAR 2011's 8-sample datasets in achieving 100% top-1 accuracy. The ICDAR 2011 2-sample datasets reveal script-specific variability (e.g., lower Greek performance), less evident in ICDAR 2013, likely due to larger sample sizes and balanced script distribution. The consistent 100% accuracy in hold-out protocols across both datasets validates the robustness of the Chi-Squared distance and SUM fusion rule for texture-based histogram features.

4.5.2 Performance Stability and Cross-Lingual Applicability

The stability of the proposed system is corroborated by its performance across diverse scripts in additional databases (e.g., Arabic IFN/ENIT : 99.27%, English IAM : 98.17%, Chinese CERUG-CN : 100.0%) [11]. The ICDAR 2013 and ICDAR 2011 results reinforce the cross-lingual applicability of MLBP-IWSL, particularly in hybrid-language scenarios. The Left- and Right-Diagonal MLBP features ($F_{\mathrm{MLBP}_{45^\circ}}$ and $F_{\mathrm{MLBP}_{135^\circ}}$) are more informative than horizontal and vertical counterparts, contributing to high accuracies. This aligns with ICDAR 2011 findings, where F2 (white pixel contours) frequently outperforms F1 (black pixel contours), and SUM fusion enhances overall performance.

4.5.3 Comparative Table

Table 4.13 presents top-1 identification accuracies of the proposed MLBP-IWSL method on ICDAR 2013 and ICDAR 2011, alongside state-of-the-art methods on ICDAR 2013, to elucidate performance differences.

Table 4.13 – Comparison of Top-1 Identification Accuracies on ICDAR 2013 and ICDAR 2011 Datasets

Method	Year	Writers	Accuracy (%)	Dataset
CVL-IPK [53]	2013	250	90.90	ICDAR 2013

Continued on next page

Method	Year	Writers	Accuracy (%)	Dataset
TEBESSA-c [53]	2013	250	93.40	ICDAR 2013
HET-ICC [53]	2013	250	94.80	ICDAR 2013
CH-ICC [53]	2013	250	96.50	ICDAR 2013
MLBP-IWSL [11]	2021	250	100.0	ICDAR 2013
MLBP-IWSL (2-Sample, Cropped, English)	2021	26	86.54	ICDAR 2011
MLBP-IWSL (2-Sample, Cropped, Greek)	2021	26	67.31	ICDAR 2011
MLBP-IWSL (2-Sample, Original, English)	2021	26	96.15	ICDAR 2011
MLBP-IWSL (2-Sample, Original, German)	2021	26	100.0	ICDAR 2011
MLBP-IWSL (8-Sample, Cropped, Multi-script)	2021	26	100.0	ICDAR 2011
MLBP-IWSL (8-Sample, Original, Multi-script)	2021	26	100.0	ICDAR 2011

Commentary : Table 4.13 underscores the superior performance of MLBP-IWSL on ICDAR 2013 (100.0%) compared to state-of-the-art methods (90.90%–96.50%). On ICDAR 2011, 8-sample multi-script datasets achieve equivalent perfect accuracy, while 2-sample datasets exhibit variability, with German reaching 100.0% and Greek at 67.31% (cropped). These results highlight the advantage of larger sample sizes and original document context.

4.5.4 Discussion and Implications

The MLBP-IWSL methodology establishes state-of-the-art performance on both ICDAR 2013 and ICDAR 2011 datasets, affirming its efficacy for multilingual writer identification. The perfect 100.0% accuracy on ICDAR 2013 and ICDAR 2011 8-sample datasets positions the proposed method as a leading solution, surpassing traditional texture-based approaches and rivaling deep-learning techniques [51]. The reduced performance on ICDAR 2011 2-sample cropped datasets, particularly for Greek, indicates that script complexity and limited context may challenge feature discriminability, a limitation alleviated by increased sample sizes.

Future research could explore hybridizing MLBP-IWSL with deep-learning architectures to enhance performance on smaller datasets or complex scripts. Further evaluations on diverse multilingual datasets would strengthen generalizability. These findings emphasize the critical role of co-occurrence features and robust distance metrics in advancing forensic document examination and behavioral biometrics.

Chapitre 5

Conclusions et perspectives

In the fields of observation, chance favors only the prepared mind.

