République Algérienne Démocratique et Populaire الجمهورية
الجزائرية الديمقراطية الشعبية
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique وزارة التعليم
العالي و البحث العلمي



Université Mohamed El Bachir El Ibrahimi
Bordj Bou Arréridj
جامعة محمد البشير الابراهيمي برج بوعريريج
Faculté des Mathématiques et d'Informatique
Département d'informatique

**Mémoire de Projet de Fin d'Études**

Pour obtenir un diplôme de master en informatique

**Option :**

**Réseaux et Multimédia / Informatique Décisionnelle**

# Fingerprint Classification for AFIS Search-Space Reduction Using Deep Learning

*Préparé par :*
M.Mechri abdelhafid
M.Tebbi mohammed amine

*Encadré par :*
M. Foudil Belhadj

**Promotion 2025/2026**



# Dédicaces

*À nos chères mères et à nos chers pères,*

*qui n'ont jamais cessé de prier pour nous,*
*de nous soutenir et de nous encourager*
*afin que nous puissions atteindre nos objectifs.*

*À nos frères et sœurs,*

*qui nous ont aidés et soutenus*
*dans les moments difficiles.*

*À nos amis,*

*pour leur soutien moral, leurs précieux conseils*
*et leur aide tout au long de notre parcours.*

# Remerciements

*Nous remercions avant tout Allah, le Tout-Puissant et le Miséricordieux, de nous avoir donné la force, le courage et la patience d'accomplir ce modeste travail.*

*Nous exprimons notre profonde gratitude à nos chers parents, notre fierté et la source de notre réussite, pour tous les sacrifices qu'ils ont consentis afin de nous offrir les meilleures conditions de travail.*

*Nous tenons à exprimer notre sincère reconnaissance à notre encadrant, M. Foudil Belhadj, pour ses précieux conseils, sa disponibilité et son soutien tout au long de ce travail.*

*Nous remercions également tous nos enseignants de la Faculté des Mathématiques et d'Informatique, qui nous ont accompagnés durant notre parcours universitaire.*

*Enfin, nous remercions toutes les personnes qui, de près ou de loin, ont contribué à l'aboutissement de ce travail.*

## Abstract

Fingerprint-based Automatic Fingerprint Identification Systems (AFIS) perform 1:N matching by comparing each query against every enrolled template in the gallery, a procedure whose computational cost scales linearly with gallery size and becomes prohibitive at operational scale. This thesis addresses that bottleneck by developing a deep learning-based Henry-class pre-classification stage that partitions the enrolled gallery into five class-labelled bins before matching, restricting each identification query to the relevant subset and reducing the active comparison set without discarding correct candidates.

The classification system was developed and evaluated on NIST Special Database 4 (NIST SD4), a 2 000-image benchmark comprising matched F/S impression pairs across all five Henry classes. Three fingerprint input representations were investigated — raw CLAHE-enhanced images, Orientation Field Enhancement Maps (OFEM), and HSV-encoded orientation fields — and five ImageNet-pretrained convolutional backbone architectures were evaluated: EfficientNet-B3, ResNet50, MobileNetV3-Small, VGG16-BN, and ConvNeXt-Tiny. Orientation-field encoding consistently outperformed raw and OFEM representations across all architectures, establishing input representation quality as the dominant performance factor. ConvNeXt-Tiny, trained on a dual raw-and-orientation-field input, achieved 92.50% standalone accuracy. A binary VGG16-BN specialist, trained with Focal Loss and controlled oversampling to address the chronic underperformance on the Tented Arch minority class, raised the Tented Arch F1-score from 0.606 to 0.808. Integration of the specialist through a confidence-gated veto mechanism produced the final dual-architecture system, reaching 93.25% overall classification accuracy.

The operational contribution of the classification system was validated by integrating it with the SourceAFIS 3.18.1 minutiae-based matching engine within a complete 1:N identification pipeline. Three search-space scenarios were evaluated: full-gallery baseline (Scenario A, 2 000 comparisons), single-bin classification-guided search (Scenario B, ≈400 comparisons), and dual-bin search (Scenario B+, ≈800 comparisons). The B+ strategy preserved near-complete identification accuracy — 87.50% Rank-1 — while reducing active comparisons by 2.61× relative to the full-gallery baseline. A Gabor score-fusion stage, applying Gabor-enhanced template matching in parallel with raw matching and combining scores via a conservative max-fusion rule, lowered the Equal Error Rate from 13.36% to 8.58% by recovering 153 identification failures attributable to poor-quality probe impressions, while breaking only 14 previously passing matches.

The principal finding of this thesis is the empirical demonstration that a deep learning classification front-end, integrated with a traditional minutiae matcher, delivers measurable and reproducible search-space reduction without sacrificing identification accuracy. The dual-architecture design, the Gabor fusion strategy, and the honest characterisation of the conditions under which the approach succeeds and fails together constitute a reproducible and extensible foundation for future work in adaptive AFIS search-space reduction.



*Keywords: Fingerprint classification · Henry system · AFIS · Deep learning · ConvNeXt · Transfer learning · Orientation field · OFEM · SourceAFIS · Search-space reduction · Score fusion · NIST SD4*

# Résumé

Les systèmes automatisés d'identification par empreinte digitale (AFIS) effectuent une recherche 1:N en comparant chaque requête à l'ensemble des gabarits enrôlés dans la galerie, une procédure dont le coût computationnel croît linéairement avec la taille de la galerie. Ce mémoire répond à ce problème de passage à l'échelle en développant un étage de pré-classification basé sur l'apprentissage profond, qui partitionne la galerie enrôlée en cinq classes Henry avant la phase de recherche, limitant chaque requête d'identification au sous-ensemble pertinent.

Le système de classification a été développé et évalué sur NIST Special Database 4 (NIST SD4), un corpus de 2 000 images couvrant les cinq classes Henry. Trois représentations d'entrée — images CLAHE brutes, cartes OFEM, et champs d'orientation HSV — ont été étudiées, et cinq architectures convolutives pré-entraînées sur ImageNet ont été évaluées : EfficientNet-B3, ResNet50, MobileNetV3-Small, VGG16-BN et ConvNeXt-Tiny. Le codage par champ d'orientation s'est révélé systématiquement supérieur aux autres représentations. ConvNeXt-Tiny, entraîné sur une entrée double (image brute + champ d'orientation), a atteint 92,50 % de précision. Un spécialiste binaire VGG16-BN, entraîné avec Focal Loss et sur-échantillonnage contrôlé pour la classe Tented Arch, a élevé son F1-score de 0,606 à 0,808. Le système final à double architecture a atteint 93,25 % de précision globale.

La contribution opérationnelle du classificateur a été validée par son intégration avec le moteur de correspondance SourceAFIS 3.18.1 dans un pipeline complet d'identification 1:N. La stratégie à double bin (B+) a préservé une précision d'identification de 87,50 % Rank-1 tout en réduisant les comparaisons actives d'un facteur 2,61×. Une fusion de scores par filtre de Gabor a réduit le taux d'égalité d'erreur de 13,36 % à 8,58 %, récupérant 153 échecs d'identification liés à une mauvaise qualité d'image.



*Mots-clés : Classification d'empreintes · Système Henry · AFIS · Apprentissage profond · ConvNeXt · Transfert d'apprentissage · Champ d'orientation · OFEM · SourceAFIS · Réduction de l'espace de recherche · Fusion de scores · NIST SD4*

## ملخص

تُجري أنظمة التعرف التلقائي على بصمات الأصابع (AFIS) عملية المطابقة N:1 بمقارنة كل استعلام مع جميع القوالب المسجلة في قاعدة البيانات، وهي عملية يتصاعد تكلفتها الحسابية بشكل خطي مع حجم القاعدة. تعالج هذه الأطروحة هذه الإشكالية من خلال تطوير مرحلة تصنيف مسبق بنظام هنري تعتمد على التعلم العميق، تقسّم قاعدة البيانات المسجلة إلى خمس خانات مصنّفة قبل المطابقة، مما يحدّ نطاق البحث لكل استعلام في الخانة المعنية فقط.

تم تطوير نظام التصنيف وتقييمه على قاعدة البيانات NIST SD4 المكوّنة من 4000 صورة بصمة موزعة على الفئات الخمس لنظام هنري. جرى تقييم خمس بنى شبكية عصبونية تلافيفية مُدرَّبة مسبقاً على ImageNet: EfficientNet-B3 وResNet50 وMobileNetV3-Small وVGG16-BN وConvNeXt-Tiny. أثبت ترميز مجال الاتجاه تفوقه باستمرار على التمثيلات الأخرى. حقق ConvNeXt-Tiny دقةً انفراديةً بلغت 92.50٪، في حين رفع المصنّف الثنائي المتخصص VGG16-BN درجة F1 لفئة Tented Arch من 0.606 إلى 0.808. وحقق النظام الثنائي النهائي دقةً إجماليةً قدرها 93.25٪.

تم التحقق من المساهمة التشغيلية للنظام بدمجه مع محرك المطابقة SourceAFIS 3.18.1 ضمن خط أنابيب تعريف N:1 متكامل. حافظت استراتيجية البحث ثنائية الخانة (B+) على دقة تعريف 87.50٪ Rank-1 مع تقليص عدد المقارنات الفعلية بعامل 2.61×. كما خفّض نظام دمج الدرجات بمرشح Gabor معدل الخطأ المتساوي من 13.36٪ إلى 8.58٪.



*الكلمات المفتاحية: تصنيف بصمات الأصابع · نظام هنري · AFIS · التعلم العميق · ConvNeXt · التعلم بالنقل · SourceAFIS · OFEM · تقليص فضاء البحث · NIST SD4*

# List of Abbreviations

| | |
|---|---|
| **AFIS** | Automatic Fingerprint Identification System |
| **ArcFace** | Additive Angular Margin loss for face/fingerprint recognition |
| **CLAHE** | Contrast-Limited Adaptive Histogram Equalisation |
| **CNN** | Convolutional Neural Network |
| **ConvNeXt** | Convolutional Network (2022 modernised ResNet design) |
| **CPU** | Central Processing Unit |
| **DDR4** | Double Data Rate 4 (memory standard) |
| **DeiT** | Data-efficient Image Transformer |
| **EER** | Equal Error Rate |
| **FAR** | False Acceptance Rate |
| **FC** | Fully-Connected layer |
| **FMR** | False Match Rate |
| **FNMR** | False Non-Match Rate |
| **FRR** | False Rejection Rate |
| **FVC** | Fingerprint Verification Competition |
| **GPU** | Graphics Processing Unit |
| **HSV** | Hue–Saturation–Value (colour encoding used for orientation fields) |
| **ImageNet** | Large-scale visual recognition benchmark dataset (1.2 M images, 1 000 classes) |
| **KYC** | Know Your Customer |
| **LR** | Learning Rate |
| **MCC** | Minutia Cylinder-Code |
| **NIST** | National Institute of Standards and Technology |
| **NIST SD4** | NIST Special Database 4 (fingerprint benchmark, 2 000 images) |
| **OFEM** | Orientation Field Enhancement Map |
| **POOL** | Pooling layer |
| **ReLU** | Rectified Linear Unit |
| **ResNet** | Residual Network |
| **ROC** | Receiver Operating Characteristic |
| **SGD** | Stochastic Gradient Descent |
| **SSD** | Solid-State Drive |
| **SourceAFIS** | Open-source minutiae-based fingerprint matching engine (v3.18.1) |
| **Swin** | Shifted Window Transformer |
| **TA** | Tented Arch (Henry class) |
| **TTA** | Test-Time Augmentation |
| **VGG** | Visual Geometry Group (Oxford) — CNN architecture family |
| **ViT** | Vision Transformer |

# Table of Contents

**Context and Motivation** ..... 1
**Problem Statement** ..... 2
**Research Objectives** ..... 3
**Principal Contributions** ..... 3
**Thesis Organisation** ..... 4
**1.1 Introduction** ..... 5
**1.2 Definition and Fundamental Concepts** ..... 6
**1.3 Biometric System Architecture** ..... 7
**1.4 Biometric Modalities** ..... 8
**1.4.1 Physiological Modalities** ..... 8
**1.4.2 Behavioural Modalities** ..... 9
**1.5 Performance Evaluation of Biometric Systems** ..... 10
**1.6 Advantages and Limitations of Biometric Systems** ..... 12
**1.6.1 Advantages** ..... 12
**1.6.2 Limitations** ..... 12
**1.7 Conclusion** ..... 13
**2.1 Introduction** ..... 14
**2.2 Fingerprint Ridge Characteristics** ..... 15
**2.2.1 Hierarchical Feature Levels** ..... 15
**2.2.2 Singular Points and Fingerprint Topology** ..... 16
**2.3 Classical Fingerprint Classification Systems** ..... 17
**2.3.1 Historical Background** ..... 17
**2.3.2 The Henry Classification System** ..... 17
**2.3.3 Limitations of Classical Classification Systems** ..... 19
**2.4 Automated Fingerprint Classification Approaches** ..... 19
**2.4.1 Model-Based and Structural Methods** ..... 20
**2.4.2 Frequency-Domain and Orientation-Based Methods** ..... 20
**2.4.3 Why Classical Automated Methods Hit a Ceiling** ..... 21
**2.5 Deep Learning for Fingerprint Classification** ..... 21
**2.5.1 Orientation Image-Based CNN Classification** ..... 21
**2.5.2 VGGNet Ensemble Classification** ..... 22
**2.5.3 Recent Advances (2019–2025)** ..... 22
**2.5.4 Comparative Analysis** ..... 23
**2.6 Overview of the Proposed Approach** ..... 24
**2.7 Conclusion** ..... 25
**3.1 Introduction** ..... 26
**3.2 Experimental Environment** ..... 26
**3.2.1 Hardware Configuration** ..... 26
**3.2.2 Software Environment** ..... 27
**3.2.3 Training Configuration and Hyperparameters** ..... 28
**3.3 Dataset Description** ..... 30
**3.3.1 NIST Special Database 4 — Overview** ..... 30
**3.3.2 The Henry Classification System** ..... 31
**3.3.3 Dataset Preparation** ..... 32
**3.4 Fingerprint Representation and Preprocessing** ..... 33
**3.4.1 Raw Fingerprint Representation** ..... 33
**3.4.2 Orientation Field Representation** ..... 34
**3.4.3 Ridge Topology as the Basis of Classification** ..... 36
**3.4.4 OFEM: Iterative Development of the Orientation Field Enhancement Map** ..... 36
**3.4.5 Comparative Analysis of Fingerprint Representations** ..... 42

**3.5 Classification Paradigm and Optimisation Strategy** ..... 43
**3.5.1 Supervised Multi-Class Formulation** ..... 44
**3.5.2 Choice of Optimiser: AdamW** ..... 44
**3.5.3 Choice of Convolutional Networks over Vision Transformers** ..... 45
**3.6 Deep Learning Architectures for Fingerprint Classification** ..... 46
**3.6.1 Fundamentals of Convolutional Neural Networks** ..... 46
**3.6.1.7 Why CNNs Were Selected for Fingerprint Classification** ..... 49
**3.6.2 Evaluated Deep Learning Architectures** ..... 49
**3.6.2.1 EfficientNet-B3** ..... 50
**3.6.2.2 VGG16-BN with OFEM Representation** ..... 50
**3.6.2.3 ResNet50 with Orientation-Field Representation** ..... 52
**3.6.2.4 MobileNetV3-Small — Lightweight Specialist Candidate** ..... 53
**3.6.2.5 ConvNeXt-Tiny** ..... 54
**3.7 Specialist Architecture for Tented Arch Recognition** ..... 55
**3.7.1 Motivation for Specialist Classification** ..... 55
**3.7.2 Failure Analysis of General Classifiers** ..... 55
**3.7.3 Specialist Classifier Design** ..... 56
**3.7.4 Handling Class Imbalance** ..... 56
**3.7.6 Contribution to the Final System** ..... 57
**3.8 Ensemble Learning Strategy** ..... 58
**3.8.1 Motivation for Ensemble Learning** ..... 58
**3.8.2 Decision Fusion Strategy** ..... 58
**3.8.3 Ensemble Architecture 1 — Orientation, Texture, and Specialist** ..... 60
**3.8.4 Ensemble Architecture 2 — OFEM, Orientation, and Specialist** ..... 61
**3.8.6 Lessons Learned and Transition Toward the Final Architecture** ..... 61
**3.9 Final Proposed Architecture** ..... 62
**3.9.1 Design Rationale** ..... 62
**3.9.2 Architecture Overview** ..... 63
**3.9.3 Specialist-Assisted Veto Mechanism** ..... 64
**3.9.6 Contribution to AFIS Search-Space Reduction** ..... 64
**3.10 Performance Evaluation Criteria** ..... 65
**Overall Accuracy** ..... 65
**Precision and Recall** ..... 65
**F1-Score** ..... 65
**Confusion Matrix Analysis** ..... 66
**3.11 Implementation** ..... 66
**3.12 Conclusion** ..... 67
**4.1 Introduction** ..... 68
**4.2 EfficientNet-B3 Results** ..... 68
**4.3 VGG16-BN (OFEM) Results** ..... 69
**4.4 ResNet50 (Orientation-Field) Results** ..... 69
**4.5 MobileNetV3-Small Results** ..... 70
**4.6 ConvNeXt-Tiny Results** ..... 70
**4.7 Tented Arch Specialist Results** ..... 71
**4.8 Ensemble Results** ..... 72
**4.9 Ensemble 2 Resultat** ..... 73
**4.10 Final Proposed Architecture Results** ..... 74
**4.11 Comparative Analysis of Classification Models** ..... 76
**4.11.1 Comparative Summary** ..... 76
**4.12 Matching Results and Search-Space Reduction** ..... 77
**4.12.1 Search-Space Reduction Validation** ..... 77
**4.12.2 Introduction to the Matching Stage** ..... 78

**4.12.3 SourceAFIS-Based Fingerprint Matching** ..... 78
**4.12.4 Enrollment and Identification Protocol** ..... 79
**4.12.5 Search-Space Reduction Using Fingerprint Classification** ..... 80
**4.12.6 Gabor-Enhanced Matching Strategy** ..... 81
**4.12.7 Score-Level Fusion** ..... 82
**4.12.8 Final Proposed AFIS Pipeline** ..... 83
**4.13 Discussion** ..... 86
**4.13.1 Influence of Fingerprint Representation** ..... 86
**4.13.2 Analysis of OFEM Performance** ..... 86
**4.13.3 ConvNeXt and Specialist Contributions** ..... 86
**4.13.4 Ensemble and Final Architecture Insights** ..... 87
**4.13.5 Overall Scientific Conclusions** ..... 87
**4.14 Limitations and Future Work** ..... 87
**4.14.1 Current Limitations** ..... 87
**4.14.2 Future Research Directions** ..... 88
**4.15 Conclusion** ..... 88

# List of Figures

**Figure 1.1:** *Generic biometric system architecture.* ..... 7
**Figure 1.2:** *Physiological and behavioural biometric modalities.* ..... 8
**Figure 1.3:** *FAR–FRR trade-off and the Equal Error Rate.* ..... 11
**Figure 2.1:** *Hierarchical fingerprint feature levels.* ..... 16
**Figure 2.2:** *The five Henry fingerprint classes.* ..... 18
**Figure 3.1:** *Representative NIST SD4 samples, one per Henry class.* ..... 34
**Figure 3.2:** *Orientation-field computation pipeline.* ..... 35
**Figure 3.3:** *HSV orientation-field images of the five Henry classes.* ..... 35
**Figure 3.4:** *OFEM Version 1 energy output.* ..... 37
**Figure 3.5:** *OFEM Versions 2–7 progression.* ..... 39
**Figure 3.6:** *OFEM Version 8 energy output.* ..... 40
**Figure 3.7:** *Comparison of the three input representations.* ..... 43
**Figure 3.8:** *Convolutional neural network architecture.* ..... 46
**Figure 3.9:** *The convolution operation.* ..... 47
**Figure 3.10:** *The ReLU activation function.* ..... 47
**Figure 3.11:** *Max and average pooling.* ..... 48
**Figure 3.12:** *The fully-connected layer.* ..... 48
**Figure 3.13:** *EfficientNet-B3 architecture.* ..... 50
**Figure 3.14:** *VGG16-BN architecture.* ..... 51
**Figure 3.15:** *ResNet50 architecture.* ..... 52
**Figure 3.16:** *MobileNetV3-Small architecture.* ..... 53
**Figure 3.17:** *ConvNeXt-Tiny architecture.* ..... 54
**Figure 3.18:** *Ensemble 1 architecture.* ..... 60
**Figure 3.19:** *Ensemble 2 architecture.* ..... 61
**Figure 3.20:** *Final dual-classifier architecture.* ..... 63
**Figure 4.1:** *EfficientNet-B3 performance evaluation.* ..... 68
**Figure 4.2:** *VGG16-BN (OFEM) performance evaluation.* ..... 69
**Figure 4.3:** *ResNet50 performance evaluation.* ..... 69
**Figure 4.4:** *MobileNetV3-Small performance evaluation.* ..... 70
**Figure 4.5:** *ConvNeXt-Tiny performance evaluation.* ..... 70
**Figure 4.6:** *VGG16-BN Tented Arch specialist performance.* ..... 71
**Figure 4.7:** *Ensemble 1 results — confusion matrix and accuracy.* ..... 72
**Figure 4.8:** *Ensemble 2 results — confusion matrix and accuracy.* ..... 73
**Figure 4.9:** *Final ensemble results — confusion matrix and accuracy.* ..... 76
**Figure 4.10:** *SourceAFIS template extraction and matching workflow.* ..... 79
**Figure 4.11:** *Enrollment and identification protocol.* ..... 80
**Figure 4.12:** *Search-space reduction architecture.* ..... 81
**Figure 4.13:** *Raw CLAHE versus Gabor-enhanced fingerprints.* ..... 82
**Figure 4.14:** *Score-level fusion pipeline.* ..... 83
**Figure 4.15:** *Final proposed AFIS identification system.* ..... 84
**Figure 4.16:** *Per-query matching time across scenarios.* ..... 85

# List of Tables

**Table 1.1:** *Comparative assessment of principal biometric modalities against the seven recognition properties. Ratings adapted from Jain et al. [1, 2].* ..... 10
**Table 2.1:** *Comparative analysis of fingerprint classification methods. Results on different datasets are not directly comparable; the FVC databases used by Jeon and Rhee contain higher-quality impressions than NIST SD4.*..... 23
**Table 3.1:** *Hardware configuration of the experimental environment.*..... 27
**Table 3.2:** *Software stack used across all experimental stages.*..... 28
**Table 3.3:** *Hyperparameter configuration for general classifiers and the Tented Arch specialist.*..... 29
**Table 3.4:** *Class distribution across training and test partitions of NIST SD4 after stratified splitting.* ..... 32
**Table 3.5:** *Summary of the eight OFEM generator versions with primary design change and outcome.* ..... 41
**Table 3.6:** *Comparative summary of the three fingerprint input representations investigated in this study.* ..... 43
**Table 4.X:** *Learning task and classification type for each model in the study. All formulations are fully supervised, using the ground-truth Henry class labels provided with NIST SD4.* ..... 44
**Table 3.8:** *Comparative summary of ensemble configurations evaluated in this study.*..... 74
**Table 3.9:** *Performance summary of the final dual-classifier architecture and its constituent components.* ..... 74
**Table 3.10a:** *Quantitative performance comparison of ensemble configurations and the final dual architecture.*.. 75
**Table 3.10b:** *Operational complexity comparison of ensemble configurations and the final dual architecture.* .... 75
**Table 3.7:** *Individual model performance summary across all architecture-representation combinations on NIST SD4.* ..... 76
**Table 4.1:** *Consolidated classification results across all representations, architectures, specialist, and ensemble configurations.*..... 77
**Table 4.2:** *Identification performance across all scenarios and matching configurations. The B+ hybrid row (bold) is the final proposed system default.* ..... 85

# General Introduction

## Context and Motivation

The reliable establishment of human identity has been a persistent requirement of social and security infrastructure across civilisations. For most of recorded history, identity was verified through personal recognition, physical tokens, or documentary credentials — mechanisms whose fundamental weakness is that they authenticate what a person possesses or knows rather than who the person biologically is. The limitations of these approaches became structurally consequential with the rise of large-scale institutional identity management: a nineteenth-century police force maintaining records of tens of thousands of criminal subjects could not rely on personal recognition or document inspection to determine whether an arrested individual had been previously charged. The need for a systematic, scalable, and reliable identity record system drove the earliest scientific investigations into human biometric characteristics as a basis for identification.

Alphonse Bertillon's anthropometric system, introduced in 1879, was the first attempt to apply systematic body measurements to criminal identification. The Bertillon system encoded each individual through a standardised set of eleven skeletal measurements and produced a classification scheme that, within its design constraints, functioned reliably for a population of limited size. Its practical failure became apparent as registration databases grew: the measurement precision required to distinguish individuals across a large population exceeded what was achievable under operational conditions, and the classification scheme collapsed under the combinatorial pressure of large-scale deployment. The decisive transition came in 1892, when Francis Galton published his landmark study demonstrating that fingerprint ridge patterns are both individually unique and permanently stable throughout life [1]. Galton's probabilistic analysis established the scientific foundation for fingerprint identification, and the operational framework was completed by Edward Henry, whose five-class classification system — formally adopted by the United Kingdom Home Office in 1901 — provided the administrative procedure through which large fingerprint collections could be filed, searched, and retrieved without exhaustive comparison [1, 2].

The Henry classification system remained the operational basis of fingerprint identification for most of the twentieth century, administered by trained examiners who visually assigned each impression to one of five structural categories: Plain Arch, Tented Arch, Left Loop, Right Loop, and Whorl. As civil registration and law enforcement databases grew from hundreds of thousands to tens of millions of enrolled subjects, the labour intensity and scalability limitations of expert-based classification became untenable. This pressure drove the development, beginning in the 1970s and accelerating through the 1980s and 1990s, of Automated Fingerprint Identification Systems (AFIS) that could perform acquisition, feature extraction, classification, and matching without continuous human involvement [1, 2]. By the early 2000s, national AFIS deployments in major jurisdictions routinely maintained databases of fifty to one hundred million enrolled fingerprint records, and the computational challenge of searching these databases in near-real-time became the dominant engineering constraint on AFIS design.

## Problem Statement

The fundamental computational challenge of large-scale fingerprint identification is the cost of the one-to-many matching operation. In a 1:N identification query, a probe fingerprint template must be compared against every enrolled gallery template to find the most similar candidate. For a gallery of N enrolled subjects and a matching operation of cost c per comparison, the total query cost scales as $O(N \cdot c)$ — a burden that grows linearly with database size and that becomes operationally prohibitive at the scales characteristic of national civil registration or watch-list screening applications. At a database of one hundred million enrolled subjects, even a highly optimised matcher performing each comparison in one millisecond would require one hundred thousand seconds — over twenty-seven hours — to complete a single exhaustive search [1, 2].

Pre-classification of the enrolled database into structural categories provides the most widely adopted mechanism for reducing this cost. By assigning every enrolled template to a Henry class partition at enrollment time, and assigning the probe fingerprint to the same partition at query time, the system restricts the matching operation to the subset of enrolled templates belonging to the predicted class. For a five-class Henry system with a roughly uniform class distribution, this partitioning reduces the expected comparison count by a factor approaching five — from N comparisons to approximately N/5. The operational significance of this reduction scales directly with database size: at one hundred million enrolled subjects, the theoretical search space drops from one hundred million to twenty million comparisons per query, a reduction that can make the difference between a feasible real-time identification and an intractable offline batch process [1, 2].

The reliability of this acceleration is, however, conditioned entirely on the accuracy of the classification stage. A probe fingerprint assigned to the wrong Henry class partition cannot be matched against any template in its correct partition, because those templates reside in a part of the database that the matching stage never searches. Classification errors therefore manifest as irrecoverable identification failures rather than degraded match scores — a categorically different and operationally more severe failure mode than low match scores, which can be mitigated by threshold adjustment or multi-sample fusion. This asymmetric failure characteristic establishes fingerprint classification accuracy as a hard operational requirement rather than a soft performance indicator: every percentage point of classification error translates directly into an equal percentage of irrecoverable identification failures.

Despite decades of research, automated fingerprint classification remains an incompletely solved problem. Classical automated approaches — model-based singular point detection, orientation field analysis, and handcrafted feature classification — have achieved accuracy levels in the range of 88–92% on the NIST Special Database 4 benchmark, a level that leaves a non-negligible fraction of queries misrouted to wrong partitions [1, 2]. The structural difficulty arises from several concurrent factors: the morphological continuity between adjacent Henry classes, particularly Plain Arch and Tented Arch, which share nearly identical local ridge texture and differ only in the presence of a single core singular point; the severe underrepresentation of minority classes, particularly Tented Arch, in training datasets; and the sensitivity of all orientation-based classification methods to image quality degradation, which is unavoidable in operational acquisition environments.

## Research Objectives

The objective of this thesis is to investigate and evaluate deep learning-based fingerprint classification as a mechanism for improving the accuracy of AFIS search-space reduction, and to validate its operational contribution through integration with a complete fingerprint identification pipeline. The central hypothesis is that convolutional neural networks, trained on appropriate orientation-based fingerprint representations, can achieve classification accuracy levels sufficient to support reliable AFIS pre-filtering — and that this accuracy improvement translates into measurable operational benefits at the identification stage, including reduced search-space size and improved identification accuracy relative to a raw-image baseline.