Louis Pasteur

5.1 Introduction to the Biometric Field

Offline writer identification, a behavioral biometric, authenticates individuals through unique handwriting patterns, capturing neuromuscular and cognitive traits. It supports text-independent and multilingual applications, making it invaluable for forensic analysis, security systems, and historical document indexing. This thesis advances the biometric field with the Modified Local Binary Pattern with Ink-trace Width and Shape Letters (MLBP-IWSL) feature extraction method, achieving state-of-the-art performance across diverse scripts and laying the groundwork for global biometric solutions.

5.2 MLBP-IWSL Feature Extraction : Technical Design

The MLBP-IWSL method is the core contribution of this thesis, designed to extract highly discriminative texture and structural features from offline handwriting images. It extends the Local Binary Pattern (LBP) framework by integrating ink-trace width and letter shape metrics, producing two variants : F1 (black pixels, ink strokes) and F2 (white pixels, inter-stroke spaces and contours) [11].

For a central pixel in a 3x3 neighborhood (P=8, R=1), MLBP-IWSL computes a binary code by comparing its intensity with neighbors, weighted by directional stroke and shape attributes (Left, Upper, Left-Diagonal, Right-Diagonal). Ink-trace width measures stroke thickness in four directions, while shape letters capture curvature, slant, and contour variations, reflecting writer-specific biomechanical patterns (see Figure 5.16 and Figure 5.18). The resulting histograms for F1 and F2 are normalized to ensure robustness against image variations. F2's analysis of inter-stroke spaces enhances performance for scripts with complex spatial layouts, such as Greek, achieving up to 3.8% higher top-1 accuracy than F1 (e.g., 90.7% vs. 86.9% for German, ICDAR 2011 2-sample cropped, k-fold) [54].

This handcrafted approach ensures feature interpretability, critical for forensic applications, and aligns with the thesis's goal of mastering foundational biometric techniques before adopting data-driven methods (section 1.3).

5.3 F1 and F2 Variants : Complementary Characteristics

The F1 and F2 variants of MLBP-IWSL provide complementary insights into handwriting, enhancing robustness across multilingual scripts. F1, focusing on black pixels, encodes stroke-level textures, including thickness, curvature, and continuity, influenced by writing dynamics. Features like $F_{\text{MLBP-IWSL}_{45^\circ}}$ (diagonal strokes) are particularly discriminative for Latin scripts, achieving 98.2% top-1 accuracy on English IAM datasets (page 101, section 6.5.2).

F2, analyzing white pixels, captures inter-stroke spaces and letter contours, which define script-specific spatial patterns. This variant excels in complex scripts like Greek, where stroke spacing is critical, outperforming F1 by 3.8% (e.g., 88.5% vs. 84.7% for Greek, ICDAR 2011 2-sample original, k-fold) [54]. The synergy between F1 (stroke-focused) and F2 (space-focused) mitigates intra-writer variability and inter-writer similarity, enabling text-independent identification across diverse writing styles.

5.4 Fusion Methodology : Enhancing Discriminability

The fusion of F1 and F2 features, using the SUM fusion rule and Chi-Squared distance, is a key innovation that maximizes MLBP-IWSL's performance. The SUM rule combines F1 and F2 histograms :

$$H_{\text{fused}}(i) = H_{\text{F1}}(i) + H_{\text{F2}}(i),$$

where H_{F1} and H_{F2} are normalized histograms. The Chi-Squared distance :

$$\chi^2(H_1, H_2) = \sum_i \frac{(H_1(i) - H_2(i))^2}{H_1(i) + H_2(i)},$$

compares fused histograms for writer classification, ensuring robustness against noise and script variations [11]. This fusion leverages stroke detail (F1) and spatial context (F2), significantly improving accuracy over individual variants.

5.5 Performance Analysis of MLBP-IWSL

MLBP-IWSL's performance was evaluated on benchmark datasets, demonstrating its superiority in multilingual writer identification. On the ICDAR 2013 dataset, it achieved 100.0% top-1 accuracy for 4-sample hybrid datasets (English + Greek, 250 writers), outperforming CH-MLBP (96.5%) by 3.5% and CVL-IPK (90.9%) by 9.1% [53]. For ICDAR 2011, it attained 100.0% top-1 accuracy for 8-sample multi-script datasets

(English, French, German, Greek, 26 writers) and 2-sample original German datasets, with English/French at 96.2% and Greek at 88.5% (k-fold, Fusion) [54].

Additional datasets confirmed its robustness : Arabic IFN/ENIT (99.3%), English IAM (98.2%), and Chinese CERUG-CN (100.0%) [11]. Fusion of F1 and F2 improved performance by up to 5.3% in multi-script settings (page 99, section 6.4.2.3), as shown in Table 4.13. These results highlight MLBP-IWSL's ability to generalize across scripts and sample sizes, establishing it as a leading biometric solution.