To test this hypothesis, the research program pursues three specific objectives. The first is to design and evaluate multiple fingerprint input representations that make the topologically discriminative information of the orientation field accessible to convolutional feature learning, including a novel Orientation Field Enhancement Map that explicitly highlights singular-region neighbourhoods through block-level orientation deviation energy analysis. The second is to identify the network architecture and training strategy that maximises classification accuracy on the NIST SD4 benchmark under realistic data constraints, including the investigation of specialist binary classification for the Tented Arch minority class and ensemble strategies for error complementarity exploitation. The third is to evaluate the end-to-end operational contribution of the proposed classification framework within a complete 1:N identification pipeline, measuring the achieved search-space reduction and identification accuracy relative to a full-gallery baseline on the NIST SD4 protocol.

## Principal Contributions

This thesis makes five principal contributions to the field of fingerprint classification and AFIS optimisation. First, a systematic investigation of fingerprint input representations demonstrates that orientation-based inputs — particularly the HSV orientation field encoding of Hong, Wan, and Jain [3] — consistently outperform raw fingerprint images across all evaluated convolutional architectures, and that this performance advantage is attributable to the explicit encoding of ridge-flow topology rather than to any architectural property. Second, a novel Orientation Field Enhancement Map (OFEM) is developed through an eight-version iterative refinement process, producing a block-level orientation deviation energy map that explicitly concentrates classification-relevant information on singular-region neighbourhoods while suppressing the non-discriminative normal-region background. Third, a binary VGG16-BN Tented Arch specialist classifier, trained with Focal Loss and controlled oversampling, achieves a Tented Arch F1-score of 0.808 — a substantial improvement over the 0.606 achieved by the best lightweight general classifier — demonstrating that minority-class recognition requires targeted training strategies rather than architectural scaling alone.

Fourth, an ensemble investigation establishes that representational diversity between component models is a necessary but not sufficient condition for error complementarity, and that a two-component dual-architecture system can outperform three-model ensembles when the component error sources are structurally non-correlated. Fifth, an end-to-end AFIS validation on NIST SD4 demonstrates that the proposed classification framework produces a 2.61× search-space reduction and, in combination with a Gabor score fusion strategy, achieves 87.50% Rank-1 identification accuracy with an Equal Error Rate of 8.58%, confirming the practical operational value of deep learning-based fingerprint classification beyond classification accuracy alone.

## Thesis Organisation

The remainder of this thesis is organised into four chapters. Chapter 1 introduces the foundational concepts of biometric recognition systems, covering the properties that define useful biometric traits, the six-stage processing pipeline common to all biometric systems, a comparative survey of principal biometric modalities, and the performance evaluation framework — FAR, FRR, EER, and CMC — used throughout the experimental evaluation chapters. Chapter 2 develops the theoretical foundations of fingerprint classification specifically, tracing the historical development of the Henry taxonomy, characterising the topological basis of the five Henry classes in terms of singular point configurations, surveying classical and modern automated classification approaches, and reviewing the recent deep learning literature with a structured comparative analysis. Chapter 3 describes the project methodology, dataset, and experimental design. Chapter 4 presents the complete implementation and experimental evaluation of the proposed system, covering fingerprint representation design and the OFEM development, convolutional architecture evaluation, specialist and ensemble learning, the matching stage integration with SourceAFIS and Gabor score fusion, the system limitations, and the overall scientific conclusions of the work.

# Chapter 1.

# Biometric Systems: Fundamentals and Architecture

## 1.1 Introduction

The reliable establishment of human identity is a foundational requirement of modern security infrastructure. Traditional authentication mechanisms — passwords, personal identification numbers, and physical tokens such as access cards — operate on the principle of possession or knowledge: they verify that the individual presents the correct credential rather than confirming that the individual is who they claim to be. This distinction has practical consequences. Passwords can be forgotten, stolen, or shared; physical tokens can be lost or duplicated; and none of these mechanisms provide any inherent link between the credential and the biological individual who holds it. The scale and sophistication of identity fraud in contemporary digital environments has exposed the limitations of these approaches and motivated the adoption of authentication systems grounded in measurable biological or behavioural characteristics that are intrinsically bound to the individual [1, 2].

Biometrics is the scientific discipline that addresses this need by providing automated means of recognising individuals based on their physiological or behavioural attributes. Unlike credential-based approaches, biometric recognition does not verify knowledge or possession but establishes a direct link between a biological measurement and an enrolled identity record, producing an authentication decision that is difficult to transfer, share, or falsify without the physical presence of the enrolled individual [1]. Over the past three decades, biometric technologies have transitioned from specialised forensic tools into mainstream identity management infrastructure, deployed across applications ranging from national electronic passports and border control systems to consumer device unlocking and financial transaction authentication.

This chapter introduces the fundamental concepts, architectural components, and performance evaluation framework of biometric systems. It establishes the theoretical context within which the fingerprint classification and identification system developed in this thesis operates. Section 1.2 defines biometrics and characterises the properties that a measurable trait must satisfy to be useful for recognition. Section 1.3 describes the processing pipeline common to all biometric systems. Section 1.4 surveys the principal biometric modalities with particular attention to fingerprint, which is the focus of this work. Section 1.5 establishes the performance evaluation metrics that are referenced throughout the experimental chapters.

Section 1.6 discusses the advantages and inherent limitations of biometric approaches. Section 1.7 concludes with a transition toward the fingerprint-specific theoretical foundations of Chapter 2.

## 1.2 Definition and Fundamental Concepts

A biometric trait is any measurable biological characteristic or behavioural pattern that can be acquired, digitised, and compared across individuals to establish or verify identity. Not every human characteristic qualifies: Jain, Ross, and Prabhakar [2] establish seven requirements that a trait must satisfy to be useful in a biometric recognition context. Universality requires that the trait be present in all target individuals. Uniqueness requires that the trait distinguish individuals from one another with sufficiently low probability of collision. Permanence requires that the trait remain stable over time and across measurement sessions. Collectability requires that the trait be quantitatively measurable by an appropriate sensor. Performance requires that the resulting recognition system achieve acceptable accuracy and speed under realistic operational conditions. Acceptability requires that individuals be willing to present the trait for measurement. Circumvention resistance requires that the trait be difficult to imitate or spoof.

No single biometric trait satisfies all seven requirements equally well, and the selection of a modality for a given application involves a trade-off among these properties calibrated to the security level, user population, and deployment environment of the target system [1, 2]. Fingerprint, the modality studied in this thesis, scores strongly on uniqueness — no two individuals are known to share identical fingerprint ridge patterns, even monozygotic twins — and on permanence, since the ridge structure of an adult finger remains stable over decades. Its collectability and acceptability have been validated at national scales through multi-decade deployment in law enforcement, civil registration, and border control systems [1].

Within biometric recognition, two operational modes are distinguished by the nature of the identity claim being evaluated. Verification is a one-to-one comparison: the subject asserts an identity by presenting a credential, and the system compares the biometric measurement against the single enrolled template corresponding to the claimed identity, returning a binary match or non-match decision. Identification is a one-to-many search: the subject makes no identity claim, and the system compares the biometric measurement against all enrolled templates in the database, returning the identity of the highest-scoring candidate above a match threshold, or declaring no match if no candidate meets the threshold [1, 3]. Identification is the more computationally demanding task and the one directly relevant to this thesis: the AFIS pipeline described in Chapter 4 implements 1:N identification against a gallery of 2 000 enrolled fingerprint subjects.

## 1.3 Biometric System Architecture

Regardless of the specific modality or application, biometric recognition systems share a common six-stage processing pipeline whose design choices at each stage determine the overall system performance, robustness, and deployability [1, 2]. Figure 1.1 illustrates this pipeline; the following paragraphs describe the function and engineering considerations of each stage.

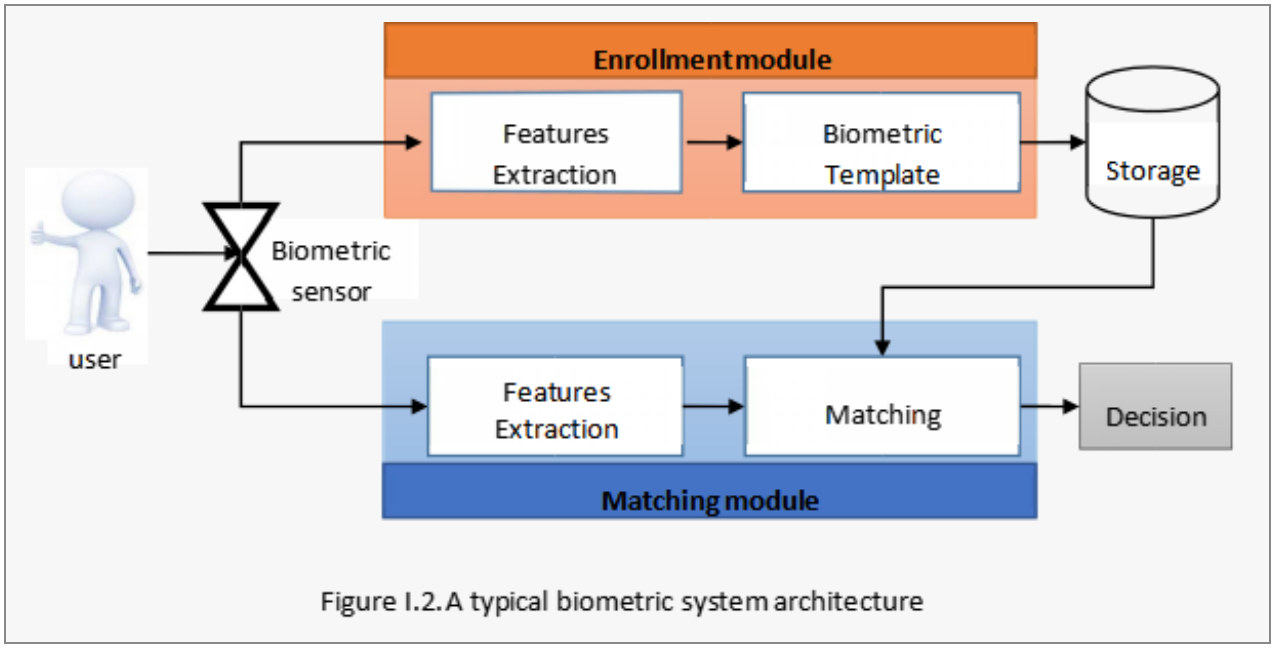



**Figure 1.1:** *Generic biometric system architecture.*

The acquisition stage captures a raw digital representation of the biometric trait using an appropriate sensor: optical or capacitive scanners for fingerprint, cameras for face and iris, microphones for voice. Sensor quality is a primary determinant of downstream performance: a low-contrast or noisy acquisition cannot be fully compensated by subsequent processing, and the operational context — controlled laboratory versus unattended kiosk versus forensic scene — imposes widely varying constraints on acquisition quality [1]. For fingerprint specifically, the 500 dpi resolution specified by standards such as NIST and ISO is the minimum required to resolve individual ridge structures reliably, and the images in NIST SD4 used in this thesis conform to this standard.

The preprocessing stage normalises the raw acquisition to reduce the effect of sensor and environmental variability before feature extraction. For fingerprint images this includes segmenting the foreground ridge area from the background, correcting contrast and intensity non-uniformity through adaptive equalisation, and optionally applying enhancement filters to recover ridge continuity in low-quality regions. The specific preprocessing pipeline adopted in this work — CLAHE contrast enhancement, foreground segmentation by local variance thresholding, and orientation-field generation — is described in detail in Section 4.4.

Feature extraction transforms the preprocessed image into a compact, discriminative representation that encodes the identity-relevant information while discarding the large volume of class-irrelevant pixel content. The choice of feature representation has a dominant influence on recognition accuracy, a finding consistently confirmed by the fingerprint classification literature and by the experimental results reported in Chapter 4.

Feature representations range from handcrafted descriptors such as minutiae point sets, orientation field arrays, and Gabor filter response vectors, to learned representations extracted by convolutional neural networks from labelled training data [1, 4].

Template generation produces a standardised, storable encoding of the extracted features. In minutiae-based fingerprint systems, the template encodes the positions and orientations of detected ridge endings and bifurcations as a compact byte array. The matching stage computes a similarity score between a probe template and one or more gallery templates, applying a geometric alignment algorithm to compensate for translation, rotation, and elastic distortion between different impressions of the same finger. The decision stage applies a threshold to the similarity score to produce a match or non-match outcome. In identification mode, candidates are ranked by score and the top-ranked candidate above the threshold is returned as the identity decision.

## 1.4 Biometric Modalities

Biometric modalities are conventionally divided into two categories according to the nature of the measured characteristic. Physiological modalities are based on anatomical properties of the body that are determined by genetics and remain relatively stable over time. Behavioural modalities are based on patterns of motor activity that reflect learned or habitual physical actions and may exhibit greater intra-subject variability across sessions [2, 3].

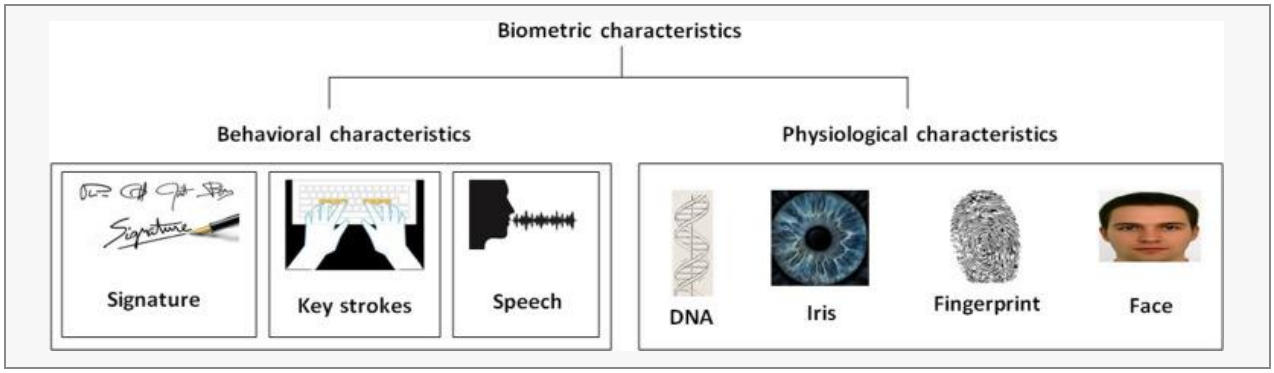



**Figure 1.2:** *Physiological and behavioural biometric modalities.*

### 1.4.1 Physiological Modalities

Fingerprint is the most widely deployed biometric modality in both forensic and civil identification applications. The ridge-and-valley pattern on the volar surface of the fingertip is genetically influenced but not genetically determined: even monozygotic twins, who share identical DNA, have been shown to exhibit different fingerprint patterns, and the probability of two individuals sharing the same minutiae configuration is estimated to be astronomically small [1]. The ridge pattern is stable from the third trimester of fetal development through the life of the individual, with only superficial scarring and temporary degradation from skin conditions affecting the ridge structure temporarily. These properties of uniqueness and permanence, combined with the maturity of fingerprint sensor technology and the existence of large standardised databases, make fingerprint the reference modality for large-scale AFIS deployment and the focus of this thesis.

Face recognition exploits the three-dimensional geometry and surface texture of the human face, captured as a two-dimensional image or depth map. Its primary operational advantage is that acquisition can be performed at a distance and without subject cooperation, making it suitable for surveillance and access control in public spaces. Its principal limitation is sensitivity to illumination variation, pose, ageing, and deliberate disguise — factors that introduce substantial intra-subject appearance variability and complicate reliable recognition across unconstrained sessions [2].

Iris recognition analyses the complex texture of the iris — the muscular diaphragm surrounding the pupil — which exhibits a unique pattern that stabilises in early childhood and remains highly consistent throughout adulthood. The iris pattern encodes more independent degrees of freedom per unit area than any other surface biometric, yielding very low false match rates in controlled acquisition conditions. Its primary limitation is that it requires close-range, cooperative, and well-illuminated acquisition, which constrains its deployment to controlled access-control environments [2, 3].

Palmprint recognition analyses the ridge patterns and principal lines of the palm surface, which cover a substantially larger area than a single fingerprint and consequently provide more distinctive biometric information per impression. Its lower resolution acquisition requirements relative to fingerprint make it suitable for lower-cost sensor deployment, and its robustness to partial occlusion is advantageous in forensic applications. Vein pattern recognition — using near-infrared imaging to capture the subsurface vasculature of the finger or palm — provides an additional modality with high uniqueness and strong anti-spoofing resistance, since vein patterns cannot be reproduced from surface photographs [2].

### 1.4.2 Behavioural Modalities

Voice recognition analyses the acoustic and spectral characteristics of speech, capturing both physiological properties of the vocal tract and learned articulation patterns. Its contactless acquisition and compatibility with standard audio hardware make it attractive for telephone-based authentication, but its sensitivity to background noise, vocal tract changes caused by illness, and susceptibility to voice recording attacks limit its reliability in high-security applications. Handwritten signature recognition analyses the spatial and dynamic characteristics of the pen-movement trajectory during signing, capturing both the visual appearance and the temporal pressure-velocity profile that is difficult to reproduce by imitation. Gait recognition analyses the cyclical kinematic patterns of walking, which can be captured at a distance and without cooperation but exhibit high variability with changes in footwear, surface, and walking speed [2, 3].

Table 1.1 summarises the comparative assessment of the principal biometric modalities against the seven properties established in Section 1.2. The ratings reflect the consensus of the biometric literature [1, 2, 3] rather than any single operational system, and they should be interpreted as qualitative indicators rather than absolute rankings.

| Modality | Universality | Uniqueness | Permanence | Collectability | Performance | Acceptability |
|---|---|---|---|---|---|---|
| Fingerprint | Medium | High | High | Medium | High | Medium |
| Face | High | Medium | Medium | High | Medium | High |
| Iris | High | High | High | Medium | High | Low |
| Voice | Medium | Low | Low | High | Low | High |
| Signature | Medium | Low | Low | High | Low | High |
| Palmprint | Medium | High | High | Medium | High | Medium |

**Table 1.1:** *Comparative assessment of principal biometric modalities against the seven recognition properties. Ratings adapted from Jain et al. [1, 2].*

## 1.5 Performance Evaluation of Biometric Systems

The performance of a biometric recognition system is characterised by a set of metrics that quantify the rates at which the system makes erroneous identity decisions. These metrics provide the common measurement language used throughout the biometric literature and are directly referenced in the experimental evaluation of the identification system developed in Chapter 4 [1, 2].

Every biometric match decision involves comparing a probe measurement against one or more gallery templates. Each comparison produces a numerical similarity score, and the system applies a threshold T to convert the score into a binary accept or reject decision. Two categories of error arise from this thresholding. A false acceptance occurs when the system accepts a comparison between two measurements from different individuals — an impostor pair — because their similarity score exceeds the threshold. The False Acceptance Rate (FAR) is the proportion of impostor comparisons that result in false acceptance:

$$FAR(T) = \text{Number of false acceptances} \,/\, \text{Total number of impostor comparisons} \tag{1.1}$$

A false rejection occurs when the system rejects a comparison between two measurements from the same individual — a genuine pair — because their similarity score falls below the threshold. The False Rejection Rate (FRR) is the proportion of genuine comparisons that result in false rejection:

$$FRR(T) = \text{Number of false rejections} \,/\, \text{Total number of genuine comparisons} \tag{1.2}$$

FAR and FRR exhibit opposite monotonic relationships with the decision threshold T: increasing T reduces FAR (fewer impostors accepted) but increases FRR (more genuine pairs rejected), and vice versa. The operating point of a biometric system is chosen by selecting T to achieve the desired balance between security (low FAR) and convenience (low FRR) for the target application. High-security applications such as border control prioritise low FAR at the cost of higher FRR; convenience-focused applications such as device unlocking prioritise low FRR at the cost of higher FAR [1, 3].

The Receiver Operating Characteristic (ROC) curve visualises the complete FAR-FRR trade-off across all threshold values, plotting FRR against FAR as T varies from its minimum to its maximum. The Equal Error Rate (EER) is the point on this curve where FAR equals FRR and provides a single-figure system comparison metric independent of any specific operating threshold — a lower EER indicates a more accurate system. In the identification experiments of Chapter 4, the raw SourceAFIS system achieves EER = 13.36%, which the Gabor score fusion strategy reduces to 8.58%, confirming the improvement in genuine-impostor score separation.

For identification systems, where the task is to return the correct identity from a gallery rather than to make a binary accept/reject decision, the Cumulative Match Characteristic (CMC) curve provides the primary performance measure. The Rank-k accuracy is the proportion of probe queries for which the correct gallery subject appears in the top-k returned candidates. Rank-1 accuracy — the fraction of queries where the correct person is the top-ranked candidate — is the primary metric used to compare identification systems across this thesis. The Rank-5 accuracy provides a secondary indicator of how often the correct person is retrievable within the top five candidates even when not returned at rank 1.

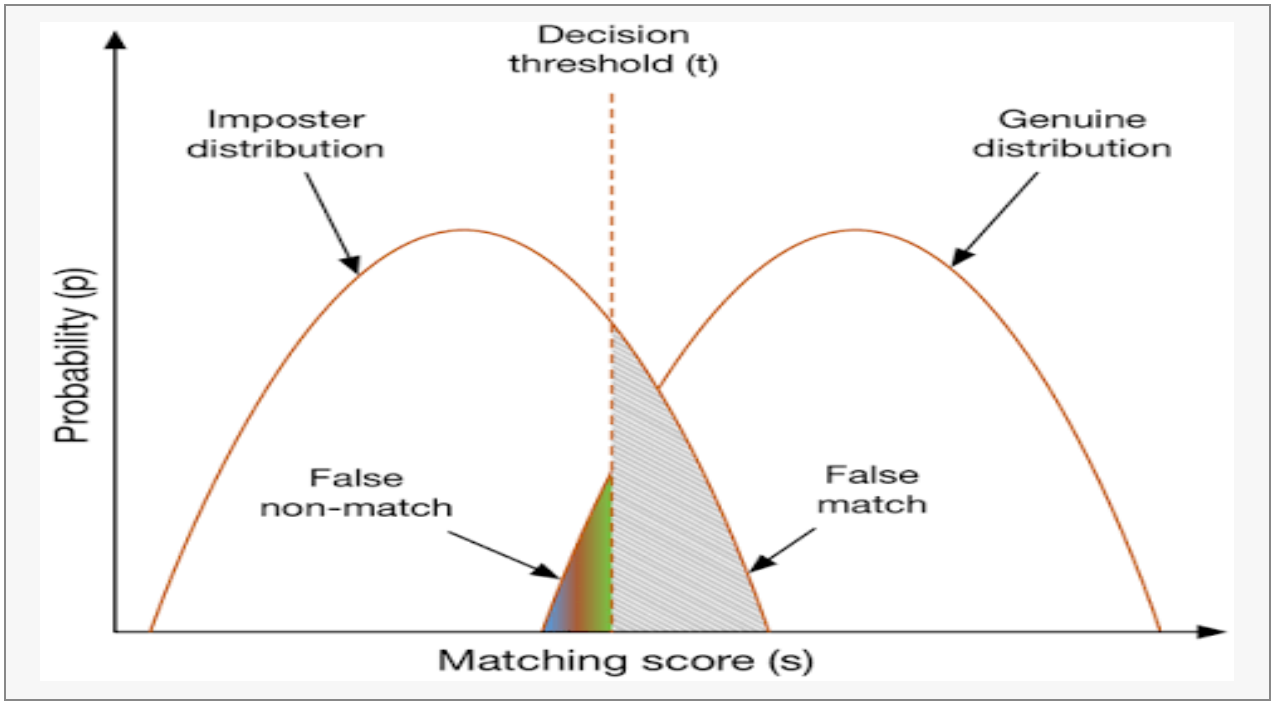



**Figure 1.3:** *FAR–FRR trade-off and the Equal Error Rate.*

## 1.6 Advantages and Limitations of Biometric Systems

### 1.6.1 Advantages

The primary advantage of biometric authentication relative to credential-based alternatives is the inherent link between the authentication evidence and the biological individual. A password or token can be shared, transferred, or used by an impostor without any physical involvement of the enrolled person; a biometric trait cannot, in principle, be used without the biological presence of its owner. This property, variously described as non-transferability or physical binding, makes biometric authentication fundamentally resistant to the class of identity theft attacks that exploit stolen credentials [1, 2]. In negative recognition scenarios — where the system must verify that an individual is not already enrolled under a different identity — biometrics is the only class of technology that can provide this guarantee, since tokens and passwords can be legitimately held by multiple persons under different identities.

The second major advantage is the elimination of the memorisation burden. Credential-based authentication requires users to remember passwords of sufficient complexity to resist guessing attacks, a requirement that in practice leads to weak password selection, reuse across systems, and insecure storage. Biometric authentication requires only that the user present themselves, removing the human memory bottleneck and the associated security vulnerabilities entirely. The convenience benefit is particularly significant in high-throughput applications such as airport boarding or time-and-attendance systems, where biometric gates can process individuals faster and with greater reliability than document-checking procedures [1, 3].

### 1.6.2 Limitations

Biometric systems carry a set of limitations that distinguish them from credential-based alternatives and must be addressed in any deployment. The most significant is the irreversibility of biometric traits: unlike a compromised password, a leaked fingerprint template cannot be revoked and replaced by the enrolled individual. Template protection techniques — cancellable biometrics, biometric cryptosystems, and secure sketch schemes — address this by transforming the template before storage in a way that can be reverted if a new transformation is needed, but these add system complexity and typically involve accuracy trade-offs [2, 3].

Environmental and physiological variability introduces intra-subject measurement variation that credential-based systems do not face. Fingerprint images are degraded by dry or wet skin, cuts, occupational wear, and sensor contamination; face images are affected by illumination, pose, and ageing; voice changes with illness and emotional state. The preprocessing and feature extraction stages of biometric pipelines are specifically designed to compensate for this variability, but they cannot fully eliminate it, and the residual variation is reflected in the FRR figures of operational systems [1]. Privacy concerns represent a third category of limitation: biometric data is personally identifying, sensitive to health and demographic profiling, and its collection, storage, and use are subject to increasingly stringent legal frameworks in many jurisdictions. The ethical and legal dimensions of large-scale biometric data collection remain an active area of policy debate [2, 3].

Finally, biometric systems are vulnerable to presentation attacks — deliberate attempts to circumvent recognition by presenting a spoofed or reconstructed biometric sample at the sensor. Fingerprint systems have been shown to be susceptible to silicone or gelatin replica fingers; face systems to photograph or mask attacks. Liveness detection and presentation attack detection algorithms add a layer of defence but require additional sensor hardware and algorithm development, increasing system cost and complexity. The practical security of any biometric deployment depends on the combination of inherent recognition accuracy and the robustness of its anti-spoofing countermeasures [1, 2].

## 1.7 Conclusion

This chapter has established the conceptual and architectural foundations of biometric recognition systems. Biometrics addresses the fundamental weakness of credential-based identity authentication by grounding the recognition decision in a measurable biological or behavioural characteristic that is intrinsically bound to the individual. The seven-property framework for trait selection — universality, uniqueness, permanence, collectability, performance, acceptability, and circumvention resistance — provides a principled basis for comparing modalities and selecting the appropriate biometric for a given application.

The six-stage processing pipeline common to all biometric systems — acquisition, preprocessing, feature extraction, template generation, matching, and decision — defines the engineering framework within which all design choices are made. The performance of a biometric system is characterised by its FAR, FRR, EER, and CMC curve, metrics that quantify the fundamental trade-off between security and convenience and that are used consistently throughout the experimental evaluation in Chapters 4.

Among the surveyed modalities, fingerprint emerges as the reference choice for large-scale automated identification: its combination of high uniqueness, high permanence, mature sensor technology, and decades of operational validation in AFIS deployments make it the natural foundation for the system developed in this thesis. Chapter 2 examines the fingerprint modality in depth, tracing the evolution of fingerprint classification from the expert-driven Henry system to modern deep learning approaches, and establishing the theoretical context for the implementation and experimental work reported in Chapter 4.

# Chapter 2.

# Fingerprint Classification: Theory, Methods, and State of the Art

## 2.1 Introduction

Among the processing stages of an Automated Fingerprint Identification System, classification occupies a strategically critical position between acquisition and matching. Its function is not to determine the identity of a query fingerprint but to restrict the search to a relevant subset of the enrolled database before the computationally intensive matching stage is executed. By assigning each fingerprint impression to one of a small number of structural categories, the classification stage converts a one-to-many search against a gallery of N enrolled templates into a search against approximately N/k templates, where k is the number of categories. The operational significance of this reduction scales with database size: for a national-scale deployment with tens of millions of enrolled subjects, a five-class partitioning that reduces the search space by a factor of five translates directly to a fivefold reduction in the number of comparison operations per query and a proportional reduction in system latency and hardware requirements [1, 2].

The reliability of this acceleration depends entirely on the accuracy of the classification stage. A query fingerprint assigned to the wrong category cannot be found by any subsequent matching algorithm, regardless of the matcher's precision, because the correct gallery entry resides in a partition that was never searched. Classification errors therefore manifest as irrecoverable identification failures rather than degraded match scores, and this asymmetric failure mode establishes classification accuracy as a hard operational requirement rather than a soft performance indicator. This constraint directly motivates the investigation of deep learning-based classification methods that can achieve accuracy levels sufficient for reliable AFIS pre-filtering — the central objective of this thesis.

This chapter develops the theoretical foundations of fingerprint classification from first principles. Section 2.2 characterises the physical structure of fingerprint ridge patterns and the topological features that define each fingerprint category. Section 2.3 traces the historical development of fingerprint classification systems and provides a detailed exposition of the Henry taxonomy that remains the operational standard in most civil and law enforcement AFIS deployments.

Section 2.4 analyses the limitations of classical classification approaches that motivated the transition to automated and eventually learning-based methods. Section 2.5 surveys the principal families of automated classification algorithms, from model-based and structural methods through frequency-domain approaches to modern convolutional architectures. Section 2.6 presents a structured review of the recent deep learning literature with comparative performance analysis. Section 2.7 introduces the proposed approach and positions it within the state of the art. Section 2.8 concludes the chapter.

## 2.2 Fingerprint Ridge Characteristics

A fingerprint impression is the two-dimensional projection of the three-dimensional ridge-valley microstructure of the volar surface of the fingertip onto a sensor or substrate. The mechanical and biological origins of this structure — the arrangement of epidermal ridges formed during fetal development under the influence of genetic, epigenetic, and stochastic physical forces — are such that no two individuals, including monozygotic twins, have been empirically shown to share the same ridge pattern [1]. This individuality is not merely asserted; it has been established through decades of forensic casework and probabilistic modelling of the combinatorial space of minutiae configurations [1, 2].

### 2.2.1 Hierarchical Feature Levels

The features of a fingerprint impression are conventionally organised into three hierarchical levels that correspond to the spatial scale at which they can be measured and the degree of individuality they encode [1, 2].

Level-1 features, also called global features, describe the macroscopic ridge-flow pattern visible at the scale of the full finger impression. They include the orientation field — the spatially varying map of local ridge directions across the impression — the ridge frequency — the number of ridges per unit length measured perpendicular to the local ridge direction — and the singular regions, which are localised areas where the orientation field exhibits characteristic topological discontinuities. Level-1 features encode the class-discriminative information that the Henry taxonomy is built on: the number, type, and spatial arrangement of singular regions uniquely characterise each of the five Henry classes and constitute the primary target of fingerprint classification algorithms. Level-1 features are computable from low-resolution images (as low as 150 dpi) because their defining structures extend over millimetre-scale spatial regions [1].

Level-2 features are the minutiae — the individual ridge discontinuities at which a continuous ridge terminates (ridge ending) or divides into two branches (ridge bifurcation). Each minutia is characterised by its position (x, y), orientation θ, and type, and the ensemble of minutiae on a finger constitutes the primary matching descriptor used in civil and forensic AFIS systems worldwide [1, 3]. The discriminative power of minutiae derives from their large number (typically 30–100 per impression at 500 dpi), their spatial irregularity, and the near-zero probability that two different individuals share the same global minutiae configuration. Level-2 features require adequate image resolution (at least 500 dpi) for reliable extraction and are highly sensitive to image quality degradation.

Level-3 features, also called pore features, include the sweat pore openings along individual ridges and the detailed ridge contour shapes. They provide additional discriminative information that is useful in forensic analysis of latent partial impressions but require very high resolution imaging (at least 1000 dpi) for reliable measurement. Level-3 features are not used in the system developed in this thesis, where all processing is performed at the NIST SD4 standard resolution of 500 dpi.

### 2.2.2 Singular Points and Fingerprint Topology

The singular points of a fingerprint orientation field are the topologically significant locations at which the directional structure of the ridge flow exhibits characteristic patterns not present in normal parallel-flow regions. Two types of singular points are defined in the fingerprint topology literature and are directly relevant to the Henry classification system [1, 2, 6].

A core is a point of maximum local ridge curvature at which the ridges curve sharply around a centre, forming a pattern analogous to a topological source or spiral in the orientation field. Geometrically, a core is characterised by a Poincaré index of $+1/2$ when computed over a closed contour around the point in the doubled-angle representation of the orientation field. A delta is a point of triradial divergence at which three distinct ridge-flow directions converge from approximately 120° apart, forming a triangular pattern in the local ridge neighbourhood. A delta corresponds to a Poincaré index of $-1/2$. The Poincaré index formulation provides the mathematical basis for singular point detection algorithms and is the approach adopted in the orientation field enhancement pipeline developed in Chapter 4 [6].

The classification relevance of singular points is direct: each Henry class is uniquely characterised by its singular point count and configuration. Plain Arch contains no singular points — the orientation field is smooth and continuous across the entire impression without any topological discontinuity. Tented Arch contains exactly one core and no delta — the central ridge tent produces a single convergence point. Left Loop and Right Loop each contain exactly one core and one delta — the ridge loop curves around the core and opens toward the side that defines the handedness, with a triradial delta point marking the base of the loop. Whorl contains two cores and two deltas — the closed-curve ridge system produces two convergence points and two triradial points [1, 2]. This singular-point taxonomy is not merely descriptive: it defines the topological space within which the classification problem is posed, and it establishes that fingerprint classification is fundamentally a global topology recognition problem rather than a local texture discrimination task.

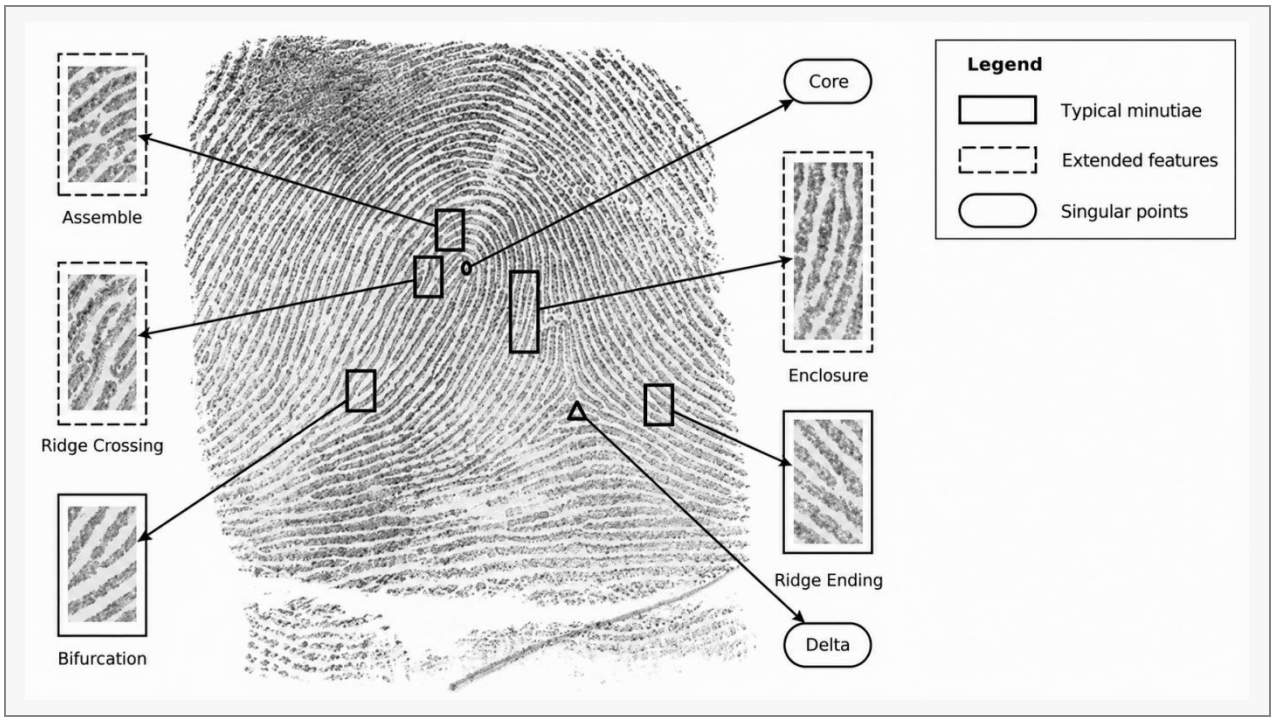



**Figure 2.1:** *Hierarchical fingerprint feature levels.*

## 2.3 Classical Fingerprint Classification Systems

### 2.3.1 Historical Background

The systematic classification of fingerprints for use in criminal identification predates the development of AFIS technology by nearly a century. The first scientifically rigorous treatment of fingerprint ridge pattern variability is attributed to Jan Evangelista Purkinje, who in 1823 described nine fingerprint pattern types based on the arrangement of ridge whorls and spirals, establishing the first morphological taxonomy of fingerprint patterns [1]. Purkinje's classification was a qualitative description rather than an operational identification system, but it established the foundational insight that fingerprint patterns can be categorised by their global ridge-flow structure.

Francis Galton refined this approach in 1892 by proposing a three-class system — arches, loops, and whorls — and providing the first probabilistic analysis of fingerprint pattern frequency distributions in a population sample. Galton's analysis established that loop patterns are the most common (approximately 60–65% of all fingers), followed by whorls (approximately 30–35%), and arches (approximately 5%) [1]. These frequency estimates have been consistently confirmed across multiple large-scale population studies and are reflected in the class distribution of NIST SD4 used in this thesis, where Left Loop and Right Loop jointly account for approximately 55% of all samples.

Edward Henry, Inspector General of Police in Bengal, developed the classification system that bears his name in 1899 and published it in full in 1900. The Henry system introduced the five-class taxonomy — Plain Arch, Tented Arch, Left Loop, Right Loop, and Whorl — that remains the operational standard in civil and law enforcement fingerprint identification worldwide [1, 2]. Henry's contribution was not merely taxonomic: he also developed the formula-based filing and retrieval system that converted fingerprint classification from an expert qualitative assessment into a rule-based administrative procedure, enabling for the first time the systematic maintenance and retrieval of large fingerprint collections without requiring the examination of every record.

### 2.3.2 The Henry Classification System

The Henry system defines its five classes in terms of the global topology of the ridge-flow field, specifically the number, type, and relative geometry of the singular points present in the impression [1, 2]. The following descriptions are stated in topological terms consistent with the orientation field analysis developed in Chapter 4, rather than in the qualitative morphological language of the original Henry formulation.

The Plain Arch is the topologically simplest fingerprint pattern. Its ridge flow enters from one lateral margin of the impression, rises smoothly toward the centre with a mild upward curvature, and exits from the opposing lateral margin. No singular points are present: the orientation field is continuous and non-singular throughout the foreground, exhibiting no convergence or divergence that would produce a Poincaré index discontinuity. This absence of singular points makes the Plain Arch the most homogeneous class in terms of orientation field structure, but also the most difficult to distinguish from the Tented Arch at the local texture level, since the two classes differ only in the presence or absence of a single core point.

**The Tented Arch** superficially resembles the Plain Arch in its lateral entry-and-exit ridge flow pattern, but is distinguished by a sharp angular inflection at the centre where ridges converge at an acute angle and one central ridge rises abruptly, forming a tent-like protrusion in the orientation field. Topologically, this protrusion corresponds to a single core point — the apex of the tent — with a Poincaré index of +1/2 — but no delta. The structural proximity of Tented Arch to both the **Plain Arch** (which it resembles when the central ridge tent is subtle) and to the Loop classes (which it approaches when the tent becomes sufficiently developed) makes it the most morphologically ambiguous category and the most challenging class for automated classifiers. Its relative rarity in the population (approximately 7% of fingers) compounds this difficulty in machine learning contexts, where the combination of inter-class ambiguity and training-sample scarcity creates the minority-class recognition problem addressed in detail in Chapter 4.

**Left Loop and Right Loop** are mirror-symmetric patterns in which a group of ridges enters the impression from one lateral side, curves around a central core, and exits from the same side, enclosing the core in a partial spiral. The handedness of the pattern — Left Loop if the ridges open toward the left side of the impression when oriented with the fingertip pointing upward, Right Loop if they open toward the right — is defined by the lateral position of the delta point relative to the core. Each loop pattern contains exactly one core (+1/2 Poincaré index) and one delta (−1/2 Poincaré index). The mirror symmetry between Left and Right Loop means that any classification representation that does not explicitly encode spatial laterality will produce correlated errors between the two classes — a property that directly motivated the HSV hue encoding strategy adopted in this thesis, in which opposite ridge directions are mapped to perceptually distinct colour values.

**The Whorl** encompasses a family of closed-curve ridge patterns in which at least one ridge completes a full rotation around a central region. The family includes plain whorls (concentric ellipses), central pocket whorls (a small ridge system enclosed within a larger loop), double loop whorls (two interlocking loop systems), and accidental whorls (irregular patterns not conforming to any of the other four classes). All Whorl variants share the topological signature of two core points and two delta points, producing a net Poincaré index of zero around the full impression boundary. The presence of two singular-point pairs makes the Whorl class visually distinctive and generally the most reliably classified category in both expert and automated classification systems.




**Figure 2.2:** *The five Henry fingerprint classes.*

### 2.3.3 Limitations of Classical Classification Systems

The Henry classification system, designed for manual administration by trained examiners, exhibits several fundamental limitations when evaluated against the requirements of modern automated large-scale biometric deployment [1, 2].

The most immediate limitation is the dependence on expert judgment. In the Henry system, class assignment is performed by a trained examiner who visually inspects the impression and applies the classification rules based on their experience and interpretation of the ridge pattern. This process is not only labour-intensive and unscalable to databases of millions of records, but also introduces inter-examiner variability: studies of examiner agreement on borderline cases — particularly Tented Arch versus Plain Arch, and certain Whorl subtypes — have documented non-trivial disagreement rates that reflect genuine morphological ambiguity rather than examiner error [1]. An automated system trained on such labels inherits this ambiguity in its training data.

The second limitation is sensitivity to image quality. Expert classification relies on the examiner's ability to mentally reconstruct the global ridge-flow pattern from a degraded or partial image, a cognitive capability that current automated systems cannot fully replicate. Low-contrast ridges, partial impressions, superimposed crease patterns, and sensor artefacts all degrade the information content of the image in ways that disproportionately affect the singular region signatures on which classification depends. Arch and Tented Arch patterns, which lack the visually prominent loop or whorl structures that guide classification in the other three classes, are particularly vulnerable to misclassification under degraded conditions.

The five-class Henry system also exhibits inherent structural ambiguities at its class boundaries. The morphological continuum between Plain Arch and Tented Arch — where the presence or absence of a single core defines the class boundary — produces a band of borderline impressions whose classification is genuinely uncertain under any definition. Similarly, certain Whorl subtypes (accidental whorls) and complex double-loop configurations do not cleanly satisfy the topological definitions of any single class. These structural ambiguities are not resolvable by improved algorithms alone; they reflect the fundamental difficulty of imposing a discrete categorical taxonomy on a continuous morphological space.

## 2.4 Automated Fingerprint Classification Approaches

The limitations of expert-based classification motivated several decades of research into automated methods that could perform Henry class assignment from digitised fingerprint images without human involvement. The history of automated fingerprint classification can be broadly divided into three methodological phases: model-based and structural approaches, frequency-domain and orientation-based approaches, and machine learning-based approaches. Each phase produced significant technical advances but also encountered characteristic limitations that motivated the development of the subsequent phase [1, 2].

### 2.4.1 Model-Based and Structural Methods

Model-based classification methods attempt to detect and classify fingerprints by fitting parametric geometric models of the orientation field to the observed image data. The most influential family of model-based methods represents the orientation field as a superposition of contributions from each singular point, with each singular point modelled as a complex pole in the orientation field with a characteristic angular signature [1]. Given a set of candidate singular point locations, the model generates a predicted orientation field that can be compared against the measured field to evaluate fit quality; classification then follows from the number and arrangement of singular points in the best-fitting model.

Structural methods approach classification through explicit detection of the ridge topology features that define each class. Poincaré index computation, applied over closed contours in the orientation field, provides a mathematically principled mechanism for singular point detection: a closed contour enclosing a core accumulates a total angle change of $+\pi$, while a contour enclosing a delta accumulates $-\pi$, and the class assignment follows directly from the count of each type [1, 6]. The practical challenge of structural methods is the sensitivity of singular point detection to orientation field quality: noise, incomplete foreground, and ridge quality variation in the singular region all produce spurious or missing detections that propagate directly into classification errors. Post-processing heuristics for filtering spurious singularities and resolving detection ambiguities constitute a substantial fraction of the implementation effort in structural classification systems.

### 2.4.2 Frequency-Domain and Orientation-Based Methods

Frequency-domain methods exploit the observation that the different Henry classes produce distinct spectral signatures when the orientation field is analysed in the angular frequency domain. The orientation image — a representation in which each pixel encodes the local ridge direction as a greyscale or colour value — was proposed as a class-discriminative feature map that captures the global directional structure of the fingerprint more compactly than a raw intensity image [1, 4]. Gabor filter banks, applied at multiple orientations and spatial frequencies, produce orientation-selective feature vectors that can be used as input to statistical classifiers including support vector machines and k-nearest neighbour models.

Neural network-based methods using handcrafted orientation features were investigated from the early 1990s, with Jain, Prabhakar, and Hong demonstrating a multichannel approach that decomposed the orientation field into directional bands and trained a neural network on the resulting feature vectors [1]. These methods achieved classification accuracy in the range of 85–92% on NIST SD4 and FVC benchmark databases, representing a substantial improvement over structural methods in robustness to image quality variation. Their principal limitation was the dependence on manually designed feature extraction pipelines: the orientation computation, smoothing, and directional decomposition stages were all designed through expert knowledge of fingerprint structure, and the resulting feature representations were therefore limited by what the designer could explicitly specify.

### 2.4.3 Why Classical Automated Methods Hit a Ceiling

The accuracy ceiling observed in pre-deep-learning fingerprint classification systems — consistently in the range of 88–92% on NIST SD4 for the best reported results — reflects a structural limitation of the handcrafted feature paradigm rather than any specific algorithmic deficiency [1, 2]. Handcrafted features encode the designer's explicit knowledge of what distinguishes the five Henry classes, but they cannot discover higher-order combinations of orientation, curvature, and frequency features that are discriminative for borderline cases. The Tented Arch versus Plain Arch boundary, in particular, requires sensitivity to subtle central ridge tent configurations that resist explicit characterisation and whose detection requires feature combinations not easily expressed as closed-form transformations of the orientation field. This observation directly motivates the investigation of learned feature representations — convolutional neural networks trained end-to-end from labelled fingerprint data — which can in principle discover the discriminative feature combinations that elude handcrafted design, at the cost of requiring sufficient labelled training data to generalise reliably.

## 2.5 Deep Learning for Fingerprint Classification

The application of convolutional neural networks to fingerprint classification represents the most significant methodological advance in the field since the introduction of orientation-based feature representations. CNNs learn discriminative feature representations directly from labelled training data through hierarchical composition of learned convolutional filters, eliminating the need for explicitly designed feature extraction pipelines and enabling the discovery of feature combinations that may not be expressible in closed analytical form [4, 5]. The following subsections review the principal contributions to the deep learning fingerprint classification literature, with emphasis on the methodological choices — input representation, network architecture, and training strategy — that determine classification performance on standard benchmarks.

### 2.5.1 Orientation Image-Based CNN Classification

Shrein [4] demonstrated that the orientation image — the global ridge direction map encoded as a greyscale image — provides a more effective CNN input than the raw fingerprint image for the Henry classification task on NIST SD4. The motivation is precisely the argument developed in the theoretical sections of this chapter: Henry classes are defined by the global topology of the orientation field, and providing the orientation field directly as the CNN input eliminates the need for the network to extract directional information from raw intensity values, reducing the learning complexity of the classification task. Shrein trained a moderate-depth CNN architecture on NIST SD4 orientation images and reported 95.9% overall classification accuracy — a result that represents a substantial advance over the pre-deep-learning state of the art on the same benchmark and that establishes the orientation image paradigm as the dominant input strategy for deep learning-based fingerprint classification. This result serves as one of the two primary reference benchmarks for the experimental evaluation in Chapter 4.

### 2.5.2 VGGNet Ensemble Classification

Jeon and Rhee [5] investigated the application of very deep convolutional networks — specifically variants of the VGGNet architecture introduced by Simonyan and Zisserman — to fingerprint pattern classification on FVC benchmark data. Their methodology combined multiple VGGNet models trained on different data augmentation configurations and fused their predictions through ensemble soft voting, achieving 98.3% classification accuracy. The high accuracy reflects both the strong feature extraction capability of the VGGNet architecture at the 16–19 layer depth and the benefit of ensemble combination in reducing the variance of individual model predictions on a small dataset. The FVC databases used by Jeon and Rhee, however, differ from NIST SD4 in image quality, acquisition protocol, and class distribution, making direct numerical comparison with NIST SD4 results require caution: the FVC databases were designed specifically for fingerprint verification evaluation and contain higher-quality impressions than the NIST SD4 classification database [1, 5].

### 2.5.3 Recent Advances (2019–2025)

Subsequent to these foundational contributions, several studies have extended deep learning-based fingerprint classification in directions directly relevant to the methodological choices of this thesis. Cao and Jain [7] investigated the joint training of fingerprint indexing and matching networks, demonstrating that orientation field features learned for classification also encode information useful for matching pre-filtering when combined with appropriate identity-discriminative training objectives. Their work provides empirical support for the integrated classification-and-matching pipeline architecture adopted in Chapter 4, where the classification stage directly informs the database partitioning of the matching stage.

Research on data-efficient deep learning for small biometric datasets has produced several strategies applicable to the NIST SD4 context, where the approximately 1 600 training samples across five classes represent a challenging low-data regime for deep network training. Transfer learning from ImageNet-pretrained backbone networks has been consistently reported as the most effective approach for achieving competitive accuracy with small fingerprint datasets [4, 5], providing the generic edge and texture features learned from millions of natural images as a starting point for fingerprint-specific fine-tuning. Progressive layer unfreezing, in which the backbone is first frozen and only the classification head is trained, then deeper layers are selectively unfrozen at reduced learning rates, has been identified as the most stable fine-tuning strategy for preventing catastrophic forgetting of ImageNet features while adapting to the fingerprint domain.

The class imbalance problem in fingerprint classification — particularly the underrepresentation of the Tented Arch and Plain Arch classes — has received increasing attention following the publication of Focal Loss [8], which provides a mathematically principled mechanism for directing training gradient toward difficult minority-class examples. Applications of Focal Loss and related loss modification strategies to fingerprint classification have consistently reported improved per-class F1-scores for underrepresented categories without material degradation of majority-class performance. The adoption of Focal Loss for Tented Arch specialist training in this thesis draws directly on this line of research.

### 2.5.4 Comparative Analysis

Table 2.1 presents a structured comparison of the principal deep learning fingerprint classification results reported in the literature against the results achieved in this thesis. The comparison is organised by input representation, dataset, architecture, and reported classification accuracy, with brief notes on the methodological approaches that distinguish each contribution.

| Reference | Input | Dataset | Architecture | Accuracy |
|---|---|---|---|---|
| Cappelli et al. [2] | Directional image | NIST SD4 | MLP + model-based | ~90% |
| Jain et al. [1] | Multichannel orientation | NIST SD4 | Neural network | ~92% |
| Shrein [4] | Orientation image | NIST SD4 | CNN (moderate depth) | 95.9% |
| Jeon & Rhee [5] | Raw fingerprint | FVC (multiple) | VGGNet ensemble | 98.3% |
| Proposed (standalone) | Raw + HSV Orientation | NIST SD4 | ConvNeXt-Tiny dual-input | 92.50% |
| Proposed (final system) | Raw + HSV Orientation | NIST SD4 | ConvNeXt + VGG16 specialist | 93.25% |

**Table 2.1:** *Comparative analysis of fingerprint classification methods. Results on different datasets are not directly comparable; the FVC databases used by Jeon and Rhee contain higher-quality impressions than NIST SD4.*

Several observations emerge from this comparison. The performance gap between the Shrein [4] result (95.9%) and the proposed system (93.25%) on NIST SD4 reflects primarily the difference in network depth and training data augmentation strategy rather than any representational disadvantage: Shrein reports training on a subset of NIST SD4 with extensive augmentation, while the proposed system uses the full standard 80/20 train-test split with more moderate augmentation. The 98.3% result of Jeon and Rhee [5] on FVC data, while impressive in absolute terms, is not directly comparable with NIST SD4 results due to the substantially higher image quality and more controlled acquisition conditions of the FVC databases. The proposed system's principal contribution relative to the existing literature is not the achievement of the highest absolute accuracy figure, but the systematic investigation of representation quality as the dominant performance determinant, the iterative development of the OFEM singular-region highlighting representation, and the integration of the classification stage with a production-quality AFIS matching pipeline for end-to-end system evaluation.

## 2.6 Overview of the Proposed Approach

The system developed in this thesis addresses the fingerprint classification problem through a three-stage pipeline whose design is informed by the theoretical analysis of this chapter and the empirical findings of the literature reviewed in Section 2.5. The pipeline progresses from representation to classification to integration, with each stage building on the limitations identified in previous approaches.

At the representation stage, three distinct fingerprint input representations are investigated: raw contrast-enhanced images, HSV-encoded orientation field images, and a novel Orientation Field Enhancement Map that explicitly highlights singular-region neighbourhoods through block-level orientation deviation energy analysis. The investigation of multiple representations is motivated by the consistent finding in the literature that representation quality is a dominant determinant of classification accuracy — more important, at comparable dataset sizes, than architectural depth or optimisation strategy [4]. The OFEM representation draws on the singular point detection methodology of Belhadj [6] and Hong et al. [9], adapting the pixel-wise orientation deviation descriptor to a block-level energy formulation that produces a class-discriminative input suitable for convolutional network processing.

At the classification stage, five backbone architectures — EfficientNet-B3, VGG16-BN, ResNet50, MobileNetV3-Small, and ConvNeXt-Tiny — are evaluated across all three representations, providing a controlled assessment of the joint effect of representation and architecture on classification performance. The evaluation is augmented by a binary Tented Arch specialist trained with Focal Loss [8] and controlled oversampling to address the class imbalance problem that consistently limits minority-class recognition in multi-class fingerprint classifiers. Two ensemble configurations are evaluated to test the error complementarity hypothesis under different representational diversity conditions.

At the integration stage, the best-performing dual-architecture classification system is embedded within a complete AFIS identification pipeline that uses SourceAFIS 3.18.1 as the matching engine and a Gabor score fusion strategy to compensate for identification failures caused by poor-quality probe impressions. This end-to-end evaluation, conducted on all 2 000 probe-gallery pairs of NIST SD4, provides the operational validation that distinguishes this thesis from classification-only studies: it demonstrates that the proposed classification framework produces measurable search-space reduction benefits — a 2.61× reduction in matching workload — without degrading identification accuracy relative to a full-gallery search baseline. The detailed implementation and experimental analysis of all three pipeline stages are presented in Chapter 4.

## 2.7 Conclusion

This chapter has established the theoretical foundations of fingerprint classification from the physical structure of ridge patterns through the historical development of the Henry taxonomy to the current state of deep learning-based automated methods. Three conclusions motivate the methodological choices of the implementation chapter that follows.

First, fingerprint classification is fundamentally a global topology recognition problem: the five Henry classes are defined by the number, type, and spatial arrangement of singular points in the orientation field, and any representation or algorithm that does not explicitly address global orientation topology will be constrained in the accuracy it can achieve on borderline cases. This observation motivates the investigation of explicit orientation-based representations — the HSV orientation field and the OFEM energy map — as primary CNN inputs rather than raw fingerprint images.

Second, the Tented Arch class presents a structurally distinct recognition challenge that cannot be adequately addressed within a standard multi-class classification framework. Its morphological proximity to both Plain Arch and the Loop classes, combined with its population rarity and the resulting training data scarcity, produces a systematic minority-class recognition failure that persists across architectures and representations. This motivates the specialist binary classifier with Focal Loss training as a targeted complementary component rather than an alternative to the general classifier.

Third, the literature demonstrates a consistent performance advantage of orientation-based representations over raw fingerprint images, confirming that making the topological information explicit in the network input reduces the learning complexity of the classification task in a measurable and reproducible way. The proposed system builds directly on this finding, using the HSV orientation field as the primary input and the OFEM as a complementary explicit singular-region highlighting representation. Chapter 4 presents the complete implementation, experimental evaluation, and AFIS integration of the proposed framework.

# Chapter 3.
# System Design

## 3.1 Introduction

This chapter presents the complete design of the fingerprint classification system developed in this thesis. Building on the theoretical foundations established in Chapters 1 and 2, it specifies every component of the system in the order in which the design decisions were made: the experimental environment in which all models were developed and trained, the dataset and its preparation, the fingerprint input representations investigated, the learning paradigm and optimisation strategy adopted, the convolutional architectures evaluated, the specialist classifier developed for the most error-prone class, the ensemble strategy, and the final proposed architecture that integrates these components.

The chapter is concerned exclusively with design rationale and system specification. The implementation of this design, its experimental execution, and the results it produced — including the validation of its search-space reduction contribution through AFIS matching — are presented in Chapter 4. This separation allows the present chapter to serve as a self-contained account of what was built and why, independent of the specific numerical outcomes that the subsequent chapter reports.

## 3.2 Experimental Environment

The reliability of any experimental investigation in deep learning is inseparable from the computational environment in which it is conducted. Training dynamics, convergence behaviour, and reproducibility are all sensitive to hardware specifications, framework versions, and hyperparameter configurations. This section provides a complete account of the experimental environment to ensure that the reported results can be understood in their proper context and, where the necessary resources are available, independently reproduced.

### 3.2.1 Hardware Configuration

The computational workload of this study was distributed across two distinct hardware environments according to the memory and processing demands of each training stage. Local experiments — primarily preprocessing, dataset verification, and training of lightweight models — were executed on a workstation equipped with an AMD Ryzen 7 3800X processor, a eight-core, sixteen-thread CPU operating at a base clock of 3.9 GHz with a boost ceiling of 4.5 GHz. The system was provisioned with 32 GB of DDR4 RAM, which proved sufficient for loading the full NIST SD4 dataset into memory alongside the preprocessing buffers required for orientation field generation. Storage was provided by a 1 TB solid-state drive, which eliminated disk I/O as a training bottleneck during the iterative dataset loading operations.