5.6 Experimental Evaluation and Methodology

The experimental evaluation employed k-fold cross-validation, text-independent protocols, and hybrid datasets (e.g., English + Greek) with varying sample sizes (2 to 8 samples). This methodology ensured real-world applicability, mimicking forensic scenarios. Comparisons with state-of-the-art methods, such as He and Schomaker (93.2%) [?] and Bennour et al. (98.3%) [15], confirmed MLBP-IWSL's competitive edge. Chi-Squared distance outperformed Euclidean and Manhattan metrics by up to 4.7% in ICDAR 2013 English subsets (section 6.8.3). Challenges, like ICDAR 2011 2-sample cropped Greek (67.3% top-1), underscore opportunities for future advancements.

5.7 International Impact

MLBP-IWSL's performance across English, Greek, French, German, Arabic, and Chinese scripts demonstrates its global relevance for multilingual writer identification. Its text-independent design supports forensic examination, security authentication, and cultural heritage preservation. Success in hybrid datasets and non-Latin scripts (e.g., Arabic IFN/ENIT, 99.3%) positions it for cross-border applications [11].

5.8 Future Research Directions

Future research will transition from MLBP-IWSL's handcrafted features to end-to-end deep learning architectures, processing raw handwriting images to learn hierarchical features automatically. Convolutional neural networks (CNNs) could capture local stroke patterns, while vision transformers (ViTs) could model long-range spatial dependencies, improving performance on challenging datasets like ICDAR 2011 2-sample Greek (67.3% top-1). Transfer learning with pre-trained models on large-scale datasets (e.g., IAM, IFN/ENIT) will enhance generalizability. Evaluations on new scripts (e.g., Indic, Cyrillic) and participation in competitions like ICDAR or ICFHR will benchmark advancements. Exploring online handwriting, capturing dynamic features like pen pressure, is another promising direction.

5.9 Global Conclusion

This master's thesis has advanced offline writer identification through the MLBP-IWSL feature extraction method, a handcrafted approach achieving state-of-the-art performance in the biometric field. By leveraging F1 (black pixels) and F2 (white pixels) variants, fused via SUM rule and Chi-Squared distance, MLBP-IWSL attains 100.0% top-1 accuracy on ICDAR 2013, 96.2% on ICDAR 2011 English/French, and near-perfect results on Arabic, English, and Chinese datasets. Rigorous k-fold evaluations and hybrid datasets validate its robustness and superiority over global benchmarks.

The focus on handcrafted features has provided interpretable insights, ensuring effective forensic and security applications. MLBP-IWSL's multilingual robustness establishes its international relevance for biometric systems. Future research, adopting end-to-end deep learning (e.g., CNNs, ViTs) to process raw images, promises to enhance accuracy and scalability, building on this thesis's contributions to advance behavioral biometrics globally.

Bibliographie

[1] M.N. Abdi and M. Khemakhem. A model-based approach to offline text-independent arabic writer identification and verification. *Pattern Recognition*, 2014.

[2] M.N. Abdi and M. Khemakhem. Offline text-independent arabic writer identification based on beta-elliptic model. *Pattern Recognition*, 2015.

[3] B. Q. Ahmed, Y. F. Hassan, and A. S. Elsayed. Offline text-independent writer identification using a codebook with structural features. *PLoS ONE*, 18(4) :e0284680, 2023.

[4] Anonymous. A survey on different levels of fusion in multimodal biometrics. *Indian Journal of Science and Technology*, 2019.

[5] B. Arazi. Handwriting identification by means of run-length measurements. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(12) :878–881, 1977.

[6] Author(s). Improving codebook-based writer recognition. *World Scientific Publishing*, 2013.

[7] Author(s). Influence of codebook patterns on writer recognition. *Expert Systems*, 2020.

[8] T. Bahram. A texture-based approach for offline writer identification. *Pattern Recognition Letters*, 33(15) :1696–1704, 2012.

[9] T. Bahram et al. Textural features for writer identification. *Journal of Computer Science*, 2016.

[10] Tayeb Bahram. Offline writer identification using texture features of connected-components. Correspondence : tayeb.bahram@univ-saida.dz.

[11] Tayeb Bahram. A texture-based approach for offline writer identification. *Arabian Journal for Science and Engineering*, 47(12) :14945–14956, 2022.

[12] Tayeb Bahram. Cours 1 - introduction aux systèmes biométriques, 2024.

[13] Tayeb Bahram. Cours 2 - systèmes biométriques, 2024.

[14] B. S. Baumer, D. T. Kaplan, and N. J. Horton. *Modern Data Science with R*. CRC Press, Boca Raton, FL, 2021.

[15] A. Bennour, C. Djeddi, A. Gattal, I. Siddiqi, and T. Mekhaznia. Handwriting based writer recognition using implicit shape codebook. *Pattern Recognition Letters*, 2019.

[16] A. Bennour, C. Djeddi, and T. Mekhaznia. Using codebooks of fragmented connected-component contours in forensic and historic writer identification. *Pattern Recognition Letters*, 27(12) :1443–1453, 2006.

[17] D. Bertolini et al. Texture-based descriptors for writer identification. *Pattern Recognition*, 2013.

[18] Diego Bertolini, Luiz S. Oliveira, Edson Justino, and Robert Sabourin. Texture-based descriptors for writer identification and verification. *Expert Systems with Applications*, 40(6) :2069–2080, 2013.

[19] A.A. Brink, J. Smit, M.L. Bulacu, and L.R.B. Schomaker. Writer identification using directional ink-trace width measurements. *Pattern Recognition*, 45(1) :162–171, 2012.

[20] M. Bulacu et al. Textural features for handwriting analysis. *Pattern Recognition*, 2007.

[21] M. Bulacu and L. Schomaker. Text-independent writer identification and verification on offline arabic handwriting. *IEEE TPAMI*, 2007.

[22] M. Bulacu and L. Schomaker. Text-independent writer identification and verification using textural and allographic features. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 769–773, 2007.

[23] Marcel Bulacu and Lambert Schomaker. Text-independent writer identification and verification using textural and allographic features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4) :701–717, 2007.

[24] A. Chahi et al. Local gradient full-scale transform patterns for writer identification. *Pattern Recognition*, 2020.

[25] Dai Dehui and Li Zhiyong. A fast connected components analysis algorithm for object extraction. In Fatos Xhafa, Srikanta Patnaik, and Albert Y. Zomaya, editors, *Advances in Intelligent Systems and Interactive Applications*, pages 353–360, Cham, 2018. Springer International Publishing.

[26] C. Djeddi et al. Influence of codebook patterns on writer recognition : An experimental study. *Pattern Recognition*, 2021.

[27] C. Djeddi, M. Kherallah, N. Ben Amara, and A. Belaid. Arabic writer identification : A review of literature. *Journal of Theoretical and Applied Information Technology*, 69(3) :474–484, 2014.

[28] C. Djeddi and L. Meslati. A texture based approach for arabic writer identification and verification. In *2010 International Conference on Machine and Web Intelligence (ICMWI)*, pages 115–120, Algiers, Algeria, 2010. IEEE.

[29] C. Djeddi, I. Siddiqi, L. Meslati, and A. Ennaji. Text-independent writer recognition using multi-script handwritten texts. *Pattern Recognition Letters*, 34(10) :1196–1202, 2013.

[30] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley-Interscience, New York, NY, USA, 2nd edition, 2000.

[31] A. Durou et al. Writer identification approach based on bag of words with obi features. *Pattern Recognition*, 2019.

[32] M. Faundez-Zanuy. Data fusion in biometrics. *IEEE Aerospace and Electronic Systems Magazine*, 20(1) :34–38, 2005.

[33] GeeksforGeeks. What are contours in computer vision ?, 2023. Accessed : 2025-06-03.

[34] G. Ghiasi and R. Safabakhsh. Offline text-independent writer identification using codebook and efficient code extraction methods. *Pattern Recognition*, 2013.

[35] Abeer George Ghuneim. Contour tracing algorithms. https://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/mmain.html#alg, n.d. Accessed : 2025-06-03.

[36] Y. Hannad et al. Writer identification using hog and glrl features. *Pattern Recognition*, 2019.

[37] Yaâcoub Hannad, Imran Siddiqi, Chawki Djeddi, and Mohamed El-Youssfi El-Kettani. Improving arabic writer identification using score-level fusion of textural descriptors. *IET Biometrics*, 8(1) :3, 2019.

[38] R. K. Hanusiak, L. S. Oliveira, E. Justino, and R. Sabourin. Writer verification using texture-based features. *International Journal on Document Analysis and Recognition (IJDAR)*, 14(3) :215–230, 2011.