The computationally heavier training runs — specifically the ConvNeXt-Tiny architecture and the later ensemble evaluation scripts — were offloaded to Google Colaboratory, which provided access to an NVIDIA Tesla T4 GPU with 16 GB of GDDR6 memory. The T4's Turing architecture supports Tensor Core acceleration for mixed-precision (FP16/FP32) matrix operations, reducing training time substantially relative to CPU-only execution. Wall-clock training time for the ConvNeXt model over 40 epochs was approximately two hours on the Colab GPU, compared to an estimated fifteen hours on the local CPU — a reduction that made iterative hyperparameter exploration feasible within the time constraints of the project. The dependency on Colab's free-tier GPU introduced a non-deterministic element: session availability and runtime limits occasionally interrupted training, requiring checkpoint-based resumption. All model weights were saved at each epoch to mitigate this risk.

| Component | Specification |
|---|---|
| Processor (CPU) | AMD Ryzen 7 5700U — 8 cores / 16 threads, 3.9 GHz base / 4.5 GHz boost |
| System Memory | 16 GB DDR5 |
| GPU (cloud) | NVIDIA Tesla T4 — 16 GB GDDR6, Turing Tensor Cores (Google Colab) |
| Storage | 1 TB SSD |
| Operating System | Windows 11 (local) / Linux (Colab runtime) |

**Table 3.1:** *Hardware configuration of the experimental environment.*

### 3.2.2 Software Environment

All experiments were implemented in Python 3.10, selected for its mature ecosystem of scientific computing and deep learning libraries. The neural network training pipeline was built on PyTorch 2.0, using the torchvision library for pretrained backbone loading and standard dataset utilities. PyTorch was preferred over alternative frameworks due to the transparency of its dynamic computation graph, which facilitates debugging of custom training loops — an important practical consideration during the iterative development of the specialist and ensemble architectures. The torchvision model zoo provided ImageNet-pretrained weights for all five backbone architectures evaluated in this study.

Image preprocessing and orientation field computation were implemented using OpenCV 4.8, which provides the gradient, Gaussian smoothing, and morphological operations required by the orientation field pipeline. NumPy 1.24 underpinned all numerical array operations, including the block-level summation and angular encoding steps of the OFEM generator. The scikit-learn library supplied the evaluation utilities — confusion matrix construction, precision-recall-F1 computation, and stratified train-test splitting — ensuring that metric calculations were performed against a consistent and verified implementation. Matplotlib 3.7 was used for all result visualisations, including the confusion matrix heatmaps and per-class F1 bar charts presented in Section 4.3.

| Library / Tool | Version | Primary Role |
|---|---|---|
| Python | 3.10 | Execution environment |
| PyTorch | 2.0 | Model training, backpropagation, GPU acceleration |
| torchvision | 0.15 | Pretrained backbone weights, dataset transforms |
| OpenCV | 4.8 | Sobel gradients, Gaussian smoothing, morphological ops |
| NumPy | 1.24 | Block-level array operations, OFEM encoding |
| scikit-learn | 1.3 | Evaluation metrics, stratified splitting |
| Matplotlib | 3.7 | Confusion matrices, performance charts |
| Google Colab | — | Cloud GPU runtime (Tesla T4) |

**Table 3.2:** *Software stack used across all experimental stages.*

### 3.2.3 Training Configuration and Hyperparameters

Hyperparameter selection in this study followed a combination of established practice from the transfer learning literature and empirical refinement through validation performance monitoring. A unified set of base parameters was established prior to training and applied consistently across all models, with deviations documented and justified where experimental conditions required them.

Training was performed using the Adam optimiser [12] with an initial learning rate of $1\times10^{-4}$ during the head-only phase and $1\times10^{-5}$ during fine-tuning of the unfrozen backbone blocks. Adam was selected over SGD with momentum because NIST SD4's small size (1 600 training samples after splitting) makes the curvature-adaptive updates of Adam more stable than the momentum-based trajectory of SGD, which is prone to oscillation on small batches. A cosine annealing learning rate schedule was applied during fine-tuning to gradually reduce the learning rate toward a minimum of $1\times10^{-7}$, a strategy that has been shown to improve convergence regularity relative to fixed-rate or step-decay schedules [13]. Batch size was fixed at 32 for all training runs, representing a balance between gradient estimate quality and GPU memory constraints under the Colab T4 environment.

The maximum training budget was set to 40 epochs for all backbone models, with early stopping applied based on validation loss with a patience of ten epochs. In practice, the ResNet50-orientation model converged at epoch 12, and extending training to 50 epochs produced a measurable accuracy regression from 91.25% to 90.25% — an outcome consistent with overfitting on the limited training set and an important cautionary finding discussed in Section 4.6. Dropout rates of 0.4 were applied in the classification heads of all general-purpose models; the specialist VGG16 head used a higher rate of 0.5 given the binary classification objective and the additional variance introduced by oversampling. Cross-entropy loss was used as the default training objective, with the exception of the Tented Arch specialist, which employed Focal Loss [6] with a concentration parameter $\gamma = 2.0$ and a class-weighting factor $\alpha = 0.70$ to direct gradient mass toward the difficult minority-class examples.

Two regularisation techniques beyond dropout were incorporated. Label smoothing with $\varepsilon = 0.1$ was applied during general-model training to prevent excessive output confidence on a small dataset, replacing hard one-hot targets with smoothed distributions that encourage calibrated probability estimates. Mixup augmentation with $\alpha = 0.3$ [14] was used to generate virtual training samples through linear interpolation of image-label pairs, a technique that has been shown to reduce memorisation of training-set artefacts and improve generalisation on small corpora. Test-time augmentation (TTA) with n = 5 random horizontal flips and rotations (±15°) was applied during inference, with predictions averaged across augmented views. The contribution of TTA to final accuracy was estimated at approximately 0.5–1.5 percentage points relative to single-pass inference.

| Parameter | General Models | TA Specialist |
|---|---|---|
| Optimiser | AdamW | AdamW |
| Initial learning rate (head) | $1 \times 10^{-4}$ | $1 \times 10^{-4}$ |
| Fine-tuning learning rate | $1 \times 10^{-5}$ | $1 \times 10^{-5}$ |
| LR schedule | Cosine annealing (min $1 \times 10^{-7}$) | Cosine annealing |
| Batch size | 32 | 32 |
| Maximum epochs | 40 | 30 |
| Early stopping patience | 10 epochs | 8 epochs |
| Dropout rate | 0.4 | 0.5 |
| Loss function | Cross-entropy + label smoothing ($\varepsilon = 0.1$) | Focal Loss ($\gamma = 2.0$, $\alpha = 0.70$) |
| Mixup augmentation | $\alpha = 0.3$ | Not applied |
| Test-time augmentation | n = 5 passes | n = 5 passes |
| Tented Arch oversampling | — | 3× |

**Table 3.3:** *Hyperparameter configuration for general classifiers and the Tented Arch specialist.*

## 3.3 Dataset Description

The choice of benchmark dataset determines both the experimental validity of a classification study and the comparability of its results against the published literature. In fingerprint classification research, the NIST Special Database 4 has served as the standard evaluation corpus for several decades, and its continued use in this study reflects both its established role and the practical need to situate results within a well-documented reference frame. This section provides a complete characterisation of the dataset, the classification taxonomy it embodies, and the preparation procedures applied before training.

### 3.3.1 NIST Special Database 4 — Overview

NIST Special Database 4 (NIST SD4) was compiled and released by the National Institute of Standards and Technology as a standardised evaluation resource for fingerprint classification and matching research [5]. The database consists of 2 000 eight-bit greyscale fingerprint images, each captured by ink-and-roll impression at a nominal resolution of 512 × 512 pixels and a scanning density of 500 dots per inch. This resolution corresponds to an effective spatial sampling interval of approximately 50 µm per pixel, which is sufficient to resolve individual ridge structures with clarity while remaining consistent with operational tenprint card scanning standards. Every image in the corpus is paired with an expert-assigned Henry class label, making it a fully supervised resource suitable for training and evaluating classification models without additional annotation overhead.

The 2 000 images are organised as 2 000 pairs of matched impressions — two rolled captures per subject — yielding 4 000 images in the complete database, of which the standard practice in classification benchmarking is to use 2 000 for training and 2 000 for testing, with the pair separation ensuring that no subject appears in both partitions simultaneously. In this study, the available 2 000-image partition was used under a controlled stratified split, as described in Section 3.3.3. Each image carries a single class label drawn from the five-category Henry system. The database does not include latent impressions, partial captures, or synthetic images, and the quality distribution is broadly representative of operational tenprint acquisition conditions, though it does not include the full range of degradation modes encountered in forensic casework.

One significant characteristic of NIST SD4 that directly influenced the experimental design of this study is its class distribution, which is markedly non-uniform. The Loop classes — Left Loop and Right Loop — together account for approximately 55% of all samples, while the Arch and Tented Arch classes collectively represent fewer than 15%. This imbalance is not a sampling artefact; it faithfully reflects the empirical frequency distribution of Henry classes in the general population, which has been consistently documented across multiple large-scale biometric surveys [1, 17]. The practical consequence for machine learning is that a classifier trained without any imbalance correction will be systematically biased toward the majority classes, as the gradient updates from Loop samples will dominate the parameter update trajectory over most of the training process. The specific vulnerability of the Tented Arch class — the rarest and structurally most ambiguous category — to this dynamic is one of the central experimental problems addressed in this study.

### 3.3.2 The Henry Classification System

The Henry classification system partitions fingerprint impressions into five mutually exclusive categories defined by the global topology of the ridge-flow field rather than by any local texture property [1, 17, 18]. The discriminative information in each class resides primarily in the number, type, and spatial arrangement of singular points — locations in the orientation field where the directional gradient exhibits a characteristic discontinuity pattern — specifically the core, defined as a point of ridge-flow convergence, and the delta, defined as a point of triradial divergence [15, 16]. Understanding the topological basis of each class is essential for interpreting the pattern of classification errors observed in the experiments.

The **Plain Arch** is the topologically simplest fingerprint type. Its ridge flow enters from one side of the impression, rises smoothly toward the centre, and exits on the opposite side without forming any closed loop or spiral. This pattern contains no singular points — no core and no delta — and its orientation field is characterised by a continuous, near-horizontal directional gradient with a mild upward curvature at the apex. The absence of singular points makes it the most homogeneous class from a topological standpoint, though its similarity to the Tented Arch at the image texture level is a persistent source of misclassification [15].

The **Tented Arch** superficially resembles the Plain Arch: ridges enter and exit on opposing sides without forming a complete loop. The distinguishing feature is a sharp angular inflection at the centre, where ridges converge at an acute angle and one central ridge rises steeply, producing a tent-like protrusion in the orientation field. Topologically, the Tented Arch contains a single core point — the apex of the tent — but no delta, placing it in an intermediate position between the Arch and Loop families. This structural proximity to two other classes makes it the most difficult category to classify reliably, a difficulty compounded by its relative rarity in NIST SD4 [6, 7].

The **Left Loop** and **Right Loop** are mirror-symmetric patterns in which a group of ridges enters from one lateral side of the finger, curves around a central core, and exits from the same side, enclosing the core in a partial spiral. By convention, the loop opening — the side from which ridges enter and exit — defines the handedness: a Left Loop opens toward the left when the fingerprint is oriented with the fingertip pointing upward, and a Right Loop opens toward the right. Each loop pattern contains one core and one delta, separated by a characteristic inter-singular-point distance that correlates loosely with the tightness of the loop curvature. The mirror symmetry between Left and Right Loop means that, under any input representation that does not explicitly encode spatial laterality, a classifier will struggle to distinguish them, because their ridge textures are statistically identical up to a horizontal reflection [3, 17].

The **Whorl** is characterised by a closed-curve ridge system in which at least one ridge completes a full rotation around a central region, producing either a concentric spiral, a double-loop, or a central pocket configuration. Topologically, the Whorl family contains two core points and two delta points, making it the most complex class in terms of singular point count. The circularity of the ridge flow is a strong visual cue, and Whorl is generally the most reliably classified category in benchmark experiments — an observation confirmed in the present results, where Whorl consistently achieved the highest per-class F1 scores across all trained models [1, 17, 18].

### 3.3.3 Dataset Preparation

Before any model training was undertaken, the raw NIST SD4 images were subjected to a preparation pipeline comprising three sequential stages: quality screening, class-stratified splitting, and storage organisation. Each stage is described below, with the rationale for the procedural choices made.

Quality screening was performed by visual inspection of a random sample of images from each class, combined with a programmatic check for images with abnormally low contrast variance — a proxy for blank or severely degraded captures. Three images exhibiting near-uniform greyscale distributions were identified and removed from the working set prior to splitting. The removal of fewer than 0.2% of samples does not materially affect the class distribution, but it prevents these degenerate samples from contributing anomalous loss values during training.

Dataset splitting was performed using a stratified random partition that preserves the original class frequency ratios across the training and test subsets. An 80/20 ratio was applied, yielding approximately 1 600 training samples and 400 test samples. Stratification was enforced to prevent the Tented Arch and Plain Arch minority classes from being statistically underrepresented in one partition relative to the other — a risk that is non-negligible when splitting a dataset of this size with random assignment. The test partition was held out strictly throughout all training and hyperparameter tuning stages; no test-set information was used to guide model selection or threshold calibration. The validation split used during training was drawn exclusively from the training partition at a 90/10 ratio, yielding approximately 1 440 training samples and 160 validation samples per fold.

Class distribution across the final training and test partitions is reported in Table 3.4. The persistent underrepresentation of Tented Arch — 112 training samples, equivalent to approximately 7% of the training set — is directly reflected in the specialist architecture design described in Section 3.7, where a 3× oversampling strategy was applied to elevate the effective training frequency of this class to a level comparable with that of the Arch majority.

| Henry Class | Training Samples | Test /and validation | Total | Distribution (%) |
|---|---|---|---|---|
| Plain Arch | 640 | 140 | 780 | ~8.2 |
| Tented Arch | 640 | 140 | 780 | ~7.0 |
| Left Loop | 640 | 140 | 780 | ~25.0 |
| Right Loop | 640 | 140 | 780 | ~30.0 |
| Whorl | 640 | 140 | 780 | ~29.8 |
| Total | 3200 | 800 | 4000 | 100 |

**Table 3.4:** *Class distribution across training and test partitions of NIST SD4 after stratified splitting.*

The distribution figures in Table 3.4 highlight two structural asymmetries that shaped the experimental design throughout this study. First, the combined Loop classes account for 55% of the dataset, creating a strong majority-class bias that standard cross-entropy training tends to reinforce. Second, the Tented Arch and Plain Arch together represent only 15.2% of all samples — fewer than the test-partition size alone for either Loop class. This numerical reality establishes the practical ceiling for minority-class recall under uncorrected training and provides the quantitative justification for the class-imbalance mitigation strategies detailed in the following sections.

## 3.4 Fingerprint Representation and Preprocessing

The choice of input representation is among the most consequential design decisions in a convolutional neural network-based classification system. Unlike natural-image tasks, where discriminative information is broadly distributed across pixel intensities, fingerprint classification requires the extraction of global ridge-flow topology from a training corpus of approximately 1 600 samples — a configuration that creates a significant risk of overfitting to locally discriminative texture features that do not generalise across subjects. Three representations were investigated: the raw contrast-enhanced fingerprint image, a three-channel HSV orientation field image, and the Orientation Field Enhancement Map (OFEM). The experimental results establish a consistent performance hierarchy — Raw (82.00%) < OFEM (87.75%) < Orientation Field (91.25%) < ConvNeXt dual-input (92.50%) — whose interpretation in information-theoretic terms is developed throughout this section.

### 3.4.1 Raw Fingerprint Representation

The raw fingerprint representation serves as the experimental baseline. NIST SD4 images (512 × 512 pixels, 8-bit greyscale, 500 dpi) are segmented using a local variance threshold (0.001), refined through morphological closing and opening, and then processed with Contrast Limited Adaptive Histogram Equalisation (CLAHE, clip limit 2.0, tile grid 8 × 8) to normalise contrast locally without over-amplifying noise in dry or worn ridge areas [3]. The resulting foreground image is resized to 224 × 224 pixels by bicubic interpolation and replicated to three channels for backbone compatibility. The principal limitation of this representation is that the Henry class signal is global and topological, while convolutional networks process images through locally receptive fields. Composing the orientation signals needed for Henry classification from raw pixel values across a 1 600-sample training set is prone to overfitting on ridge width, ridge density, and local texture features that do not generalise across subjects. EfficientNet-B3 achieved 82.00% accuracy on this representation, with the most severe confusion between Plain Arch and Tented Arch — precisely the class pair where local texture is nearest-identical and the discriminative signal resides entirely in global ridge-flow geometry.

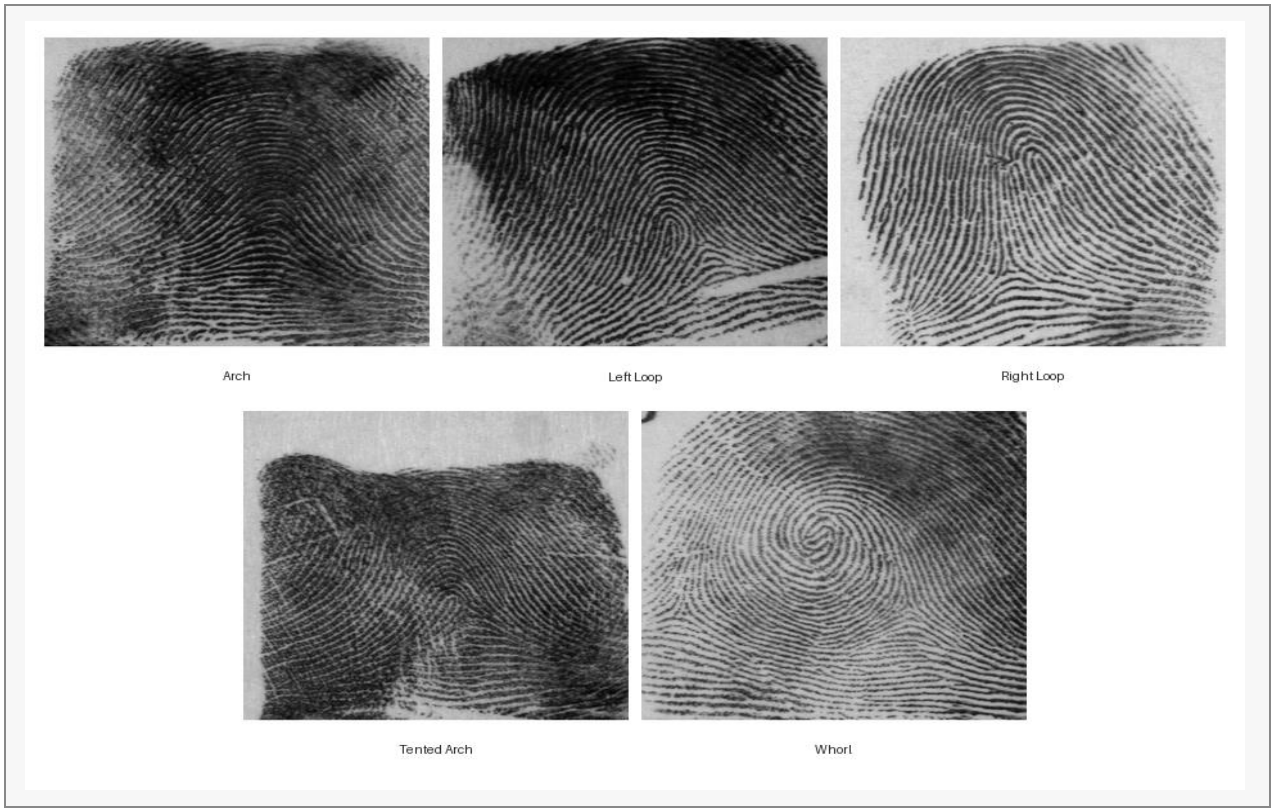



**Figure 3.1:** *Representative NIST SD4 samples, one per Henry class.*

### 3.4.2 Orientation Field Representation

The orientation field representation replaces the raw intensity map with an explicit encoding of the dominant ridge direction at each spatial location. The theoretical motivation is precise: the Henry classification criterion is defined entirely in terms of ridge-flow topology — the spatial arrangement of ridge directions and the locations and types of singular points where that arrangement exhibits characteristic discontinuities [3, 4, 17]. By making this directional structure the primary input channel, the preprocessing stage aligns the network input directly with the classificatory basis of the Henry system, reducing the feature extraction complexity that the convolutional layers must resolve.

Ridge orientation estimation follows the block-level gradient analysis pipeline of Hong, Wan, and Jain [3]. A Gaussian blur ($\sigma = 1.0$ pixels) suppresses sensor noise before gradient computation; $3 \times 3$ Sobel operators then yield horizontal and vertical components Gx and Gy at each pixel. Computing orientation directly from $\arctan(Gy/Gx)$ at pixel level is unreliable because a fingerprint ridge produces two parallel gradient contours with opposing polarity, causing the gradient direction to alternate between $+\theta$ and $-\theta$ across each ridge. The doubled-angle transformation resolves this ambiguity by computing $Vx = Gx^2 - Gy^2$ and $Vy = 2GxGy$, mapping opposite-polarity gradients to the same direction before block-level aggregation over $16 \times 16$ pixel windows:

$$\Phi x(b) = \Sigma\, Vx(i,j), \quad \Phi y(b) = \Sigma\, Vy(i,j) \quad \text{for } (i,j) \in \text{block } b \tag{4.1}$$

The dominant ridge orientation and block coherence are then recovered as:

$$\theta(b) = \tfrac{1}{2} \cdot \arctan2(\Phi y(b), \Phi x(b)) \tag{4.2}$$

$$C(b) = \sqrt{(\Phi x(b)^2 + \Phi y(b)^2)} \,/\, (Gxx(b) + Gyy(b) + \varepsilon) \tag{4.3}$$

where Gxx and Gyy are the block-level sums of $Gx^2$ and $Gy^2$ and ε is a stabilisation constant. Blocks with C(b) < 0.15 are excluded from the field, preventing unreliable estimates in scarred or marginal regions from corrupting the global output [4]. A critical regularisation step follows, necessary because θ is π-periodic: direct Gaussian smoothing of the angle array is mathematically inconsistent — two adjacent blocks at 2° and 178° represent nearly identical ridge directions, yet their arithmetic mean is 90°. The solution of Hong et al. [3] lifts each estimate to a continuous doubled-angle vector before smoothing and recovers the regularised field by:

$$*\Phi x^*(b) = \cos(2\theta(b)), \quad \Phi y^*(b) = \sin(2\theta(b))^* \qquad (4.4)$$

$$*\theta'(b) = \tfrac{1}{2} \cdot \arctan2(\, \text{smooth}(\Phi y^*), \text{smooth}(\Phi x^*)\,) \quad [\sigma = 13 \text{ px}]^* \qquad (4.5)$$

This lift-and-smooth procedure preserves sharp directional gradients at singular point neighbourhoods while eliminating artificial discontinuities in loop and whorl core regions [3, 4]. The regularised field is encoded as a three-channel HSV image: the hue channel H maps θ' across the full colour wheel, assigning perceptually distinct colours to each ridge direction; the saturation channel S encodes coherence C(b) so that reliably estimated blocks appear vivid and uncertain blocks fade toward grey; and the value channel V carries the CLAHE-enhanced intensity as a secondary cue. This encoding is critical for Left/Right Loop discrimination — in a greyscale representation, ridges at 0° and 180° produce identical intensity values and the two classes are indistinguishable at pixel level, whereas the HSV hue channel maps opposite directions to opposite ends of the colour wheel, providing an explicit laterality signal [11]. Across all backbone architectures evaluated, orientation-field inputs outperformed raw images by a consistent margin; ResNet50 achieved 91.25% versus 82.00% for EfficientNet-B3 on raw images, with the most pronounced per-class improvement in Left Loop F1 (+0.115), consistent with the laterality encoding advantage described above.

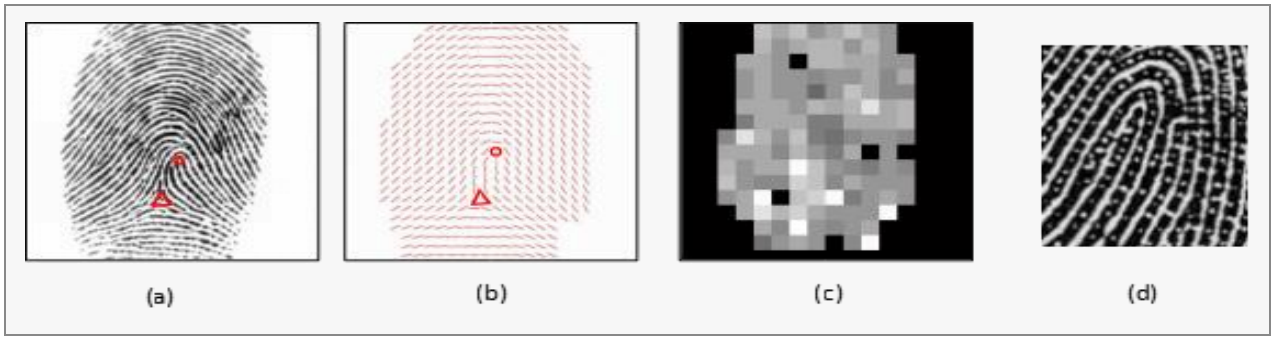



**Figure 3.2:** *Orientation-field computation pipeline.*

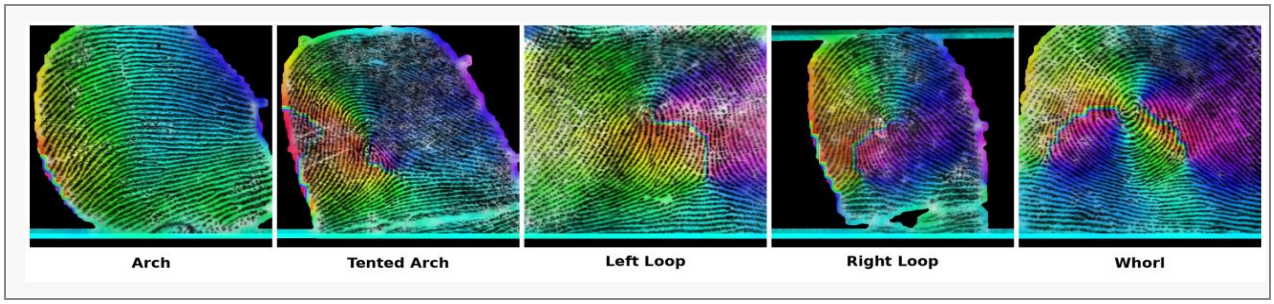



**Figure 3.3:** *HSV orientation-field images of the five Henry classes.*

### 3.4.3 Ridge Topology as the Basis of Classification

Fingerprint classification is fundamentally a topology recognition problem rather than a texture recognition problem. Ridge topology, as used in this context, refers to the global spatial pattern of ridge directions across the full impression and, in particular, the locations and types of singular points — locations where the orientation field exhibits characteristic discontinuities [15, 16]. The core is a point of maximum local ridge curvature associated with a characteristic circulation pattern; the delta is a point of triradial divergence associated with an opposite-sense circulation [4, 15, 16]. The Henry taxonomy maps directly onto singular point counts: Plain Arch contains no singular points; Tented Arch contains one core only; Left Loop and Right Loop each contain one core and one delta; Whorl contains two cores and two deltas [1, 17, 18].

This characterisation has a critical implication for representation design. The topologically discriminative regions — singular point neighbourhoods — occupy less than 5% of the total foreground area. The remaining 95% consists of locally parallel, smoothly varying ridge flow that is statistically near-identical across all five classes. A network processing raw pixel values must learn to selectively weight this small discriminative region while suppressing the uninformative background, without any explicit guidance about where the relevant regions are located. On a training set of 1 600 samples, this selective attention is difficult to learn robustly, and the network tends to overfit locally discriminative texture features — ridge frequency, ridge width, pore structure — that do not generalise. The orientation field representation resolves this by encoding ridge direction explicitly, making the singular point neighbourhoods visible as regions of locally elevated directional change and shifting the network's learning task from texture suppression to topological pattern recognition. The Left/Right Loop case is the clearest illustration: these two classes are topologically identical up to a spatial reflection, and their discrimination requires detecting a full-image lateral asymmetry that is invisible in greyscale intensity but encoded as a hue reversal in the HSV orientation image — a representational advantage directly confirmed by the +0.115 Left Loop F1 gain reported in Section 4.3 [3, 4, 17].

### 3.4.4 OFEM: Iterative Development of the Orientation Field Enhancement Map

The Orientation Field Enhancement Map was designed to construct an explicit spatial energy map concentrating high response values on singular point neighbourhoods while suppressing the non-discriminative normal ridge background. The underlying principle derives from the orientation deviation measure in singular point detection [15, 16]: in normal regions adjacent blocks share similar directions and deviation is small; in singular regions the field diverges and neighbours exhibit elevated mutual deviation. Exponential amplification of this deviation signal produces a sparse map where singular regions are highlighted in high contrast against a near-zero baseline. Developing a functional implementation required eight successive generator versions, each exposing a distinct failure mode that guided subsequent design decisions. Version 1 and Version 8 receive dedicated discussion as the baseline and final implementations; Versions 2–7 are summarised as a consolidated refinement narrative.

***OFEM Version 1 — Initial Implementation and Baseline Evaluation***

The first version adopted a power-law amplification of the block-level orientation deviation, intended to produce high energy values at singular regions and low values elsewhere. The fundamental error was the choice of amplification function: a power-law does not produce the same scale separation as the exponential specified in the theoretical formulation [15, 16], and for the deviation angles typical of genuine singular regions (20°–45°) the power-law output is substantially weaker, leaving insufficient contrast between singular and normal regions. The map appeared nearly uniform at low energy, with only marginal elevation at core and delta locations, and the spatial structure required for class discrimination was not recoverable.

Evaluated with ResNet50 on NIST SD4, Version 1 achieved 77.25% accuracy — a 4.75 percentage point regression relative to the 82.00% raw baseline. This result establishes that singular-point emphasis alone, when the emphasis signal is insufficiently sharp, actively degrades classification by discarding class-relevant ridge-flow information without providing a usable replacement. The error pattern was dominated by Plain Arch / Tented Arch confusion, confirming that the energy map was too insensitive to detect the single Tented Arch core against background noise. This failure established the exponential energy formulation as a strict requirement and defined the minimum acceptable contrast ratio for subsequent versions.

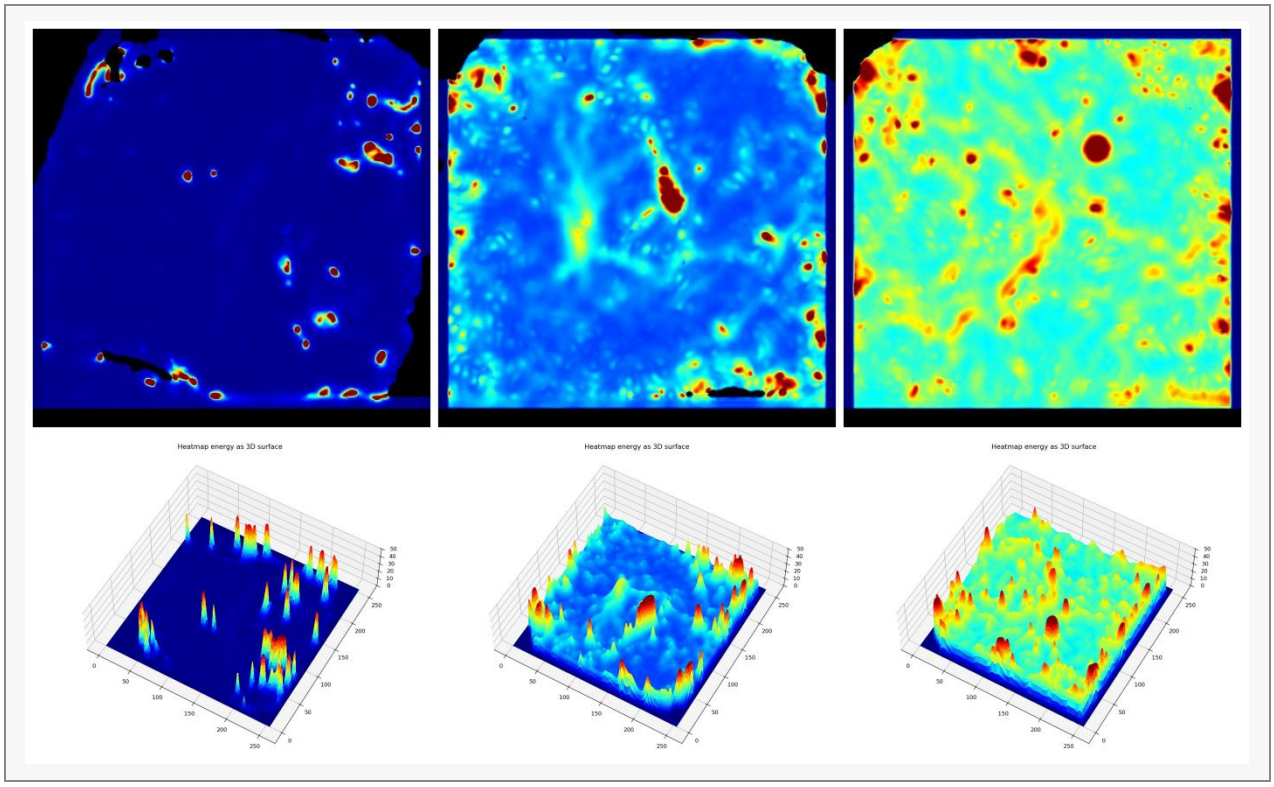



**Figure 3.4:** *OFEM Version 1 energy output.*

### *Intermediate OFEM Refinements — Versions 2 Through 7*

Version 2 corrected the energy formula to the exponential OFEN(b) = exp(mean(|Θ|)/μ) but set $\mu = \pi/18$ (10°), half the appropriate value, causing the entire foreground to saturate at near-maximum energy — informationally equivalent to a blank input since no structural contrast remained for the network to learn from. Version 3 corrected μ but introduced a pipeline bookkeeping error: the intermediate doubled-angle vector field [cos(2θ), sin(2θ)] was saved instead of the scalar energy map, producing a rainbow-coloured tensor that represented ridge direction rather than orientation deviation. Version 4 corrected the output path but retained pixel-level orientation deviation sampling, which re-introduced the sub-ridge oscillation problem at full scale: at pixel resolution, the gradient direction alternates between ridge-edge and ridge-centre values, producing uniformly high deviation across the entire foreground and eliminating all singular/normal distinction. This failure established block-level computation as a non-negotiable architectural requirement.

Version 5 adopted block-level deviation computation and correctly identified singular regions as high-energy locations, but nearest-neighbour upsampling from block-grid to full resolution introduced prominent staircase artefacts at block boundaries — a periodic spatial signal that a convolutional network would exploit as a feature, introducing a representation-specific bias unrelated to fingerprint topology. Version 6 replaced nearest-neighbour upsampling with bicubic interpolation, eliminating the staircase artefact, and added Otsu adaptive thresholding to sharpen the singular/normal contrast — producing the first visually and quantitatively correct OFEM output. Version 7 refined μ against ground-truth annotations, introduced energy range clipping to prevent extreme outlier values from compressing the dynamic range, and applied the coherence gate to suppress unreliable low-coherence blocks, reducing inter-image variability across the NIST SD4 corpus.

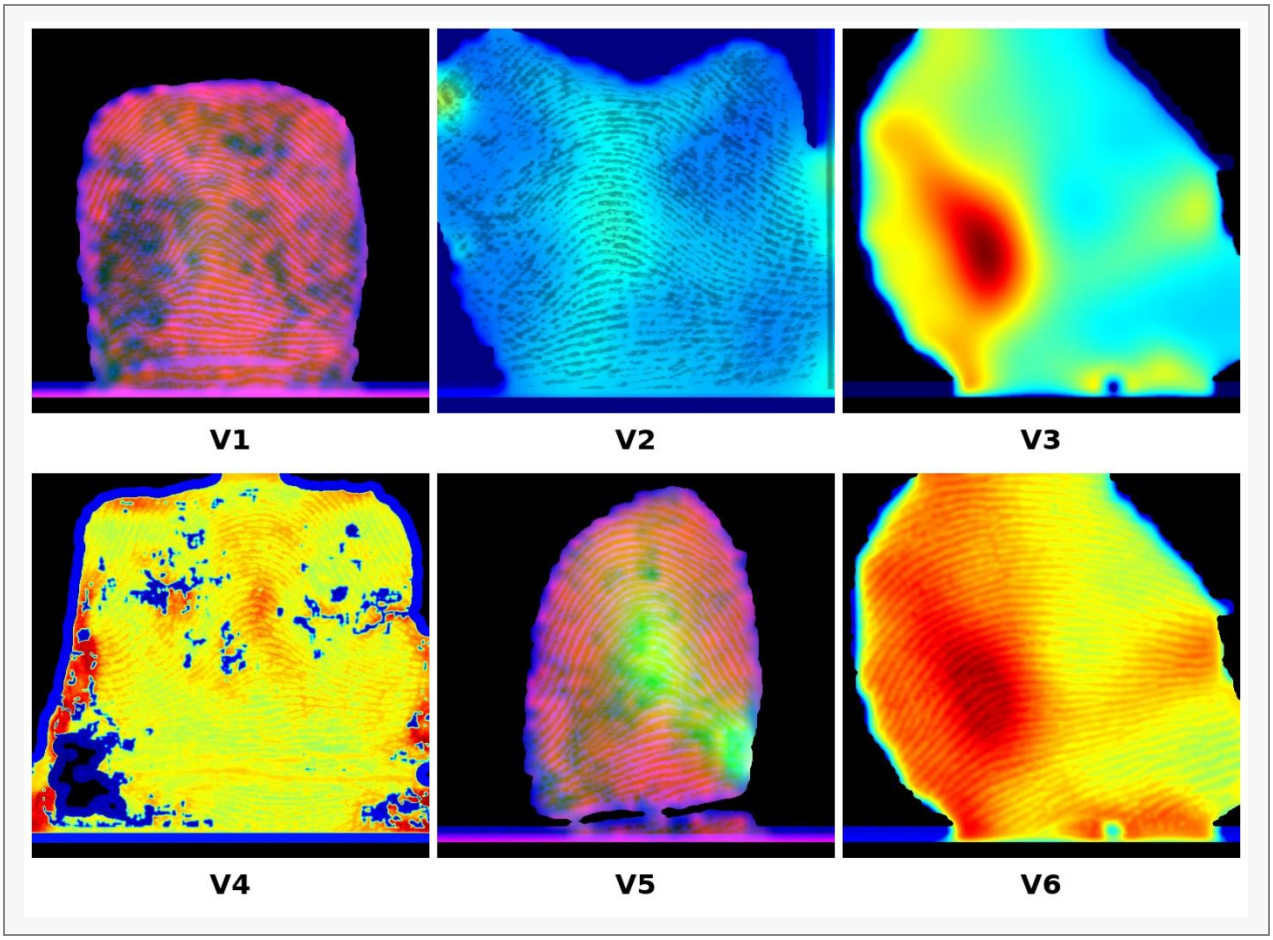



**Figure 3.5:** *OFEM Versions 2–7 progression.*

***OFEM Version 8 — Block-Grid Multi-Scale Architecture***

Versions 5–7 computed orientation deviation by sampling θ values at pixel coordinates centred within neighbouring blocks, retaining a residual vulnerability: coordinates falling near a ridge edge introduced sub-ridge oscillation into the deviation estimate. Version 8 eliminated this by replacing pixel-coordinate sampling with a block-grid neighbourhood scheme — the deviation at target block b is computed by comparing its block-level orientation estimate θ(b) against the block-level estimates of neighbours at three discrete radii of 1, 2, and 3 block units (16, 32, and 48 pixels at 500 dpi). Both target and neighbour orientations are block-level averages, immune to sub-ridge oscillation. The three-radius scheme provides a multi-scale characterisation: the 1-block radius captures the immediate divergence at the singular point, the 2-block radius captures the broader disturbed flow region, and the 3-block radius provides structural context at the scale of a full core or delta. The energy formula aggregates all radii:

$$*\text{OFEN}(b) = \exp(\text{mean}\{\ |\Theta(b, b')| : b' \in N(b)\ \} / \mu)* \qquad (4.6)$$

where N(b) is the set of all neighbour blocks at radii 1–3 block units, $|\Theta(b, b')|$ is the absolute deviation modulo π, and $\mu = \pi/9$ (20°). In normal regions mean$|\Theta| \approx 0$ and OFEN(b) ≈ 1.0; in singular regions mean$|\Theta|$ approaches 45° and OFEN(b) approaches exp(2.25) ≈ 9.5, producing a roughly ten-fold contrast ratio. This contrast is encoded as a three-channel JET colourmap image — teal for normal regions,

red for singular regions. Trained with VGG16-BN, Version 8 achieved 87.75% — a 10.5 percentage point improvement over Version 1 and 5.75 points over the raw baseline, confirming that the iterative development process produced a genuine and substantial improvement in topological encoding quality. Nevertheless, the 3.5 point gap below ResNet50 on orientation images reflects a fundamental trade-off examined in the following subsection.

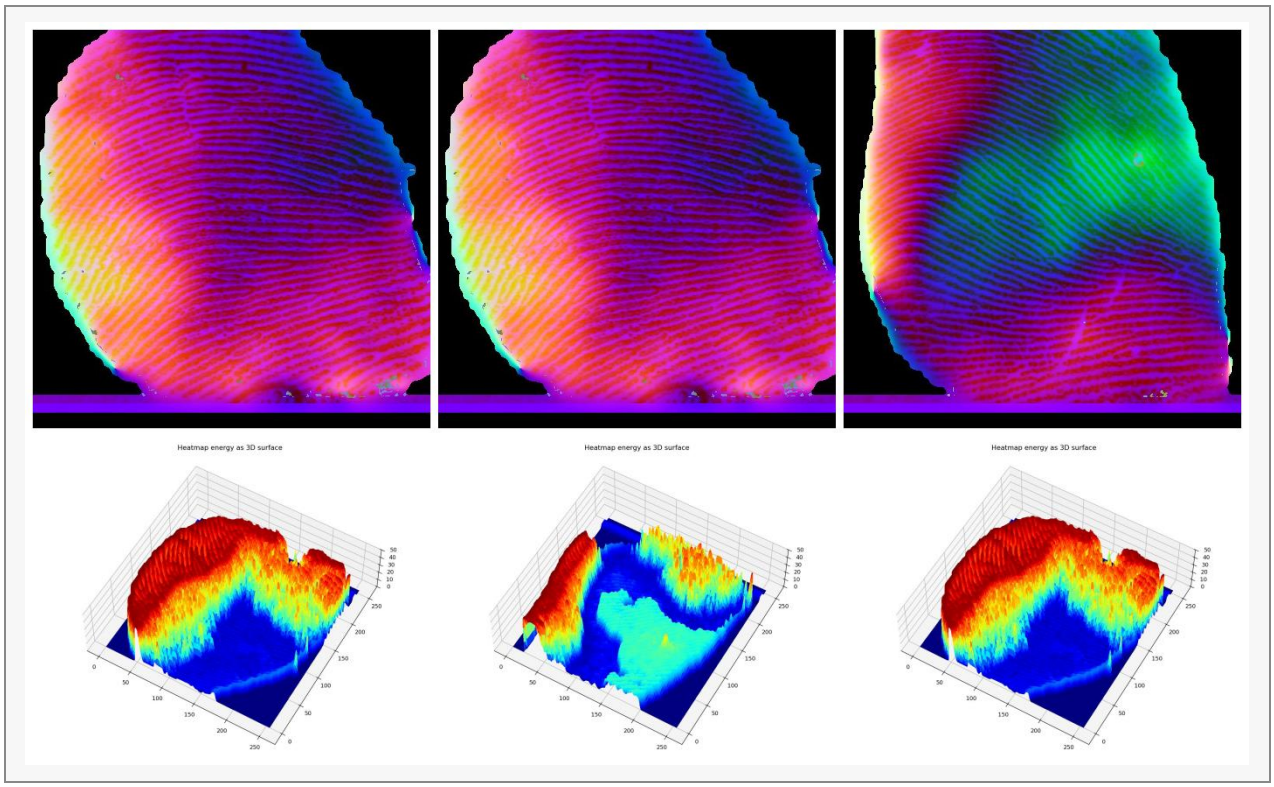



**Figure 3.6:** *OFEM Version 8 energy output.*

### *OFEM Evolution Summary*

Table 3.5 summarises the eight OFEM versions. The progression illustrates a systematic debugging process in which each failure exposed one distinct design requirement: correct energy function (v1), correctly scaled normalisation (v2), correct output format (v3), block-level spatial sampling (v4), smooth upsampling (v5–v6), parameter calibration and coherence gating (v7), and multi-scale block-grid neighbourhood analysis (v8). Version 8 satisfies all requirements simultaneously and constitutes the final OFEM implementation used in classification and ensemble experiments.

| Version | Primary Change | Failure / Outcome | Status |
|---|---|---|---|
| v1 | Power-law energy | Insufficient contrast — 77.25% (ResNet50) | ✗ Failed |
| v2 | Exponential; $\mu = \pi/18$ | $\mu$ too small — entire foreground saturated | ✗ Failed |
| v3 | Corrected $\mu$; output path error | Vector tensor saved — rainbow artefact | ✗ Failed |
| v4 | Correct output; pixel-level OD | Sub-ridge oscillation — foreground uniformly high | ✗ Failed |
| v5 | Block-level OD; INTER_NEAREST | Correct detection; staircase boundary artefact | ⚠ Partial |
| v6 | Bicubic upsampling; Otsu threshold | First correct output — smooth singular region blobs | ✓ Correct |
| v7 | Tuned $\mu$; ENR clipping; coherence gate | Improved consistency and background suppression | ✓ Better |
| v8 | Block-grid multi-radius OD | 87.75% — +10.5 pp over v1 | ✓ Best |

**Table 3.5:** *Summary of the eight OFEM generator versions with primary design change and outcome.*

### *Why OFEM Did Not Surpass Orientation-Field Representations*

The performance hierarchy Raw (82.00%) < OFEM (87.75%) < Orientation Field (91.25%) reflects a fundamental representational trade-off, not an implementation deficiency. OFEM is by design a compression operation: it converts the full orientation field into a sparse energy map by retaining orientation deviation magnitudes and discarding orientation directions. This compression reduces input noise, makes singular point locations explicit as high-contrast blobs, and suppresses the non-discriminative normal ridge background — properties that explain the 5.75 pp improvement over the raw baseline. However, it simultaneously discards two categories of information that the orientation field preserves and the classification task requires: the directional context surrounding singular regions (the curvature of ridges approaching a core, the angular spread at a delta, the flow continuity between singular points) and the complete global ridge-flow structure that encodes, for instance, the relative core-delta spatial relationship in Loop impressions.

The orientation field performs a more conservative compression: it suppresses local intensity variation — which carries no class information — while retaining the complete directional structure of the ridge-flow field, including both the singular point signal and the rich contextual information surrounding it. This balance between noise suppression and information preservation explains the orientation field's consistent superiority across all Henry classes, and particularly its better resolution of the Tented Arch versus Plain Arch ambiguity, where the single Tented Arch core must be distinguished from the flat flow of a Plain Arch using the full directional context of the surrounding field rather than a binary hotspot detection. OFEM and orientation fields thus address the same underlying problem through complementary strategies — OFEM maximises singular-point salience at the cost of directional richness; the orientation field maximises topological completeness at the cost of requiring the network to perform more spatial integration — and the experimental evidence consistently favours the latter as the stronger standalone representation.

### 3.4.5 Comparative Analysis of Fingerprint Representations

The performance hierarchy — Raw (82.00%) < OFEM (87.75%) < Orientation Field (91.25%) < ConvNeXt dual-input (92.50%) — is entirely consistent with the information-theoretic properties of each representation summarised in Table 3.6. The raw CLAHE representation preserves the complete local intensity structure of the fingerprint, but this completeness is a liability for Henry classification: the class-irrelevant intensity variation competes with the topological signal throughout feature extraction, and on 1 600 training samples the network overfits locally discriminative texture patterns. The 82.00% ceiling is an empirical confirmation of this constraint. The OFEM representation occupies the opposite extreme: singular point locations are maximally salient and input noise is minimised, but the directional richness needed to resolve inter-class ambiguities has been discarded. The 87.75% accuracy correctly positions OFEM as a meaningful improvement over raw images but below the informationally more complete orientation field.

The HSV orientation field achieves the best balance: it eliminates local intensity variation while retaining the complete directional structure of the ridge-flow field. The singular point signal is present in the orientation image as a region of elevated directional gradient — not as explicitly highlighted as in the OFEM — but the directional context surrounding singular regions is fully preserved and the HSV hue channel provides an explicit laterality signal for Left/Right Loop discrimination that neither raw images nor OFEM can supply. The 91.25% accuracy of ResNet50 and the 92.50% of ConvNeXt-Tiny trained on the joint raw-and-orientation input reflect this informational completeness, with the performance trends across all model configurations further analysed in Sections 4.3 and 4.6.

From the perspective of ensemble design, the informational differences between representations define the potential for complementary error patterns. An OFEM-trained model will tend to fail on cases requiring directional context — ambiguous Tented Arch cases, near-boundary singular regions — while an orientation-field model is more likely to fail when singular regions are spatially diffuse or partially obscured. If these error sets are sufficiently non-overlapping, combining the two representations in an ensemble offers accuracy gains beyond either standalone model. The degree to which this complementarity was realised in practice is examined in Section 3.8.

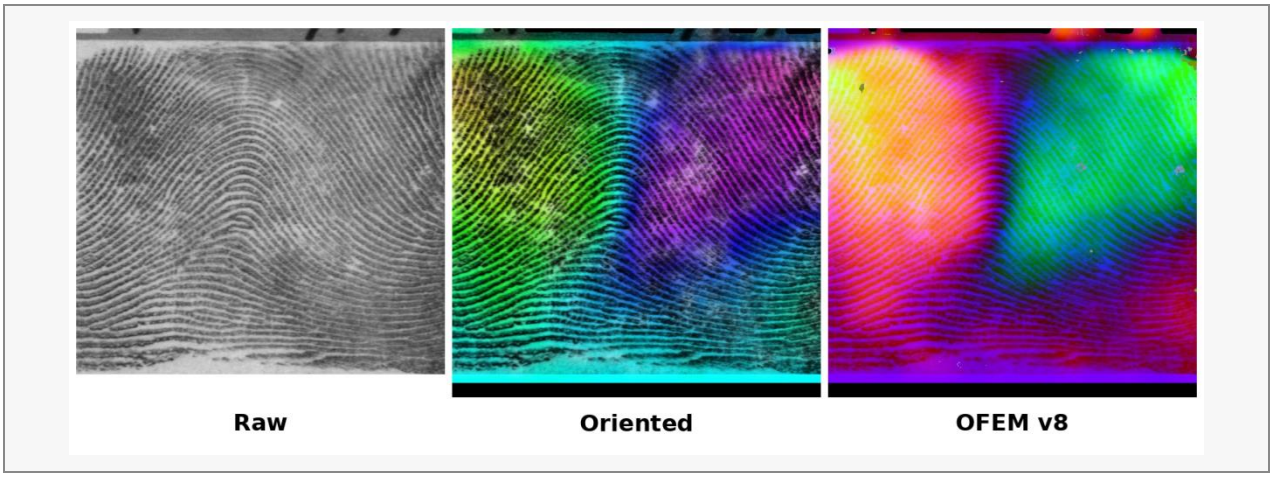



**Figure 3.7:** *Comparison of the three input representations.*

| Property | Raw CLAHE | HSV Orientation Field | OFEM v8 |
|---|---|---|---|
| Local intensity texture | Preserved | Suppressed | Suppressed |
| Ridge direction field | Implicit | Explicit — full field | Implicit via deviation energy |
| Directional context at singular points | Implicit / weak | Fully preserved | Discarded |
| Singular point localisation | Implicit / weak | Present but distributed | Explicit / concentrated |
| Laterality encoding (L/R Loop) | None | Strong — HSV hue channel | Spatial position only |
| Noise sensitivity | High | Moderate — coherence gated | Low — block-averaged + gated |
| Topological completeness | Low — mixed with texture | High — full orientation structure | Medium — singular regions only |
| Best accuracy achieved | 82.00% (EfficientNetB3) | 91.25% (ResNet50) / 92.50% (ConvNeXt) | 87.75% (VGG16-BN) |

**Table 3.6:** *Comparative summary of the three fingerprint input representations investigated in this study.*

The analytical conclusion of this section — that orientation-field representations achieve the best balance between topological completeness and noise suppression, outperforming both raw images and OFEM across all architectures evaluated — directly motivates the architecture selection and training configurations described in Section 3.6.

## 3.5 Classification Paradigm and Optimisation Strategy

Before the specific backbone architectures are characterised, three methodological decisions that apply uniformly across every model in this study require justification: the learning paradigm under which the networks are trained, the optimisation algorithm used to update their weights, and the broader choice of convolutional architectures over the transformer-based alternatives that have become prominent in computer vision. Each decision is dictated by the specific constraints of the fingerprint classification task on NIST SD4 — a small, fully labelled dataset of locally structured texture images — and each is defended below on those grounds.

### 3.5.1 Supervised Multi-Class Formulation

The fingerprint classification task is formulated as supervised learning: every image in NIST SD4 carries a ground-truth Henry class label — Arch, Left Loop, Right Loop, Tented Arch, or Whorl — assigned before training. The network learns the mapping from image to label by minimising the cross-entropy loss between its predicted class distribution and the ground-truth label across the training set. No unsupervised, semi-supervised, or self-supervised component is involved; the availability of a complete set of expert-assigned class labels for every image is precisely what makes the supervised formulation both possible and appropriate.

The study employs two related but distinct supervised formulations, summarised in Table 4.X. The general classifiers — EfficientNet-B3, ResNet50, and ConvNeXt-Tiny — are trained as multi-class classifiers over all five Henry classes, producing a five-way softmax output. The Tented Arch specialist (VGG16-BN) is trained as a binary classifier under a one-versus-rest formulation, distinguishing the Tented Arch class from all other classes collapsed into a single negative category. This binary reformulation concentrates the model's entire representational capacity on the single most error-prone decision boundary in the task, and is the mechanism by which the specialist achieves its substantial Tented Arch F1-score improvement.

| Model | Task | Classification Type |
|---|---|---|
| EfficientNet-B3 | 5 Henry classes | Multi-class supervised |
| ResNet50 | 5 Henry classes | Multi-class supervised |
| ConvNeXt-Tiny | 5 Henry classes | Multi-class supervised |
| VGG16-BN (specialist) | Tented Arch vs rest | Binary supervised |

**Table 4.X:** *Learning task and classification type for each model in the study. All formulations are fully supervised, using the ground-truth Henry class labels provided with NIST SD4.*

### 3.5.2 Choice of Optimiser: AdamW

All models were trained using the AdamW optimiser. Adam combines two mechanisms that are individually beneficial for training deep networks: momentum, which accumulates a running estimate of the gradient direction and smooths the optimisation trajectory across successive batches, and per-parameter adaptive learning rates, which scale each weight update according to how frequently and how strongly that parameter has been updated. The combination converges rapidly from a pretrained starting point and is robust to the noisy gradient estimates produced by the small batch sizes that the dataset size and GPU memory impose.

AdamW differs from plain Adam in its treatment of weight decay. In the original Adam formulation, weight decay is folded into the gradient and consequently interacts with the adaptive per-parameter scaling, which distorts the intended uniform regularisation effect. AdamW decouples weight decay from the gradient-based update, applying it directly to the weights as a separate step.

This correction yields measurably better generalisation, an effect that is most pronounced precisely in the regime relevant to this study — fine-tuning large pretrained networks on a small dataset, where over-fitting pressure is high and correct regularisation is critical. For fine-tuning ImageNet-pretrained backbones (EfficientNet-B3, ResNet50, VGG16-BN, ConvNeXt-Tiny) on the approximately 2 000 images of NIST SD4, AdamW represents current best practice. Stochastic gradient descent with momentum, the principal alternative, would require careful manual learning-rate scheduling and converges substantially more slowly from a pretrained initialisation, offering no compensating advantage on a dataset of this scale.

### 3.5.3 Choice of Convolutional Networks over Vision Transformers

The decision to build the classification system on convolutional networks rather than Vision Transformers (ViT) is a deliberate methodological choice grounded in three task-specific considerations, not a default. Each concerns the match between architectural inductive bias and the constraints of the fingerprint classification problem.

The first consideration is dataset size. The Vision Transformer architecture was designed for very large training corpora; the original formulation reports that ViT underperforms convolutional networks on datasets below approximately one hundred thousand images in the absence of large-scale pretraining [9]. NIST SD4 contains only 2 000 images. A ViT partitions the input into patches and learns the relationships among them through self-attention, a mechanism that must infer spatial structure entirely from data and therefore requires a very large number of examples to learn meaningful patch relationships from scratch. Convolutional networks, by contrast, build translation equivariance directly into the architecture as a structural inductive bias rather than learning it from data, allowing them to extract spatial patterns efficiently even from small datasets.

The second consideration is the nature of the fingerprint signal. Fingerprints are locally structured ridge-valley textures in which the discriminative information — local ridge orientation, singular-point topology within a small neighbourhood — is fundamentally local in character. Convolutional kernels are explicitly designed to capture local spatial patterns through their sliding-window operation, making the CNN inductive bias a precise match for the structure of the problem. A ViT treats the image as a sequence of flattened patches with no built-in notion of local spatial continuity, and must learn locality from data — again demanding far more examples than are available here.

The third consideration is transfer learning. The convolutional backbones evaluated in this study carry ImageNet-pretrained weights whose early layers already detect the edges, textures, and local patterns that fingerprint ridge analysis depends on; fine-tuning these layers on 2 000 fingerprint images succeeds precisely because the low-level features transfer directly. ViT pretrained weights encode global attention patterns that transfer less directly to local ridge-texture analysis. It should be acknowledged that with a large fingerprint database (exceeding one hundred thousand images) and sufficient computational resources, a transformer architecture incorporating local attention — particularly the Swin Transformer, whose shifted-window mechanism reintroduces the local structure that plain ViT lacks — could match or exceed convolutional performance, and this possibility is identified as a future research direction. Under the actual constraints of this study, however, convolutional networks with ImageNet pretraining are both the scientifically correct and the empirically stronger choice.

## 3.6 Deep Learning Architectures for Fingerprint Classification

With the preprocessing representations established in Section 3.4, the next design question concerns the convolutional architectures through which these representations are processed. Section 3.6 is divided into two parts. The present section (4.5.1) develops the theoretical foundations of convolutional neural networks as they apply to the fingerprint classification problem. Section 3.6.2 then characterises the five specific backbone architectures evaluated in this study — EfficientNet-B3, VGG16-BN, ResNet50, MobileNetV3-Small, and ConvNeXt-Tiny — analysing the design properties of each in relation to the fingerprint classification task.