[39] R. K. Hanusiak, Luiz S. Oliveira, Edson Justino, and Robert Sabourin. Writer verification using texture-based features. *International Journal on Document Analysis and Recognition (IJDAR)*, 14(3) :215–230, 2011.

[40] S. He, P. Samara, J. Burgers, and L. Schomaker. Image-based historical manuscript dating using contour and stroke fragments. *Pattern Recognition Letters*, 35 :157–164, 2014.

[41] S. He and L. Schomaker. Delta-n hinge : rotation-invariant features for writer identification. *Pattern Recognition*, 63 :312–323, 2015.

[42] S. He, M. Wiering, and L. Schomaker. Junction detection in handwritten documents and its application to writer identification. *Pattern Recognition*, 2015.

[43] Sheng He and Lambert Schomaker. Writer identification using curvature-free features. *Pattern Recognition*, 63 :451–464, 2017.

[44] Inter-university Consortium for Political and Social Research (ICPSR). *Codebook Preparation Manual*. University of Michigan, 1983. Available at : https://www.icpsr.umich.edu/web/pages/ICPSR/guide.html.

[45] KDnuggets. Introduction to image segmentation with k-means clustering. https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html, 2019. Accessed : 2025-06-03.

[46] Y. Kessentini et al. Edge-hinge and run-length features in dst for writer identification. *Pattern Recognition*, 2018.

[47] E. Khalifa, S. Al-Maadeed, M. A. Tahir, et al. Off-line writer identification using an ensemble of grapheme codebook features. *Pattern Recognition Letters*, 59 :18–25, 2015.

[48] E. Khalifa et al. Off-line writer identification using an ensemble of grapheme codebook features. *Pattern Recognition*, 2015.

[49] F.A. Khan et al. Robust off-line text independent writer identification using bagged discrete cosine transform features. *Pattern Recognition*, 2017.

[50] F.A. Khan et al. Sift and rootsift descriptors with gaussian mixture models. *Pattern Recognition*, 2019.

[51] A. Kumar, S. S. S. R. Depuru, et al. Handwriting identification and verification using artificial intelligence and texture features. *Scientific Reports*, 2023.

[52] R. Kumar and J.D. Sharma. Dcwi : Distribution descriptive curve and cellular automata descriptors. *Pattern Recognition*, 2019.

[53] G. Louloudis, B. Gatos, N. Stamatopoulos, and A. Papandreou. Icdar2013 competition on writer identification. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1397–1401, 2013.

[54] G. Louloudis, N. Stamatopoulos, and B. Gatos. Icdar 2011 writer identification contest. In *2011 International Conference on Document Analysis and Recognition*, pages 1475–1479, 2011.

[55] D. Lu and Q. Weng. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*, 28(5) :823–870, 2007.

[56] Timothy M. What is thresholding in image processing ? a guide, July 2024. Accessed : 2025-06-04.

[57] U.-V. Marti and H. Bunke. The iam-database : an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 5(1) :39–46, 2002.

[58] MathWorks. Label and measure objects in a binary image. `https://www.mathworks.com/help/images/label-and-measure-objects-in-a-binary-image.html`, n.d. Accessed : 2025-06-03.

[59] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1) :62–66, 1979.

[60] S. Prabhakar and A. K. Jain. Decision-level fusion in fingerprint verification. *Pattern Recognition*, 35(4) :861–874, 2002.

[61] S. RahmanGour, C. KarmakerRobert, and J. Bignall. Improving image classification using extended run length features. In D. Huijsmans, D. P. Dionysius, and W. A. Smeulders, editors, *Visual Information and Information Systems*, pages 475–482. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.

[62] R. Raju and P. Reddy. Writer identification using texture features in kannada handwritten documents. *International Journal of Engineering Research and Technology*, 7(13) :1–7, 2018.

[63] A. Ross and R. Govindarajan. Feature level fusion using hand and face biometrics. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*, volume 5779, pages 196–204, Orlando, USA, 2005. SPIE.

[64] A. Ross and A. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24(13) :2115–2125, 2003.

[65] A. Sabri, A. Irfan, G. Wasfi, M. Alshayeb, M. Parvez, M. Märgner, and G. Fink. Writer identification for arabic and latin scripts using codebook features. *Pattern Recognition*, 47(3) :1096–1112, 2014.

[66] Krystian Safjan. Metrics used to compare histograms. `https://safjan.com/metrics-to-compare-histograms/`, 2021. Accessed : 2025-06-05.

[67] S. Said et al. Writer identification using texture descriptors of handwritten fragments. *Pattern Recognition Letters*, 2000.

[68] I. Siddiqi and N. Vincent. Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features. *Pattern Recognition*, 2010.

[69] I. Siddiqi and N. Vincent. Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features. *Pattern Recognition*, 43(11) :3853–3865, 2010.

[70] B. Singh et al. Local binary pattern and center symmetric local binary co-occurrence pattern. *Pattern Recognition*, 2018.

[71] Substance Abuse and Mental Health Services Administration (SAMHSA). *Codebook Development and Usage*, 2024. Available at : https://www.samhsa.gov/data/codebook-development.

[72] Vipin. Mastering contouring in opencv : A comprehensive guide, 2024. Accessed : 2025-06-03.

[73] X. Wu et al. Scale invariant feature transform for writer identification. *Pattern Recognition*, 2014.

[74] Y.-F. Yao, X.-Y. Jing, and H.-S. Wong. Face and palmprint feature level fusion for single sample biometrics recognition. *Neurocomputing*, 70(7) :1582–1586, 2007.

[75] D. Zhang, F. Song, Y. Xu, et al. Decision level fusion. In *Advanced Pattern Recognition Technologies with Applications to Biometrics*, pages 328–348. IGI Global, USA, 2009.

ملخص

يهدف هذا البحث إلى تطوير نظام فعّال للتعرف على هوية الكاتب في بيئات متعددة اللغات، وذلك من خلال استخراج خصائص معتمدة على حدود النصوص اليدوية.

تمت عملية التقييم باستخدام قاعدة بيانات ICDAR 2013، التي تحتوي على عينات كتابية باللغتين الإنجليزية واليونانية وتعميم النتائج على قاعدة بيانات ICDAR 2011

يعتمد النظام المقترح على واصفتين رئيسيتين: F1، التي تستند إلى تحليل البكسلات السوداء، وF2، التي تعتمد على البكسلات البيضاء.
تم استخدام كل واصفة بشكل مستقل، وكذلك تم دمجهما باستخدام قواعد مختلفة لدمج البيانات.

أظهرت النتائج أن استخدام مسافة كاي-تربيع بالتكامل مع قاعدة الدمج الجمعي (SUM rule) يحقق أفضل أداء، من حيث نسب التعرف الصحيحة.
تؤكد هذه النتائج فعالية المنهج المعتمد، وخصوصًا في سياقات متعددة اللغات، كما تُبرز أهمية تنوع العينات في تحسين دقة النظام.

الكلمات المفتاحية: تحديد هوية الكاتب، القياسات البيومترية السلوكية، MLBP-IWSL، تحليل الكتابة اليدوية، استخراج الميزات، التعلم العميق.

Abstract

This work focuses on the development of a multilingual writer identification system that leverages contour-based features extracted from document images. Using the ICDAR 2013 dataset, we conduct mono-script and cross-script evaluations, with particular attention to English and Greek scripts. Two key descriptors, F1 (black pixel-based) and F2 (white pixel-based), are used independently and in fusion. Several distance metrics and fusion rules are compared to identify optimal configurations. Notably, the Chi-squared distance and SUM rule consistently yield the highest accuracy. Our findings demonstrate the robustness of the proposed method across mono- and multi-script scenarios, particularly under hybrid settings that simulate real-world multilingual writing conditions. The system shows strong scalability and performance improvement with increased sample diversity.

Keywords : Writer Identification, Behavioral Biometrics, MLBP-IWSL, Handwriting Analysis, Feature Extraction, Deep Learning.

Résumé

Ce travail porte sur le développement d'un système d'identification de l'auteur dans un contexte multilingue, basé sur des descripteurs extraits des contours des documents manuscrits. Les expérimentations ont été réalisées sur la base de données ICDAR 2013 en anglais et en grec. Deux descripteurs, F1 (pixels noirs) et F2 (pixels blancs), sont utilisés individuellement et en combinaison selon différentes règles de fusion. Les résultats montrent que la distance du chi-deux et la règle de la somme offrent les meilleures performances, démontrant la robustesse de la méthode proposée dans des contextes multilingues, notamment lorsque la diversité des échantillons augmente.

Mots clés : Identification d'écrivains, Biométrie comportementale, MLBP-IWSL, Analyse d'écriture, Extraction de caractéristiques, Apprentissage profond.