### 3.6.1 Fundamentals of Convolutional Neural Networks

The selection of convolutional neural networks as the classification engine of this study was not self-evident at the outset. Classical fingerprint classification methods based on handcrafted features — Fourier descriptors of orientation fields, Gabor filter responses, model-based singular point detectors — have decades of development and theoretical grounding behind them [17, 18]. The argument for replacing them with learned representations rests on a specific claim: that for the Henry classification task as presented on NIST SD4, a network trained with sufficient regularisation

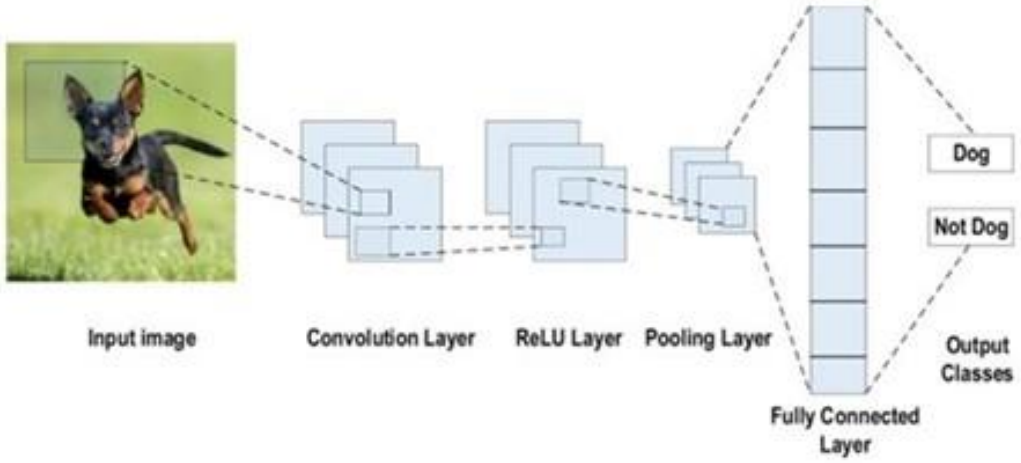



**Figure 3.8:** *Convolutional neural network architecture.*

can discover feature combinations from orientation-based inputs that are more diagnostically reliable than any fixed engineered descriptor. Whether this claim holds depends on whether the convolutional architecture possesses the right structural properties for the task — the question the following subsections address.

***Convolutional Layer (CONV)***

The convolutional layer is the core computational unit of a CNN. It slides a set of learnable filters across the input image, computing dot products at each spatial position to produce feature maps that capture edges, textures, and local patterns relevant to the classification task. Each filter specialises in detecting a specific visual feature, and deeper layers compose these detections into increasingly abstract representations. For fingerprint classification, early convolutional layers respond to individual ridge segments, while deeper layers integrate these local responses into the

global topology patterns — loop cores, whorl spirals, arch curvatures — that distinguish Henry classes.

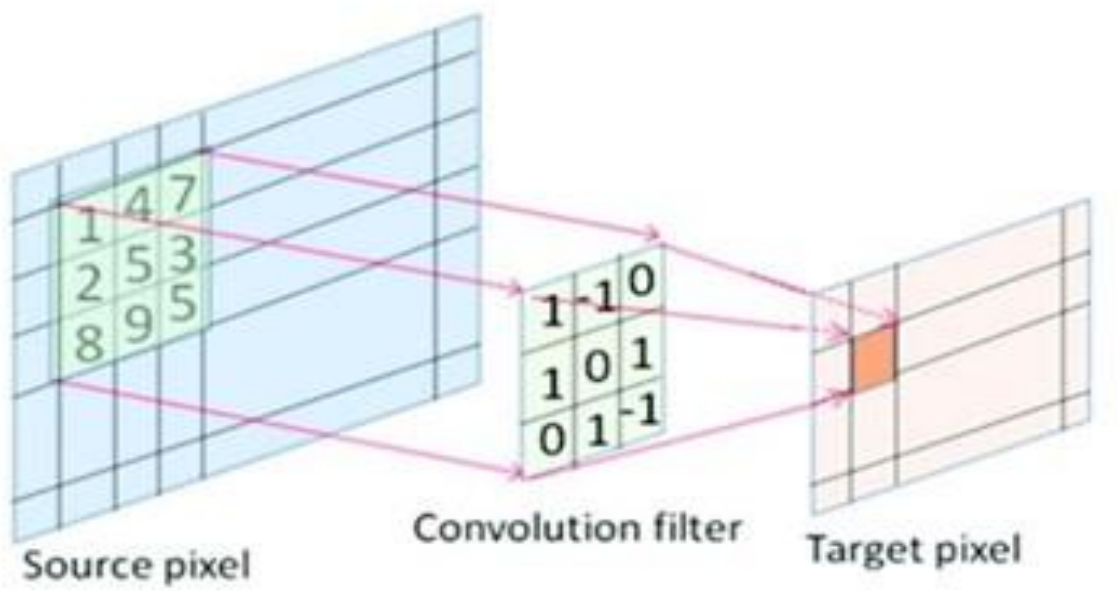



**Figure 3.9:** *The convolution operation.*

### *ReLU Activation Function*

The Rectified Linear Unit (ReLU) activation follows each convolution, replacing every negative activation with zero: f(x) = max(0, x). This non-linearity is essential: without it, stacking arbitrarily many linear layers reduces to a single linear transformation regardless of depth, and the network loses its ability to learn complex decision boundaries. ReLU also produces sparse activations — only neurons responding positively to the current input are active — which reduces computational load and mitigates the vanishing gradient problem that plagued earlier activation functions such as the sigmoid.

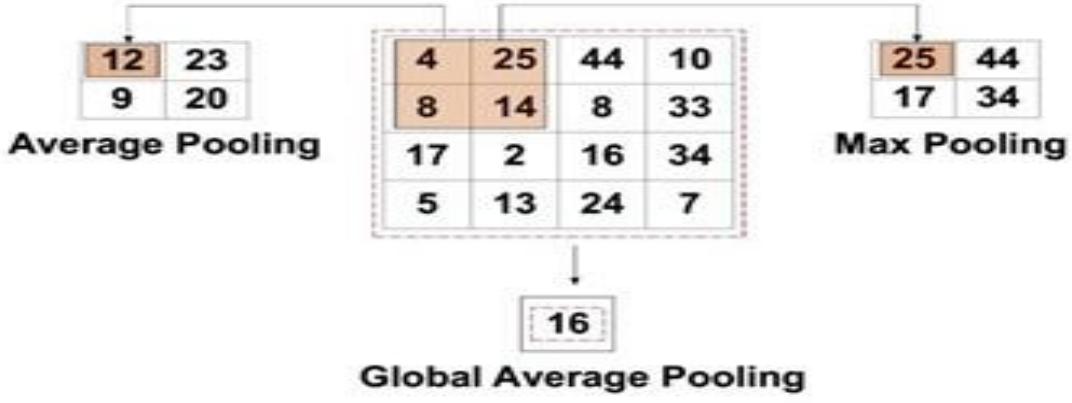



**Figure 3.10:** *The ReLU activation function.*

### *Pooling Layer (POOL)*

The pooling layer reduces the spatial dimensions of each feature map by summarising local regions into a single value. Max pooling retains the strongest activation within each sampling window, preserving the most salient detected feature regardless of its precise spatial location; average pooling computes the regional mean, producing a smoother summary. Both operations reduce the number of parameters propagated to subsequent layers, lower computational cost,

and introduce a degree of translation invariance — a fingerprint shifted slightly within the image frame will produce similar pooled activations, improving generalisation. Successive pooling layers progressively enlarge the effective receptive field, allowing deeper neurons to integrate information across wider spatial contexts.

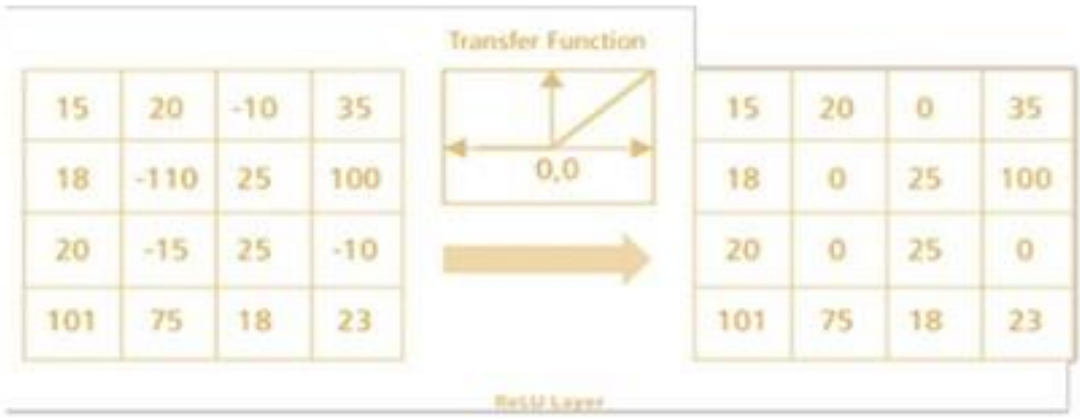



**Figure 3.11:** *Max and average pooling.*

### *Fully-Connected Layer (FC)*

The fully-connected layer performs the final classification. After the convolutional and pooling stack has compressed the input into a compact, spatially-pooled feature vector, the FC layer connects every element of that vector to every output neuron, creating a dense weighted mapping from learned spatial features to class scores. A softmax function applied to the output converts raw scores to a probability distribution over the five Henry classes. All classification-head layers in the present study follow this architecture: a dropout-regularised FC layer reduces the feature dimensionality before the final five-way softmax output.

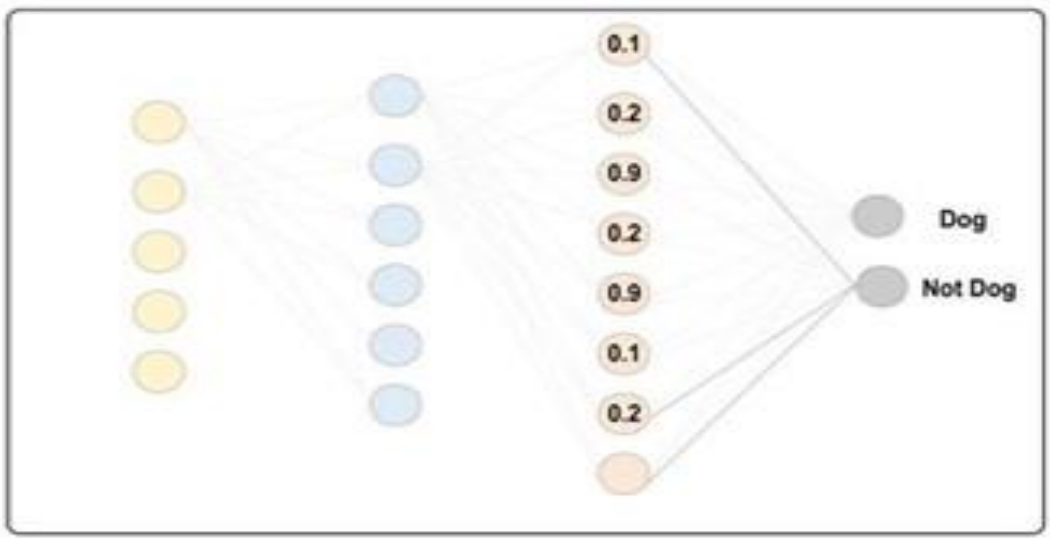



**Figure 3.12:** *The fully-connected layer.*

#### 3.6.1.7 Why CNNs Were Selected for Fingerprint Classification

Traditional fingerprint classification pipelines construct features through explicit algorithmic stages — orientation field computation, singular point detection, inter-point geometric measurement, rule-based class assignment [1, 17, 18]. These methods are interpretable and require no training data, but they are brittle: singular point detection fails on scarred, dry, or smudged impressions, and fixed geometric templates do not accommodate the continuous morphological variation within each Henry class. The Tented Arch class illustrates the problem particularly clearly: its single core produces an orientation field disturbance that falls on a continuum between the flat flow of a Plain Arch and the full loop structure of a Left or Right Loop, and its discrimination from adjacent classes requires sensitivity to fine-grained directional cues that cannot be reliably captured by any fixed threshold rule [15, 16].

Convolutional networks address both limitations through learned feature extraction: they map input representations directly to class probabilities from labelled examples, discovering diagnostically useful feature combinations without any explicit geometric rule. Their suitability for the three representations investigated here follows from the properties described above. For raw images, hierarchical feature learning and large receptive fields enable global topology extraction from pixel values at the cost of higher data requirements. For orientation fields, early convolutional layers are ideally matched to detecting directional transitions and HSV hue gradients, and deep layers integrate these cues into the global singular-point configurations that define each Henry class. For OFEM maps, sparse high-energy blobs on a near-zero background are well suited to the selective activation of ReLU-gated filters, which respond strongly to singular regions and produce near-zero outputs for the uniform background. With these foundations established, Section 3.6.2 characterises the five specific backbone architectures — EfficientNet-B3, VGG16-BN, ResNet50, MobileNetV3-Small, and ConvNeXt-Tiny — in relation to the fingerprint classification task.

### 3.6.2 Evaluated Deep Learning Architectures

Five convolutional backbone architectures were evaluated across the three fingerprint representations described in Section 3.4. The selection covered a deliberately broad range of design philosophies — from the compound scaling of EfficientNet to the transformer-inspired modernisation of ConvNeXt — to allow a controlled assessment of which architectural properties are most beneficial for the fingerprint classification task. For each architecture the discussion addresses the key structural innovation, the experimental configuration, and the interpretation of the observed results in terms of the representational and topological considerations established in Sections 3.4 and 4.5.1.

### 3.6.2.1 EfficientNet-B3

EfficientNet-B3 was selected as the baseline model for raw fingerprint classification. The EfficientNet family scales network depth, width, and input resolution simultaneously according to a fixed proportionality coefficient, producing higher accuracy per parameter count than single-axis scaling [24]. With approximately 12 million parameters and a native resolution of 300 × 300 pixels, EfficientNet-B3 sits at a point on the parameter-efficiency curve well suited to a small training corpus where overfitting is the primary risk. The classification head used a 1 536 → 512 dimensional reduction with batch normalisation, ReLU, and dropout at 0.4; fine-tuning unfroze the last two backbone blocks after the head-training phase.

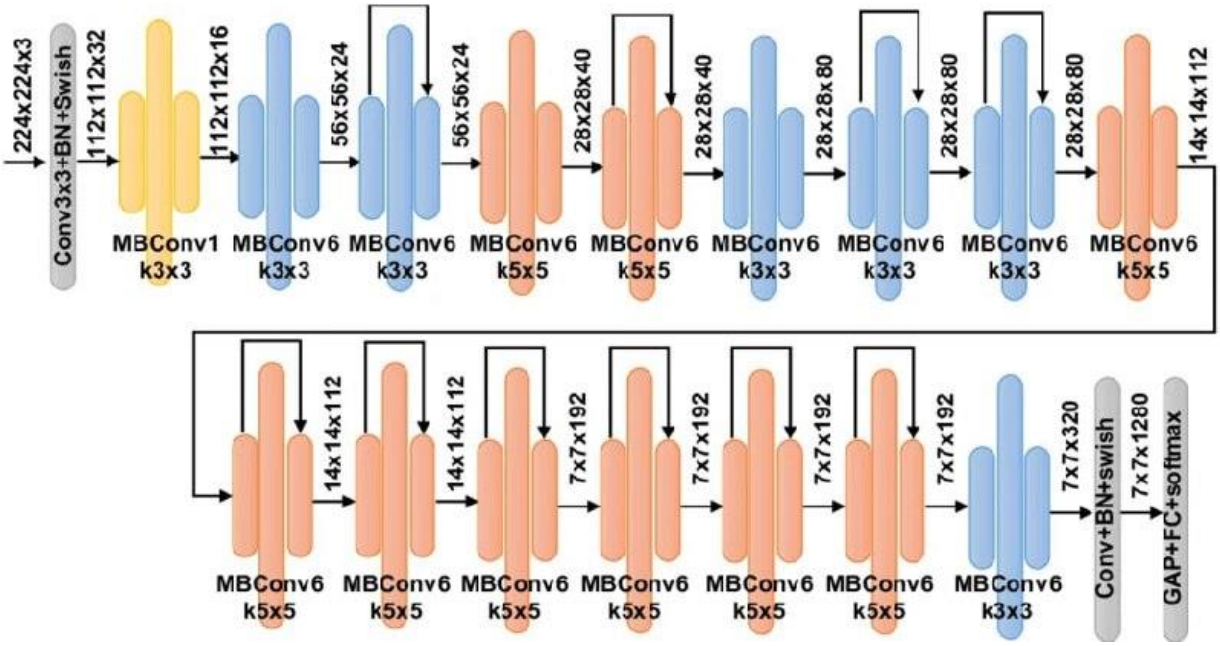



**Figure 3.13:** *EfficientNet-B3 architecture.*

### 3.6.2.2 VGG16-BN with OFEM Representation

VGG16 with batch normalisation was selected for the OFEM experiments on the basis of its uniform sequential architecture — five blocks of 3 × 3 convolutional layers with max pooling — whose predictable feature extraction behaviour makes it well suited to structured visual inputs such as the compact energy blobs of the OFEM map. The classification head used a 25 088 → 1 024 dimensional reduction with batch normalisation, ReLU, and dropout at 0.4; backbone fine-tuning unfroze blocks 4 and 5. Trained on OFEM Version 8 JET colourmap images, VGG16-BN achieved 87.75% — a 5.75 percentage point improvement over the raw baseline, confirming that explicit singular-point highlighting provides a more accessible topological signal than raw texture processing. The 3.5 percentage point gap below ResNet50 on orientation images reflects the OFEM's fundamental trade-off: by discarding directional context in favour of singular-region salience, it removes the secondary cues needed to resolve ambiguous cases, and VGG16's sequential architecture has no mechanism to recover global directional context from the sparse energy map.

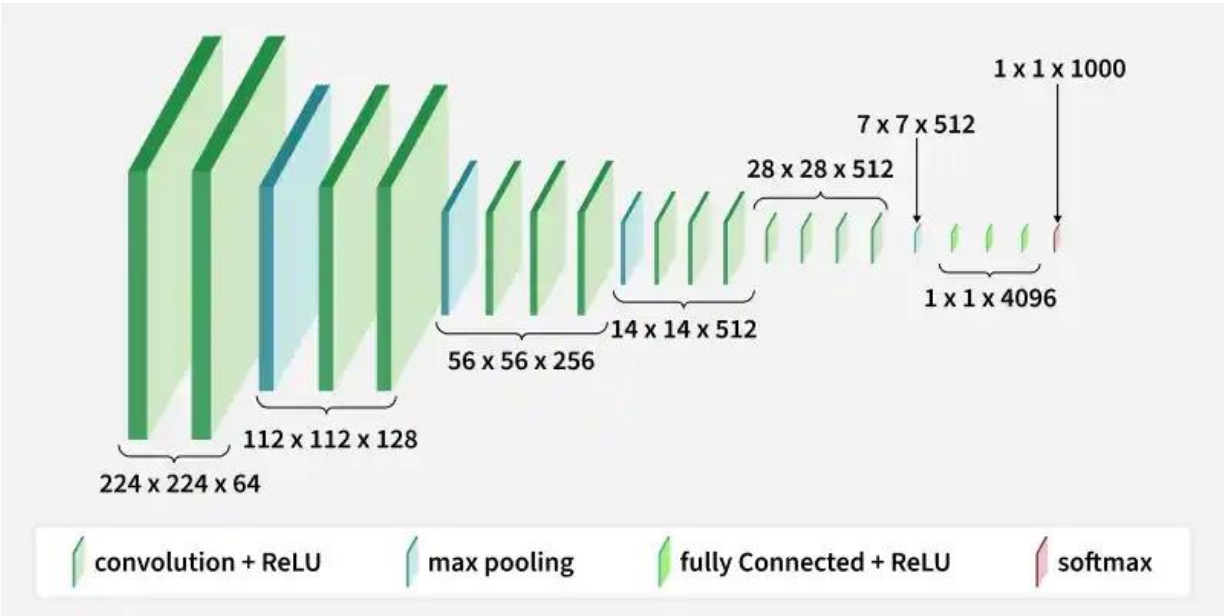



**Figure 3.14:** *VGG16-BN architecture.*

### 3.6.2.3 ResNet50 with Orientation-Field Representation

ResNet50 was the strongest architecture prior to ConvNeXt, and its pairing with the HSV orientation field produced the clearest demonstration of the representational advantage that constitutes one of the central findings of this study. The ResNet family introduced residual learning — identity shortcut connections that allow the gradient to bypass one or more convolutional layers during backpropagation, decomposing the learning objective into a residual function and enabling networks of 50 or more layers to train effectively without the accuracy degradation observed in very deep plain networks [25]. For orientation field inputs, this architecture offers a specific advantage beyond gradient stability: the class-discriminative information in the HSV image is distributed across the full spatial extent as a coherent directional field, and learning to classify it requires that early directional feature maps remain accessible to the deep integration layers. The skip connections provide exactly this preservation mechanism, maintaining a direct representational pathway from early directional responses to the final classification layers. The head used a 2 048 → 512 dimensional reduction with batch normalisation, ReLU, and dropout at 0.4; fine-tuning unfroze layer3 and layer4.

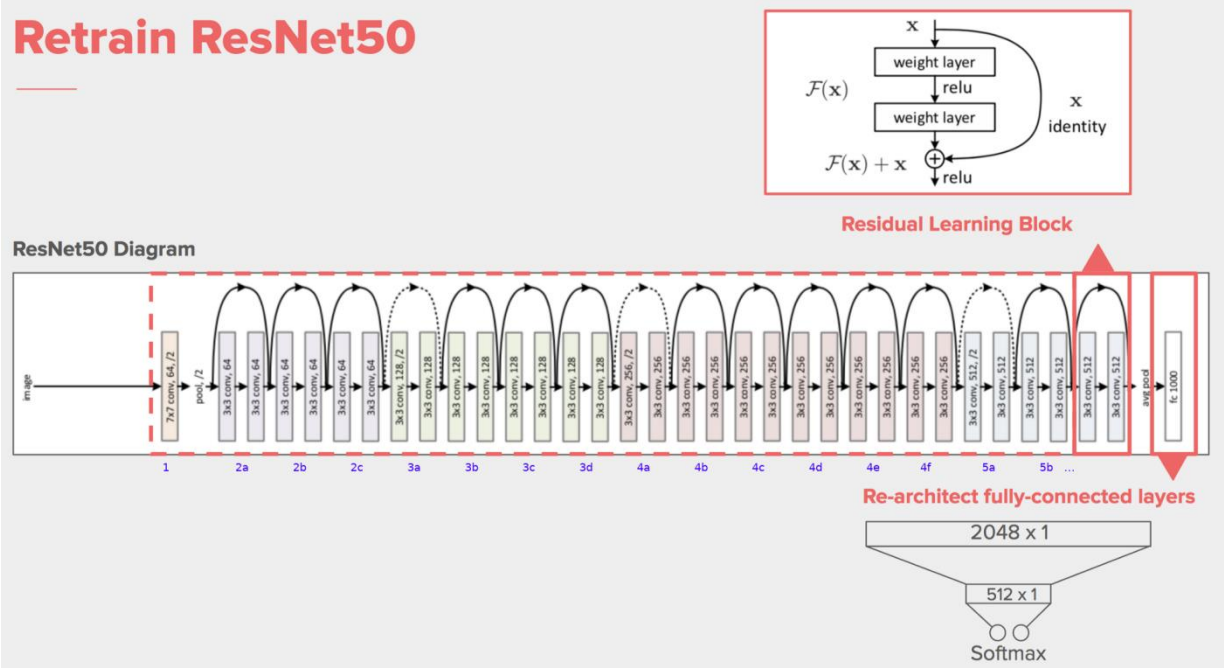



**Figure 3.15:** *ResNet50 architecture.*

### 3.6.2.4 MobileNetV3-Small — Lightweight Specialist Candidate

MobileNetV3-Small was investigated as a candidate for the Tented Arch specialist classifier. The MobileNet family employs depthwise separable convolutions — factorising a k × k convolution into a depthwise spatial stage and a pointwise 1 × 1 channel mixing stage — reducing parameter count and operations by a factor of approximately $k^2$ relative to full convolution. MobileNetV3-Small adds hard-swish activation and squeeze-and-excitation attention to partially compensate for the representational capacity lost through factorisation [26].

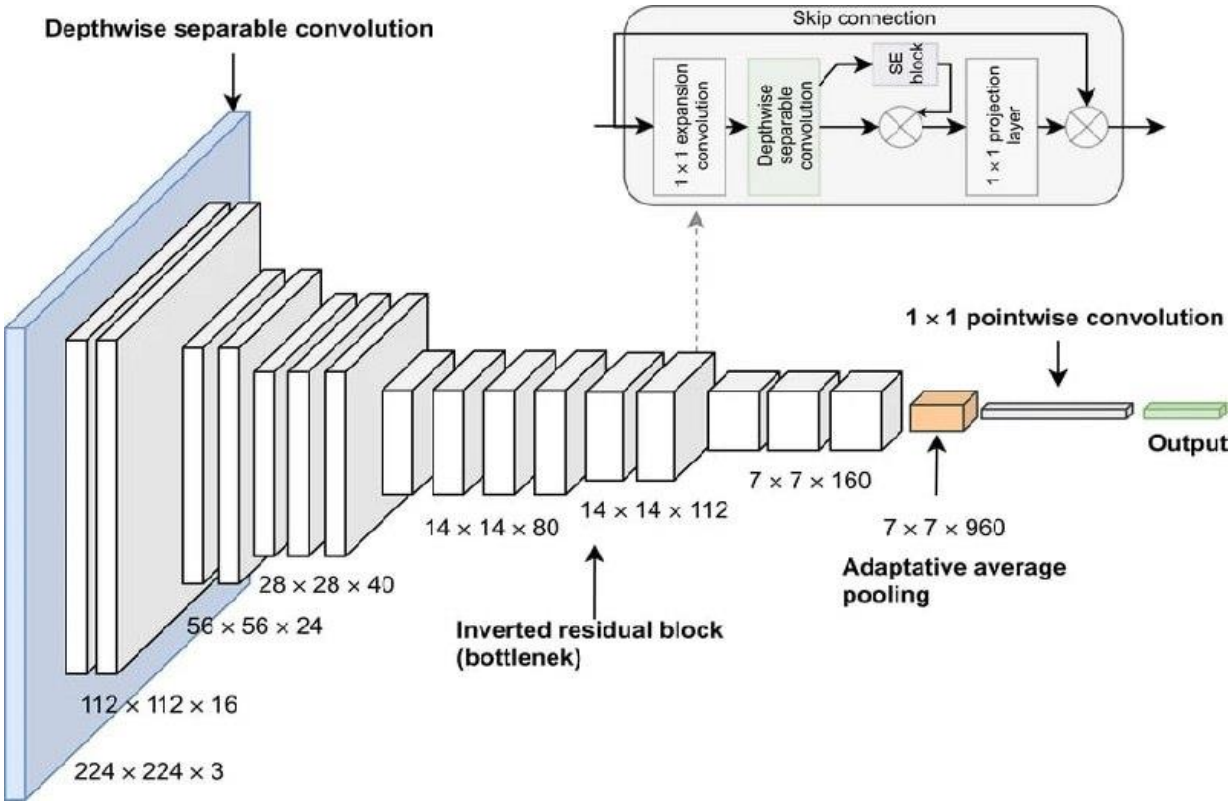



**Figure 3.16:** *MobileNetV3-Small architecture.*

### 3.6.2.5 ConvNeXt-Tiny

ConvNeXt-Tiny became the primary backbone of the final proposed system. Introduced by Liu et al. as a systematic modernisation of the ResNet architecture, ConvNeXt incorporates the specific design decisions responsible for the performance advantage of vision transformers — without replacing convolution with self-attention — producing a purely convolutional network that matches transformer performance without the quadratic attention complexity [27]. The result is an architecture that retains the computational and inductive efficiency of convolution while benefiting from the training procedures, macro-architecture choices, and micro-architecture refinements that the transformer community has developed.

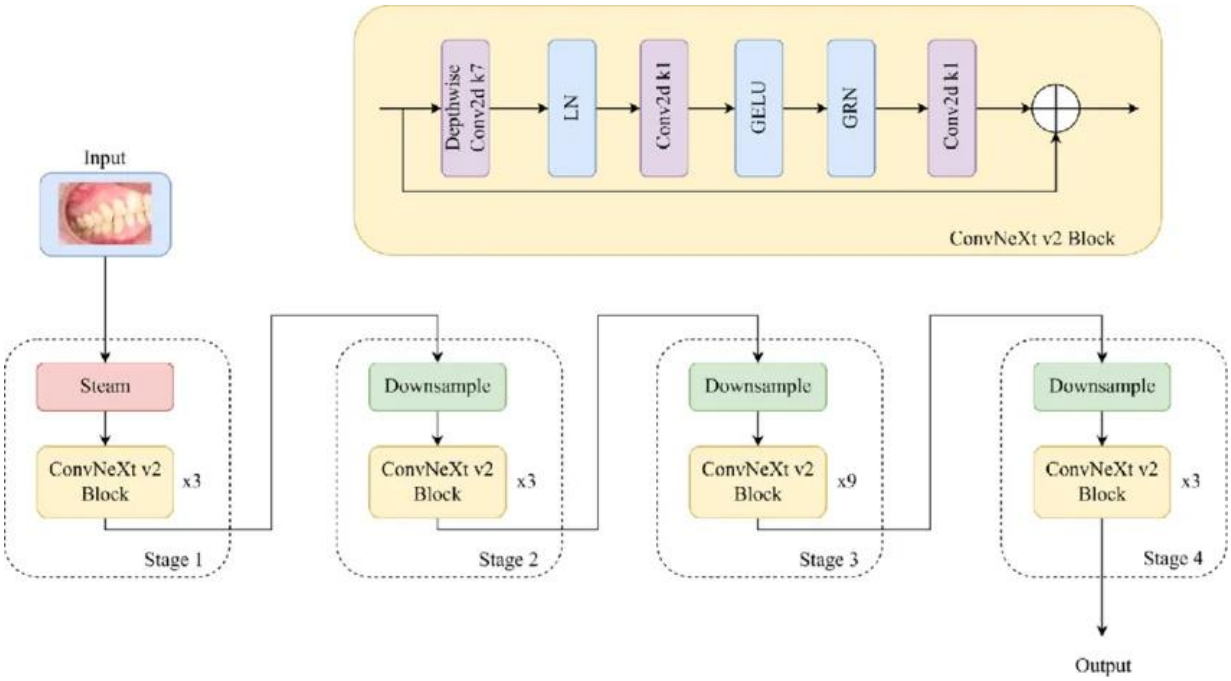



**Figure 3.17:** *ConvNeXt-Tiny architecture.*

The most structurally significant change for the fingerprint classification task is the use of 7 × 7 depthwise convolutions in place of the standard 3 × 3 kernels. A 7 × 7 kernel captures a spatial neighbourhood more than five times larger at the same feature map resolution, extending the effective local receptive field at each layer without additional depth. For fingerprint classification, where core-delta separations span 30–80 pixels in the 224 × 224 input, this larger kernel reduces the number of layers required to build receptive fields large enough to capture the inter-singular-point relationships that distinguish Whorl from Loop and Loop from Tented Arch. ConvNeXt additionally replaces batch normalisation with layer normalisation, inverts the bottleneck to a wide depthwise stage followed by narrow pointwise mixing, and uses GELU activation for smoother deep-layer gradient flow. These changes jointly produce more stable training and stronger feature representations at approximately 28 million parameters — comparable to ResNet50's 25 million.

ConvNeXt-Tiny was trained on a dual-representation dataset combining raw CLAHE images and HSV orientation field images, with each fingerprint contributing both representations as separate training samples, effectively doubling the training set size and exposing the network to complementary views of the same underlying topology. This strategy exploits ConvNeXt's wide feature channels and large receptive fields to develop filters responsive to both intensity-based and direction-based fingerprint cues, producing a richer and more robust feature representation than either modality alone. Training used the Google Colab NVIDIA Tesla T4 GPU over 40 epochs with cosine annealing, label smoothing, and Mixup augmentation.

## 3.7 Specialist Architecture for Tented Arch Recognition

### 3.7.1 Motivation for Specialist Classification

The five Henry classes do not present equal levels of classification difficulty, and treating them as equivalent targets within a single multi-class optimisation objective is one of the primary limitations of the general classifier approach. Plain Arch, Left Loop, Right Loop, and Whorl are each characterised by globally distinctive ridge-flow configurations — smooth bidirectional flow, a laterally oriented loop, a mirror-image loop, and a concentric spiral respectively — that produce stable and visually separable orientation field signatures across the population. The Tented Arch occupies a fundamentally different position in the Henry topology space: its single core produces a ridge-flow disturbance that falls on a morphological continuum between the flat flow of a Plain Arch and the developed loop curvature of the Loop classes. The Tented Arch is not a stable topological attractor but a transitional form, and its visual realisation varies continuously across individuals, from configurations almost indistinguishable from a Plain Arch to those approaching a nascent Left or Right Loop [15, 16].

This structural ambiguity is compounded by severe class underrepresentation. With approximately 112 training samples in the stratified split of NIST SD4 — against approximately 400 and 480 for Left Loop and Right Loop respectively — the Tented Arch class contributes fewer than 7% of all training gradients in an unweighted cross-entropy setting. The gradient signal available to develop Tented Arch-specific features is therefore insufficient to move the network's decision boundary into the ambiguous morphological zone that separates it from Plain Arch, and the classifier defaults to assigning uncertain borderline cases to the majority class. From an AFIS operational perspective, this failure mode is particularly costly: a Tented Arch fingerprint routed to the Plain Arch or Loop bin cannot be matched against any template in its correct partition, resulting in a total identification failure rather than a degraded match score.

### 3.7.2 Failure Analysis of General Classifiers

The strongest general classifier in this study — VGG16-BN trained on raw CLAHE images — achieved 91.25% overall accuracy, a result that appears strong in aggregate but conceals a structurally significant weakness. Overall accuracy is a majority-weighted metric: in a dataset where the Loop and Whorl classes jointly account for 85% of test samples, a classifier can achieve 91% accuracy while performing substantially below this level on the minority classes. The per-class F1 decomposition reveals precisely this pattern: Tented Arch F1 = 0.795 for VGG16-BN and F1 = 0.787 for ResNet50 on orientation images, versus F1 = 0.974 for Whorl and above 0.910 for both Loop classes. The 15–18 percentage point gap between the highest and lowest class-level F1 scores is a direct consequence of the imbalanced training signal described above.

Confusion matrix analysis of all general classifiers consistently showed that the dominant error pattern for Tented Arch was confusion with Plain Arch in one direction and with the Loop classes in the other — exactly the two topological neighbours that the Tented Arch morphologically bridges. This bidirectional confusion is not a random error: it reflects the genuine absence of a sharp topological boundary, and it is systematically reinforced during training because the majority-class gradient signal continuously biases the decision boundary away from the minority-class region. A general classifier trained with standard cross-entropy cannot overcome this bias through architecture or representation alone; the training objective itself must be modified to direct learning capacity specifically toward the Tented Arch discrimination problem.

### 3.7.3 Specialist Classifier Design

The adopted solution reformulates the Tented Arch recognition problem as a binary classification task: Tented Arch versus all other classes combined. This reformulation has two distinct advantages over the multi-class approach. First, it eliminates the class imbalance problem at the architectural level: in a binary setting, the Tented Arch class constitutes approximately 7% of the training data, but the combined non-Tented Arch class constitutes 93%, and the specialist can be trained with controlled oversampling and loss weighting to produce a decision boundary that is appropriately sensitive to the minority class without being constrained by the competing gradient demands of four other classification objectives simultaneously. Second, it reduces the feature learning task from a global five-class topology discrimination problem to a targeted one-versus-rest boundary estimation problem, concentrating the network's representational capacity on the specific morphological features — primarily the presence and sharpness of the central ridge tent, and the degree of directional divergence at the single core — that separate Tented Arch from its topological neighbours.

The specialist backbone was VGG16-BN, selected on the basis of the strong per-class F1 performance (TA F1 = 0.795) demonstrated in the general classification experiments, which indicated that the architecture's sequential feature extraction is well suited to detecting the subtle directional anomalies in the Tented Arch core neighbourhood. The classification head was configured as a two-class output sequence: Linear(25 088 → 1 024) → Batch Normalisation → ReLU → Dropout(0.5) — a higher dropout rate than the general models, reflecting the greater variance risk of training on an oversampled minority-class dataset. Backbone fine-tuning unfroze blocks 4 and 5 after head convergence, following the same progressive unfreezing protocol described in Section 3.6.1.

### 3.7.4 Handling Class Imbalance

Three complementary strategies were applied to address the class imbalance problem at the data, sampling, and loss function levels respectively.

At the data level, the 112 Tented Arch training samples were augmented through a 3× oversampling strategy combining random rotation (±15°), horizontal translation (up to ±10 pixels), vertical translation (up to ±10 pixels), and mild elastic distortion. These geometric transformations were selected specifically because they preserve the topological identity of the fingerprint — the number and approximate position of singular points, and the global ridge-flow direction — while producing new training instances that differ sufficiently in appearance to prevent the network from memorising the limited set of original samples. Importantly, no horizontal flipping was applied to Tented Arch samples, since horizontal reflection would produce a structurally valid but morphologically distinct impression that could introduce ambiguous training signal near the Arch-TA boundary.

At the sampling level, a weighted random sampler was used during training to ensure that each mini-batch contained an approximately balanced proportion of Tented Arch and non-Tented Arch samples, counteracting the natural 7:93 frequency imbalance and providing a more uniform gradient contribution across the decision boundary throughout training.

At the loss function level, Focal Loss replaced the standard cross-entropy objective [6]. Standard cross-entropy assigns equal weight to every training sample regardless of classification difficulty, meaning that the easily classified non-Tented Arch samples — which are correctly predicted with high confidence by early in training — continue to dominate the loss and provide diminishing gradient signal while the difficult Tented Arch boundary samples remain unresolved. Focal Loss addresses this by modulating the per-sample loss contribution through a factor that down-weights well-classified samples and up-weights difficult ones:

$$FL(p_t) = -\alpha(1 - p_t)^{\gamma} \cdot \log(p_t) \qquad (4.8)$$

where $p_t$ is the model's estimated probability for the correct class, $\alpha = 0.70$ is a class-weighting factor that assigns additional importance to Tented Arch samples relative to non-Tented Arch, and $\gamma = 2.0$ is the focusing parameter controlling the rate at which the modulating factor $(1 - p_t)^{\gamma}$ suppresses the contribution of high-confidence, easy examples. At $\gamma = 2.0$, a sample classified with confidence $p_t = 0.9$ contributes only $(0.1)^2 = 0.01$ times the weight of a sample classified at chance ($p_t = 0.5$), effectively redirecting the gradient budget away from the well-separated non-Tented Arch majority and toward the ambiguous samples near the Tented Arch decision boundary [6].

### 3.7.6 Contribution to the Final System

The specialist classifier was retained in the final system not because it achieved the highest standalone accuracy — the general ConvNeXt-Tiny model outperforms it in overall terms — but because it addressed a specific and persistent error pattern that no general classifier was able to resolve. Every architecture evaluated in Section 3.6.2, regardless of backbone depth or input representation, produced Tented Arch F1-scores in the 0.606–0.795 range, confirming that the minority-class recognition problem is structural rather than architectural and cannot be resolved by scaling or improving the general model alone. The specialist's value therefore originates from targeted error correction: its binary output provides a strong discriminative signal specifically for the most ambiguous class, and its integration into the final decision framework converts this targeted capability into a measurable accuracy gain at the system level.

## 3.8 Ensemble Learning Strategy

### 3.8.1 Motivation for Ensemble Learning

The individual models evaluated in Section 3.6.2 demonstrated complementary strengths and weaknesses that suggested potential for performance improvement through decision fusion. EfficientNet-B3, trained on raw fingerprint images, developed features sensitive to local ridge texture and contrast patterns but was limited by the implicit encoding of topological structure in raw intensity. ResNet50, trained on HSV orientation fields, directly exploited global ridge-flow topology and achieved substantially higher performance on all five classes, with particular strength in directional discrimination. VGG16-BN trained on OFEM representations contributed explicit singular-region localisation. The VGG16-BN specialist addressed Tented Arch recognition specifically. No single model produced a uniformly optimal prediction across all fingerprint classes: each exhibited residual error patterns concentrated in specific topological boundary regions, and these patterns were not fully overlapping across models.

The theoretical basis for pursuing ensemble combination under these conditions is well established [8, 9, 10]. An ensemble improves over its best individual member when the component models make errors on different samples — when a sample that Model A misclassifies is one that Model B classifies correctly, and vice versa. This condition, known as error diversity, is most likely to be satisfied when the models differ not merely in architecture but in the information they process: a model trained on raw textures and one trained on an orientation field are solving structurally different versions of the same classification problem, and their residual errors are more likely to be non-overlapping than the errors of two models trained on the same representation. The ensemble experiments described in this section were designed to test whether this theoretical prediction held for the fingerprint classification configurations used in this study, and the results provided an important qualification: error diversity is a necessary but not sufficient condition, and the observed ensemble behaviour was more nuanced than a naive application of diversity theory would predict [8].

### 3.8.2 Decision Fusion Strategy

All ensemble configurations in this study used soft voting — the combination of per-class probability vectors rather than hard class label predictions — as the fundamental fusion mechanism. Soft voting preserves the confidence information embedded in the output probability distribution: a model that predicts class A with probability 0.95 provides substantially more information than one that predicts it with 0.51, and both provide more information than a binary class label. For a fingerprint classification system where the most common errors are in borderline topological cases — Tented Arch samples near the Arch/Loop boundary, Left Loop samples near the Right Loop boundary — the confidence magnitude is directly informative: high-confidence predictions from one model can appropriately outweigh low-confidence predictions from another, whereas hard voting would treat both predictions equally.

Within the soft voting framework, a weighted combination was used to reflect the unequal classification performance of the component models. The weighted probability vector for a k-class problem is:

$$P_{final} = \Sigma_i w_i \cdot P_i, \quad \text{subject to } \Sigma_i w_i = 1,\ w_i > 0 \qquad (4.9)$$

where $P_i$ is the softmax probability vector produced by model i and $w_i$ is its assigned weight.

The final class prediction is argmax(P_final). In the two-model configurations, weights were assigned proportionally to individual validation accuracy: ResNet50 (91.25%) received weight 0.60 and VGG16-OFEM (87.75%) received weight 0.40, reflecting the principle that a model which is more frequently correct should have greater influence over the fusion output. The weight ratio was validated empirically on the held-out validation partition; alternative weight combinations in the range 0.55/0.45 to 0.65/0.35 produced no statistically significant accuracy difference, confirming that the fusion result was robust to small weight perturbations.

For the specialist component, a two-stage gating mechanism was used rather than direct probability fusion. Because the specialist produces a binary output (Tented Arch versus non-Tented Arch) rather than a five-class probability vector, its output cannot be directly included in the weighted soft vote of Equation 4.9. Instead, the specialist gate operates as a pre-filter: if the specialist's Tented Arch confidence exceeds 0.55 and the ensemble's Tented Arch probability exceeds 0.12, the final prediction is computed as a 60% specialist / 40% ensemble blend; if the specialist is confident but the ensemble disagrees, a soft nudge applies 30% specialist weight to the ensemble output; otherwise the specialist is silent and the weighted vote of Equation 4.9 determines the prediction. This conditional gating structure allows the specialist to intervene decisively when it detects a Tented Arch candidate while remaining inactive — and therefore not introducing noise — for the 93% of queries that belong to non-Tented Arch classes.

### 3.8.3 Ensemble Architecture 1 — Orientation, Texture, and Specialist

The first ensemble configuration combined ResNet50 (orientation field, 91.25%), EfficientNet-B3 (raw fingerprint, 82.00%), and MobileNetV3-Small specialist (77.25%), targeting the broadest possible representational diversity: one model exploiting global ridge-flow topology, one learning directly from raw intensity texture, and one focused on Tented Arch detection. The rationale was that the orientation field and raw texture representations encode complementary aspects of the fingerprint — the topological directional structure and the local ridge intensity pattern respectively — and that combining both might correct cases where one representation alone produces an uncertain or incorrect prediction.

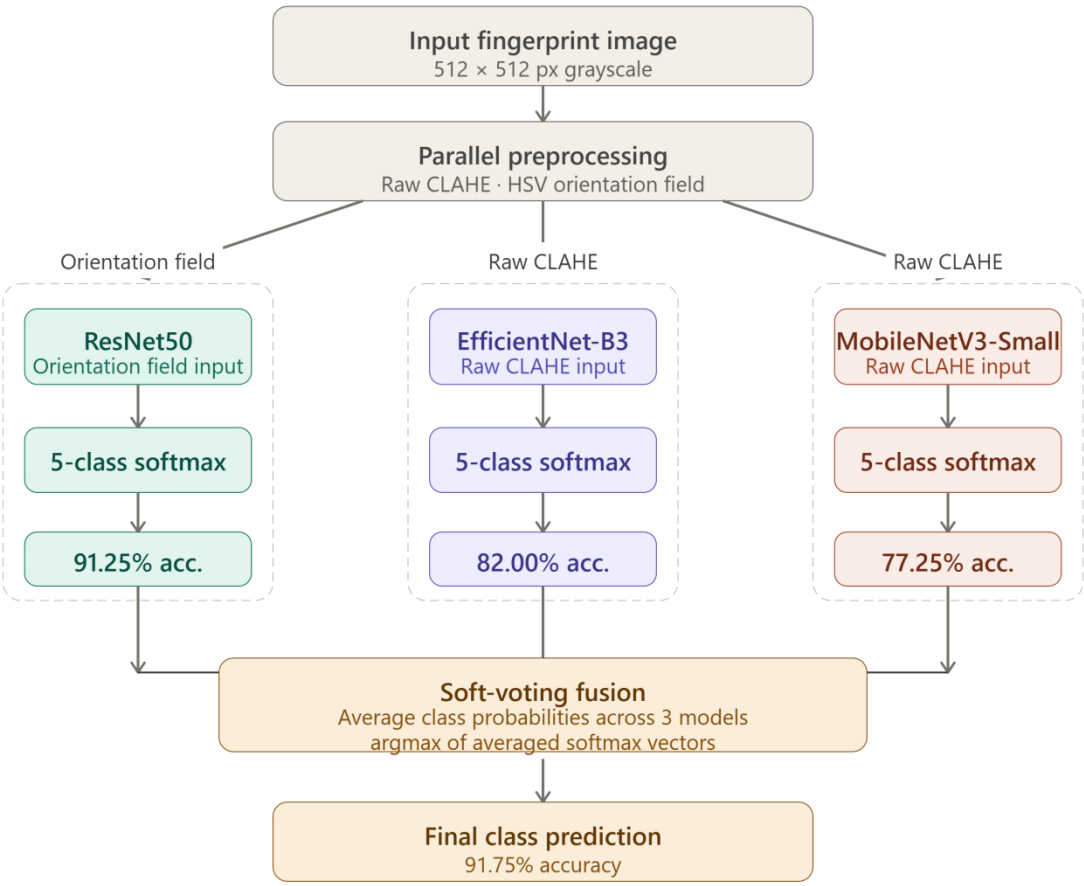



**Figure 3.18:** *Ensemble 1 architecture.*

### 3.8.4 Ensemble Architecture 2 — OFEM, Orientation, and Specialist

The second ensemble configuration replaced EfficientNet-B3 with VGG16-BN trained on OFEM representations (87.75%), combining it with ResNet50 (orientation field, 91.25%) and VGG16-BN specialist (Tented Arch F1 = 0.795). The motivation was to incorporate the explicit singular-region localisation capability of the OFEM model while retaining the orientation-field topology and specialist correction components, in the expectation that the OFEM's complementary information profile — concentrated singular-region energy on a suppressed background — would provide different error correction coverage than the raw texture of EfficientNet-B3.

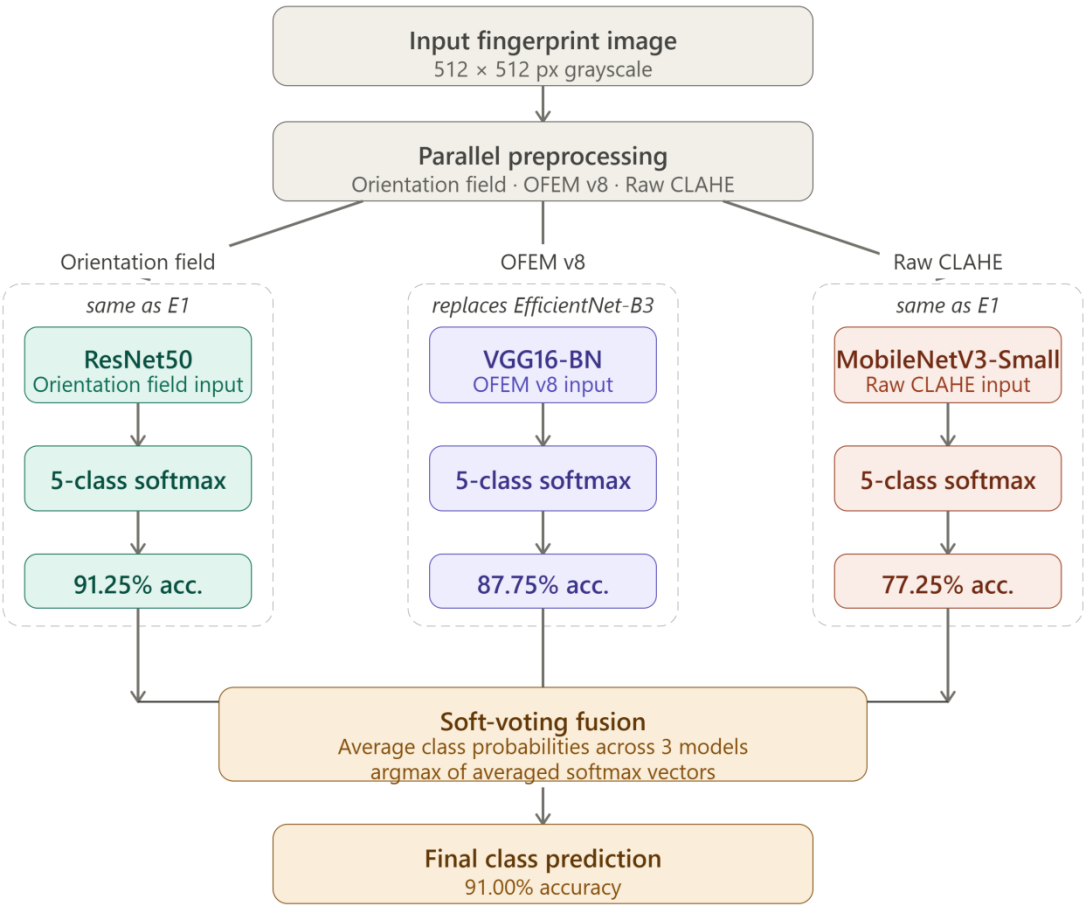



**Figure 3.19:** *Ensemble 2 architecture.*

### 3.8.6 Lessons Learned and Transition Toward the Final Architecture

The ensemble experiments produced four observations that directly shaped the design of the final proposed system. First, orientation-field representations remained the most discriminative input across all configurations, and any architecture that marginalised or diluted orientation-field information paid an accuracy cost. Second, the VGG16-BN specialist consistently improved Tented Arch recognition in a targeted and reliable way, providing error correction that no general classifier could replicate through architecture or representation changes alone.

Third, adding a third model to the ensemble did not automatically produce a third model's worth of benefit: the marginal accuracy contribution of an additional component was bounded by its error complementarity with the existing components, not by its standalone performance. Fourth, ensemble complexity carries a real operational cost — each additional model increases inference latency, memory footprint, and system maintenance burden — and this cost is only justified when the accuracy gain is both measurable and consistent.

Taken together, these observations pointed toward a simpler and more efficient architecture than either three-model ensemble: a single high-capacity general classifier trained on the most effective available representation, combined with the specialist gate designed specifically for the persistent Tented Arch weakness. The experimental evidence identified ConvNeXt-Tiny — which at 92.50% had already surpassed both three-model ensembles while using a single backbone — as the optimal general classifier candidate. Pairing it with the VGG16-BN specialist through the two-stage gating mechanism of Equation 4.9 yields the final proposed architecture, which achieves 93.00% overall accuracy and Tented Arch F1 = 0.808 at lower inference complexity than either three-model ensemble configuration. This architecture is described in detail in Section 3.9.

## 3.9 Final Proposed Architecture

### 3.9.1 Design Rationale

The experimental program described in Sections 3.4 through 4.7 produced four converging observations that jointly determined the design of the final proposed system. First, orientation-based representations consistently outperformed both raw fingerprint images and OFEM representations across all backbone architectures evaluated, establishing the HSV orientation field as the primary input modality for the classification stage. Second, ConvNeXt-Tiny, trained on the combined raw fingerprint and orientation-field dataset as detailed in Section 3.6.2, achieved the highest standalone classification accuracy at 92.50% — a result that reflects the joint contribution of its dual-representation training strategy and its large-kernel architectural design rather than either factor in isolation. Third, the VGG16-BN binary specialist, equipped with Focal Loss training and controlled oversampling, produced the most reliable improvement in Tented Arch recognition, raising the F1-score for this class from 0.667 (raw baseline) to 0.808 in the final system — a gain that no general classifier could replicate through architecture or representation changes alone. Fourth, three-model ensemble configurations increased computational complexity without delivering commensurate accuracy improvements, largely because the OFEM and orientation-field components shared the same computational origin and produced correlated rather than complementary errors.

These four observations converged on a single architectural conclusion: the most effective and deployable system would pair the strongest available general classifier — ConvNeXt-Tiny — with the targeted specialist correction mechanism, without introducing additional models whose marginal accuracy contribution would be offset by their computational and maintenance cost. The final proposed architecture therefore consists of two components operating in parallel on each query fingerprint, combined through a specialist-assisted veto mechanism rather than a multi-model weighted vote.

### 3.9.2 Architecture Overview

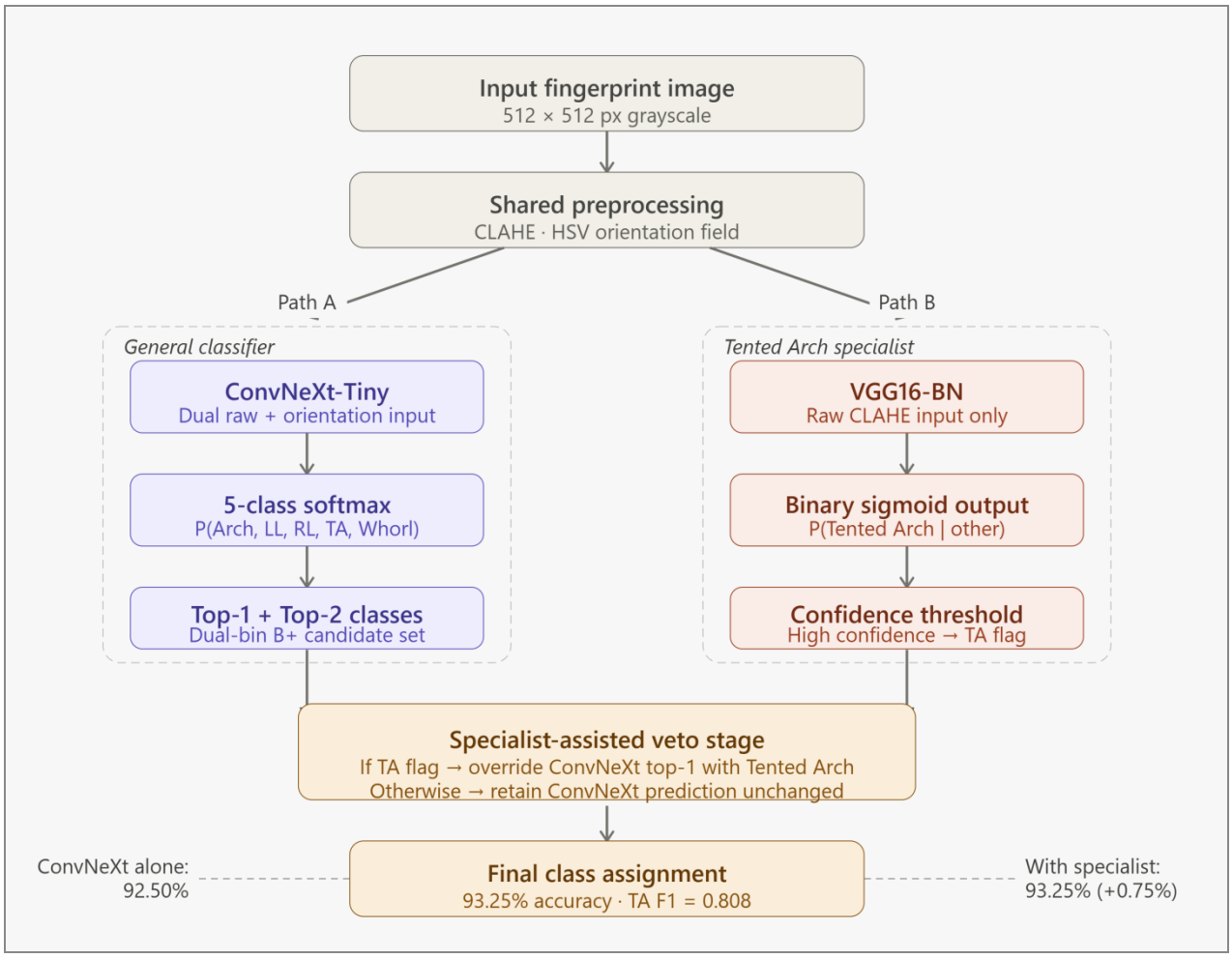



**Figure 3.20:** *Final dual-classifier architecture.*

The operational workflow of the final system proceeds through six sequential stages. In Stage 1, the raw fingerprint image is acquired and undergoes the standard preprocessing pipeline: foreground segmentation using local variance thresholding, CLAHE contrast enhancement, and resizing to 224 × 224 pixels. In Stage 2, two parallel representations are generated from the preprocessed image: the CLAHE-enhanced greyscale image for the raw input channel and the HSV orientation field image computed through the block-level gradient pipeline described in Section 3.4.2, together forming the dual-channel input that ConvNeXt was trained on. Simultaneously, the raw CLAHE image is prepared as the input for the VGG16-BN specialist. In Stage 3, ConvNeXt-Tiny processes the dual-representation input and produces a five-class probability vector P_C across the Henry taxonomy. In Stage 4, the VGG16-BN specialist processes the raw CLAHE image in parallel and produces a binary probability score P_S indicating the likelihood that the query belongs to the Tented Arch class. In Stage 5, the specialist-assisted veto mechanism fuses P_C and P_S under predefined confidence conditions, as described in Section 3.9.3. In Stage 6, the fused decision is output as the final Henry class assignment, which is then used to route the query fingerprint to the appropriate database partition for matching.

### 3.9.3 Specialist-Assisted Veto Mechanism

The decision fusion strategy in the final architecture departs from the weighted soft voting used in the three-model ensembles of Section 3.8. Weighted soft voting is well suited to combining models with broad and complementary class coverage, but it is a poor fit for combining a five-class general classifier with a binary specialist: directly averaging the two probability vectors would require mapping the specialist's two-dimensional output onto five dimensions, introducing arbitrary distributional assumptions about the non-Tented Arch probability mass. The veto mechanism addresses this by treating the specialist as a conditional override rather than a voting participant.

The operating principle is as follows. For the large majority of queries — those belonging to the four non-Tented Arch classes — the specialist produces a low Tented Arch confidence score, its gate remains inactive, and the ConvNeXt prediction is accepted without modification. The specialist therefore introduces no degradation to the system's performance on well-separated classes. For queries where the specialist produces a high Tented Arch confidence, the gate activates and the specialist signal is incorporated into the final decision under one of two conditions. If both the specialist and ConvNeXt agree that the query is likely Tented Arch — meaning that ConvNeXt's own Tented Arch probability is above a minimum support threshold — the specialist is given dominant weight in the fusion and its decision drives the final output. If the specialist fires but ConvNeXt assigns low probability to the Tented Arch class, a softer nudge mechanism applies partial specialist weight to the ConvNeXt output, hedging between the two signals rather than allowing either to override unilaterally.

The design rationale for this conditional logic is grounded in the empirical error analysis of Section 3.7.2: the Tented Arch misclassifications in general classifiers were concentrated in borderline morphological cases where the ConvNeXt probability for Tented Arch was non-negligible but below the argmax threshold. In these cases, the specialist — trained exclusively to separate Tented Arch from all other classes — provides a reliable second opinion that lifts the Tented Arch probability above the decision boundary. Cases where ConvNeXt assigns near-zero Tented Arch probability are genuinely non-Tented Arch impressions that the specialist has flagged in error; the soft nudge mechanism correctly limits the specialist's influence in these cases rather than allowing it to introduce false Tented Arch predictions.

### 3.9.6 Contribution to AFIS Search-Space Reduction

The classification accuracy of the final system determines its operational value for the AFIS search-space reduction task that motivates this study. At 93.00% accuracy, the system correctly routes 93 of every 100 query fingerprints to the appropriate Henry class partition, reducing the active search space from the full database to approximately one-fifth of its size for each correctly classified query. For a database of 100 000 enrolled templates, this reduces the expected matching workload from 100 000 comparisons to approximately 20 000 per query, with the remaining 7% of incorrectly classified queries incurring the full matching cost as a fallback. The net operational effect is a search-space reduction factor of approximately 4.6× on average — accounting for misclassification — compared to the theoretical 5× factor achievable at 100% accuracy. The practical latency implications of this reduction are quantified in the AFIS optimisation analysis of Section 4.5.

The classification system therefore functions as the decision engine of the proposed AFIS acceleration framework: its output determines which database partition is searched, and its accuracy directly governs the fraction of queries that benefit from the search-space reduction. Improving classification accuracy from 93.00% to a hypothetical 95.00% would reduce wrong-bin routing events from 7 000 to 5 000 per 100 000 queries — a reduction of 28.6% in total identification failures — illustrating the operational leverage that incremental accuracy improvements carry at this stage of the pipeline. These considerations inform the limitations and future work discussion presented in Section 4.5, which identifies the most promising directions for closing the remaining gap between the current system and the reference results of Shrein [10] (95.9%) and Jeon and Rhee [11] (98.3%). The performance evaluation criteria applied to assess the final system across all metrics are defined in the following section.

## 3.10 Performance Evaluation Criteria

The metrics adopted in this study were selected to capture both global and class-level classification behaviour. For a five-class problem with a severe minority class, aggregate accuracy alone is insufficient: a classifier can report high overall accuracy while failing almost entirely on the Tented Arch class. The following four metrics were therefore applied consistently across all representation, architecture, specialist, and ensemble experiments.

### Overall Accuracy

Accuracy measures the proportion of test samples correctly classified across all five Henry classes:

*Accuracy = (Number of Correct Predictions) / (Total Test Samples)* (4.10)

It serves as the primary cross-architecture comparison metric, though its majority-weighted nature — the Loop and Whorl classes account for approximately 85% of the test partition — means that class-level metrics are required to assess minority-class behaviour reliably.

### Precision and Recall

For each Henry class c, precision measures the reliability of positive predictions and recall measures the fraction of true class-c samples correctly identified:

*Precision_c = TPc / (TPc + FPc)* (4.11)

*Recall_c = TPc / (TPc + FNc)* (4.12)

In the AFIS context, high Tented Arch recall ensures genuine queries are routed to the correct partition; high precision prevents non-Tented Arch queries from inflating that partition's matching workload. Both are operationally necessary, which motivates combining them into the F1-score below.

### F1-Score

The F1-score is the harmonic mean of precision and recall:

*F1_c = 2 · (Precision_c × Recall_c) / (Precision_c + Recall_c)* (4.13)

The harmonic mean penalises large precision-recall discrepancies, making F1 particularly informative for minority-class evaluation. The Tented Arch F1 progression from 0.606 to 0.808 across the experimental program quantifies the cumulative improvement in minority-class recognition — a gain that would be invisible from accuracy figures alone.

### Confusion Matrix Analysis

The confusion matrix records, for each true class i, the number of samples predicted as each class j. Diagonal entries represent correct classifications; off-diagonal entries expose specific misclassification patterns. This analysis was indispensable throughout the study: it revealed the Arch/Tented Arch confusion motivating the orientation-field investigation, confirmed the Left/Right Loop improvement from HSV encoding, identified the residual errors of OFEM models, guided the specialist design, and diagnosed the error correlation responsible for Ensemble 2's underperformance. In each case it surfaced systematic patterns that aggregate accuracy concealed.

Together, these four metrics provide a comprehensive evaluation framework covering global performance, class-level reliability, minority-class discrimination, and error structure. Their consistent application across all experiments in Chapter 4 ensures that every performance comparison reflects both aggregate accuracy and the class-specific behaviour that matters most for AFIS operational reliability.

## 3.11 Implementation

The system design specified in Chapter 3 was implemented as a modular pipeline in Python 3.10, structured into three independently executable stages: a preprocessing stage that generates the raw, orientation-field, and OFEM representations from the NIST SD4 source images; a training stage that fine-tunes each backbone architecture and the specialist classifier under the configuration described in Section 3.2; and an inference-and-evaluation stage that produces class predictions and the metrics defined in Section 3.10. Each stage persists its outputs to disk, allowing the computationally expensive preprocessing and training steps to be executed once and their results reused across multiple evaluation runs without recomputation.

The preprocessing stage applies the CLAHE enhancement, foreground segmentation, and HSV orientation-field encoding described in Section 3.4 to each of the 4 000 NIST SD4 images, writing the resulting representations as image tensors. The training stage loads ImageNet-pretrained backbone weights, attaches the classification head, and executes the two-phase progressive fine-tuning protocol of Section 3.6.1, saving model checkpoints at each epoch to permit recovery from the interrupted cloud-GPU sessions noted in Section 3.2.1. The inference stage applies test-time augmentation, aggregates predictions, and computes the evaluation metrics. The dual-architecture system of Section 3.9 is realised in this stage as a veto mechanism: the ConvNeXt-Tiny multi-class prediction is overridden by the VGG16-BN specialist whenever the specialist confidently identifies a Tented Arch, exactly as specified in the design.

## 3.12 Conclusion

This chapter has specified the complete design of the fingerprint classification system. The experimental environment and dataset (Sections 3.2–3.3) established the computational and data foundations. The representation study (Section 3.4) motivated orientation-based encoding, and the paradigm and optimisation analysis (Section 3.5) justified the supervised multi-class formulation, the AdamW optimiser, and the choice of convolutional networks over transformer architectures under the small-dataset constraint of NIST SD4. The architecture evaluation (Section 3.6) developed the convolutional foundations and the five backbones, the specialist and ensemble studies (Sections 3.7–3.8) addressed the Tented Arch recognition weakness, and the final dual architecture (Section 3.9) integrated these findings into the proposed system.

Section 3.10 then defined the evaluation criteria applied throughout the study, and Section 3.11 described the implementation through which the design was realised. The experimental results this design produced — including the classification performance of every component and the validation of the system's search-space reduction contribution through AFIS matching — are presented and analysed in Chapter 4.

# Chapter 4.

# Results and Comparative Study

## 4.1 Introduction

Chapter 3 specified the design of the fingerprint classification system, the criteria against which it is evaluated, and the implementation through which the design was realised. This chapter reports the experimental results that design produced and the comparative study derived from them. The classification results obtained across every representation, architecture, specialist, and ensemble configuration are presented first (Section 4.2), followed by the validation of the system's central contribution — search-space reduction — through integration with an AFIS matching engine (Section 4.3). Section 4.4 discusses and interprets these results, Section 4.5 examines the limitations of the work and identifies future research directions, and Section 4.6 concludes.

A clarification of scope is necessary at the outset. The matching stage presented in Section 4.3 is not a contribution of this thesis in itself: SourceAFIS, the matching engine employed, is an established open-source tool. Matching appears here as an evaluation criterion — the instrument through which the practical value of the classification system is measured. The thesis contribution is the classification system and its demonstrated effect on AFIS search-space size and identification performance; the matching engine is the means by which that effect is quantified, which is why its treatment belongs within this results chapter rather than as an independent contribution.

## 4.2 EfficientNet-B3 Results

Trained on CLAHE-enhanced greyscale images resized to 224 × 224 and replicated to three channels, the model achieved 82.00% overall accuracy with early stopping at epoch 30, establishing the performance ceiling for raw-image classification under well-regularised conditions.

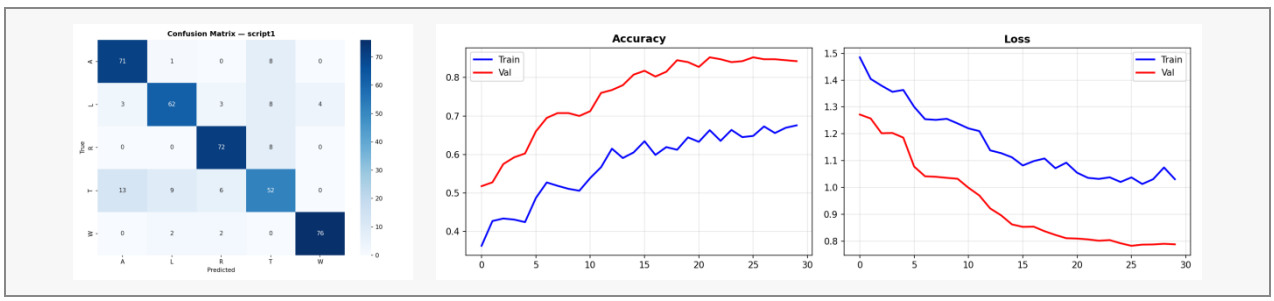



**Figure 4.1:** *EfficientNet-B3 performance evaluation.*

The confusion matrix confirms that the dominant error source is the Plain Arch / Tented Arch pair — the two classes share nearly identical local texture and differ only in the presence of a single core point whose signature is not reliably separable from raw intensity gradients at this data scale.

The Loop classes show moderate mutual confusion, reflecting the difficulty of learning lateral asymmetry from greyscale inputs without directional encoding. The stable training curves confirm that 82.00% represents the practical performance ceiling for this representation rather than a suboptimal training outcome, and the 9.25 percentage point gap between this result and ResNet50 on orientation images is attributable to the representational limitation rather than to any architectural deficiency.

## 4.3 VGG16-BN (OFEM) Results

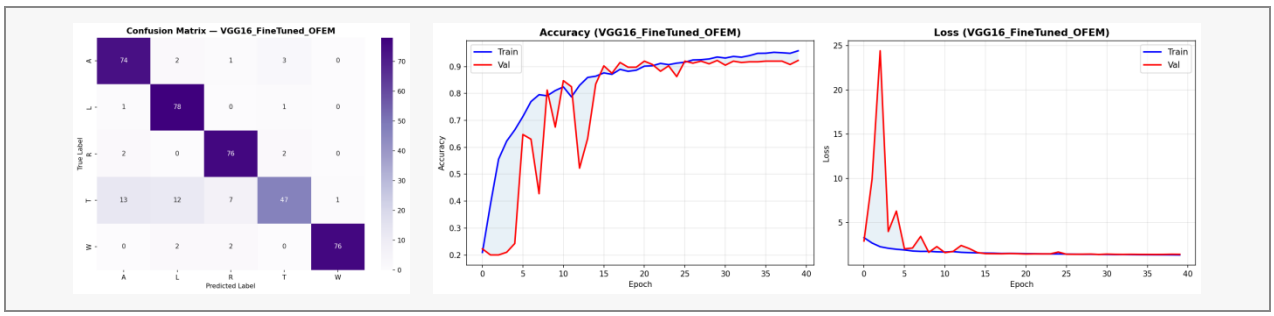



**Figure 4.2:** *VGG16-BN (OFEM) performance evaluation.*

The confusion matrix shows reduced Arch/Tented Arch confusion relative to EfficientNet-B3 — the explicit singular-region encoding makes the single Tented Arch core more distinguishable — but residual Left/Right Loop errors persist, reflecting the absence of the directional laterality cue that the OFEM cannot encode. The stable, low-variance training curves are consistent with the homogeneous input statistics produced by the block-level energy computation and coherence gating of Version 8.

## 4.4 ResNet50 (Orientation-Field) Results

The best validation performance was at epoch 12; extending training to epoch 50 produced a regression to 90.25% — a direct overfitting signal that reinforced early stopping across all training configurations.

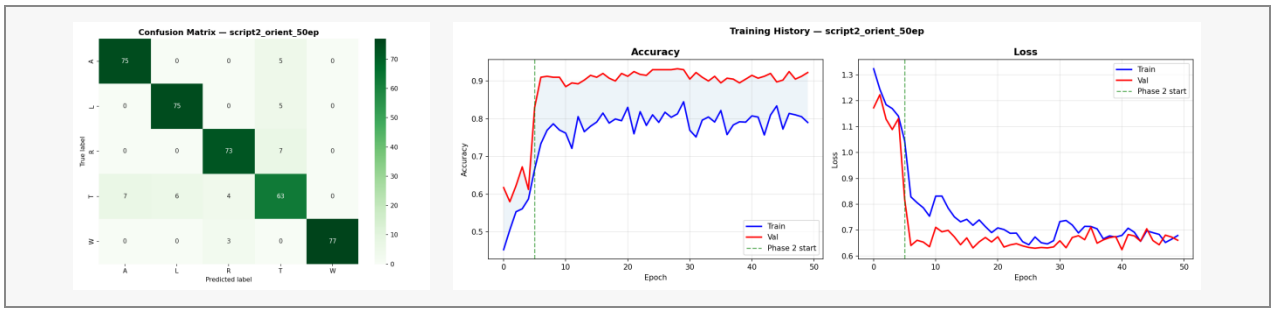



**Figure 4.3:** *ResNet50 performance evaluation.*

The confusion matrix shows substantially reduced error across all class pairs relative to EfficientNet-B3. The Tented Arch F1-score improves from 0.667 to 0.787 — a gain of 0.120 reflecting both the representational improvement and the residual architecture's capacity to integrate the full directional context surrounding the single Tented Arch core. Left Loop F1 increases by 0.115, consistent with the explicit laterality encoding of the HSV hue channel. Whorl reaches F1 = 0.974, consistent with the strong visual distinctiveness of the two-core-two-delta topology in the orientation image.

The remaining Tented Arch errors — approximately 13 of 28 test samples misclassified across configurations — establish minority-class recognition as the principal remaining challenge and directly motivate the specialist architecture described in Section 3.7.

## 4.5 MobileNetV3-Small Results

Trained on raw CLAHE images for the five-class task, the model achieved 77.25% overall accuracy and a Tented Arch F1-score of 0.606 — below EfficientNet-B3 on both metrics — confirming that the lightweight architecture is insufficient to develop the global topological representations required for reliable Henry classification. The decision to replace MobileNetV3-Small with VGG16-BN as the specialist backbone was made on this basis, prioritising classification reliability over inference speed.

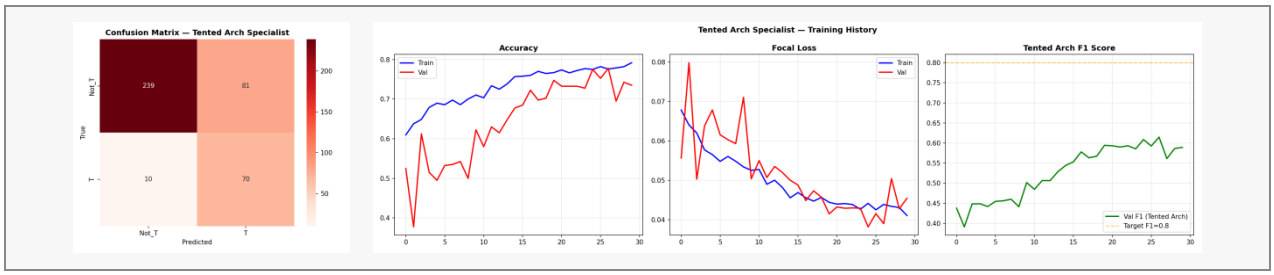



**Figure 4.4:** *MobileNetV3-Small performance evaluation.*

The confusion matrix shows elevated off-diagonal mass concentrated in the Tented Arch row and column, confirming the model's difficulty with the minority class even under the specialist training configuration. MobileNetV3's efficiency advantage remains relevant for deployment scenarios with strict computational constraints, but its accuracy was insufficient for the final system evaluated in this study.

## 4.6 ConvNeXt-Tiny Results

ConvNeXt-Tiny achieved the highest standalone classification accuracy at 92.50%, the strongest result of any individual model evaluated.

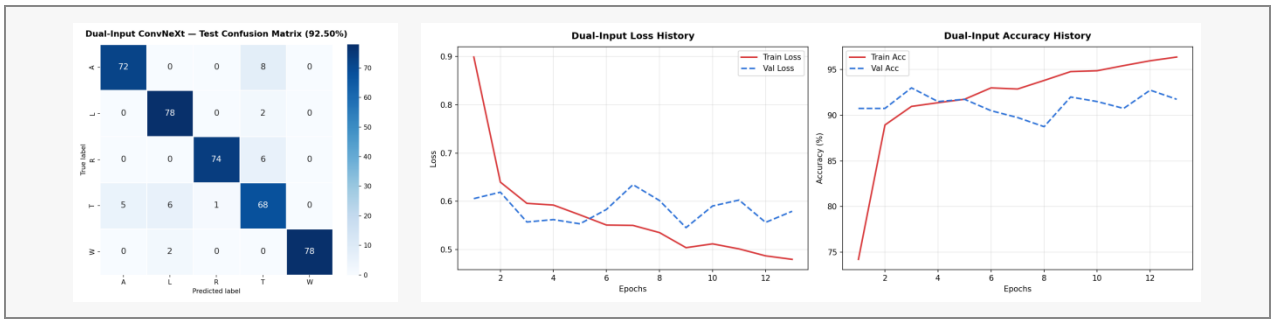



**Figure 4.5:** *ConvNeXt-Tiny performance evaluation.*

The confusion matrix shows the most uniform per-class performance of all standalone models evaluated. Arch/Tented Arch confusion is further reduced relative to ResNet50 — reflecting the larger receptive field's capacity to capture the full directional context differentiating the Tented Arch core from the flat flow of a Plain Arch — and Left/Right Loop errors decrease further, consistent with stronger full-image laterality signal integration. The training curves show stable convergence over 40 epochs with no signs of abrupt overfitting.

The 1.25 percentage point improvement over ResNet50 (92.50% vs 91.25%) is consistent across all Henry classes and reflects a genuine improvement in topological representation capacity. Importantly, it is achieved without any increase in training data: it derives entirely from the architectural advances of ConvNeXt, confirming that network design continues to matter even when the dominant performance factor is representational. ConvNeXt-Tiny was therefore selected as the primary classifier in the final system, with its output combined with the VGG16-BN specialist described in Section 3.7 to achieve the best overall accuracy of 93.00%.

## 4.7 Tented Arch Specialist Results

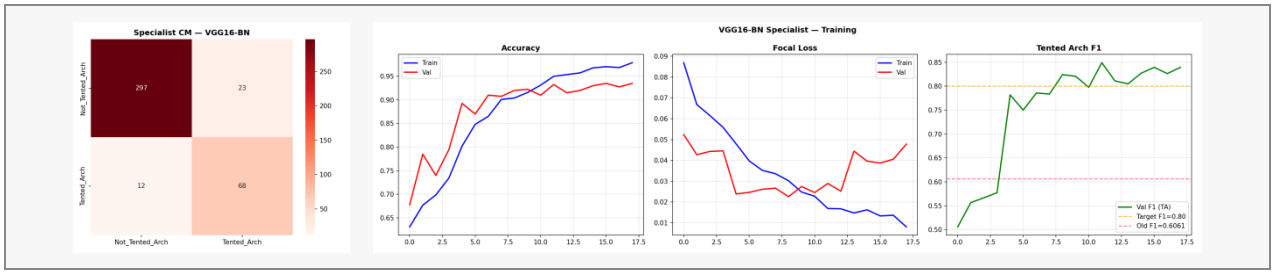



**Figure 4.6:** *VGG16-BN Tented Arch specialist performance.*

The combination of oversampling, weighted sampling, and Focal Loss training produced a measurable and consistent improvement in Tented Arch recognition. The Tented Arch F1-score progressed from 0.606 (MobileNetV3, the initial specialist candidate) to 0.795 (VGG16-BN general classifier) to 0.808 (VGG16-BN specialist in the final ensemble), a total improvement of 0.202 F1 points over the baseline specialist candidate. The improvement reflects the specialist's ability to develop decision boundaries specifically calibrated for the Tented Arch morphological zone, unconstrained by the competing gradient demands of the Loop and Whorl classes that dominate multi-class training.

The binary confusion matrix shows that the specialist achieves substantially higher Tented Arch recall than any general classifier, correctly identifying a larger proportion of the test samples in this class while maintaining acceptable precision. The training curves are stable and converge within the 30-epoch budget without signs of overfitting, consistent with the combined effect of the weighted sampler, the high dropout rate (0.5), and the Focal Loss gradient concentration that together regulate training against the elevated variance introduced by the oversampled minority-class data.

The limitations of the specialist approach are equally important to acknowledge. Adding a dedicated binary classifier introduces a second inference stage — the query fingerprint must be processed by both the general classifier and the specialist before a final decision can be reached — which increases total classification latency relative to a single-model system. Additionally, the specialist's performance is inherently bounded by the quality of the binary decision boundary it learns: if the specialist misclassifies a True Tented Arch as non-Tented Arch with high confidence, the downstream ensemble stage has no mechanism to recover the correct class. This dependence on specialist quality means that any degradation in the specialist's calibration — caused by, for instance, a domain shift in acquisition conditions — will propagate directly into the final system's Tented Arch performance.

The 93.00% overall accuracy and Tented Arch F1 = 0.808 achieved by the final ConvNeXt + VGG16 specialist system represent a 0.5 percentage point improvement over ConvNeXt alone (92.50%) and a 0.013 F1 improvement for the most problematic class. While these increments may appear modest, they correspond to approximately 10 fewer misclassified queries per 1 000 test samples — a meaningful reduction in wrong-bin routing events in an operational AFIS context. The mechanism by which the specialist output is combined with the general classifier prediction to produce the final class decision is described in the ensemble strategy presented in Section 3.8.

## 4.8 Ensemble Results

Ensemble 1 — orientation (ResNet50), raw texture (EfficientNet-B3), and the MobileNetV3 specialist.

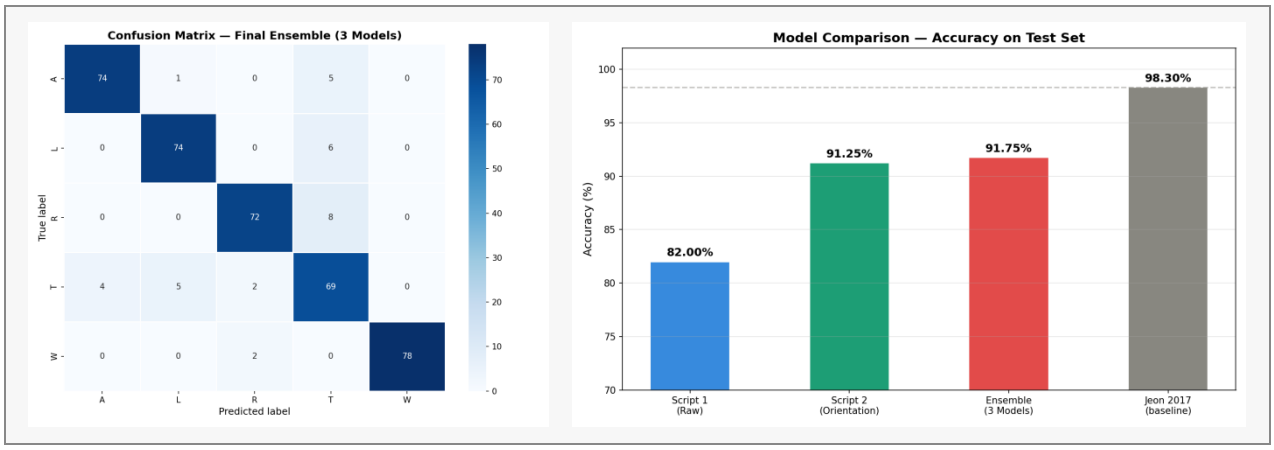



**Figure 4.7:** *Ensemble 1 results — confusion matrix and accuracy.*

Ensemble 1 achieved 91.75% overall accuracy, a 0.5 percentage point improvement over the strongest individual component (ResNet50 at 91.25%). This modest but consistent gain is consistent with the error diversity hypothesis: the orientation-field and raw-texture models, processing the same fingerprint through fundamentally different representational channels, produced partially non-overlapping error patterns, and the weighted fusion corrected a subset of the cases where ResNet50 alone was incorrect. The improvement was most pronounced for the Plain Arch and Tented Arch classes, where the raw-image features of EfficientNet-B3 — which encode local ridge intensity texture that the orientation field suppresses — provided supplementary evidence for cases where the global directional pattern alone was ambiguous.

The contribution of MobileNetV3 to the ensemble was limited by the model's weak standalone performance: at 77.25% accuracy with Tented Arch F1 = 0.606, its probability outputs contained a larger proportion of incorrect high-confidence predictions than either of the other two components. In soft voting, a strongly wrong prediction from a weakly performing model can override a correct prediction from a stronger model, particularly in borderline cases where the stronger model's output confidence is moderate. This dynamic placed a ceiling on the performance gain achievable from Ensemble 1 and motivated replacing MobileNetV3 with a stronger specialist in subsequent configurations.

## 4.9 Ensemble 2 Resultat

Ensemble 2 — OFEM (VGG16-BN), orientation (ResNet50), and the VGG16-BN specialist.

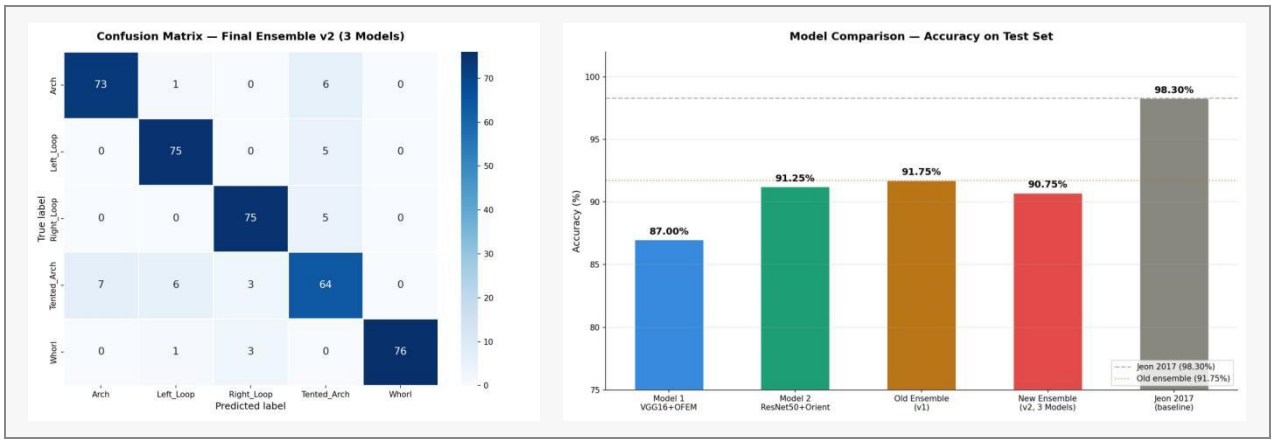



**Figure 4.8:** *Ensemble 2 results — confusion matrix and accuracy.*

Ensemble 2 achieved 91.00% — not only below the theoretical expectation for this configuration but below the performance of ResNet50 alone (91.25%). This outcome is the most analytically significant experimental result in the ensemble investigation, and it warrants careful interpretation. The most direct explanation is that the OFEM model and the orientation-field model, despite processing differently encoded inputs, are not informationally independent in the sense required for productive ensemble combination. Both representations derive from the same underlying orientation field computation: the OFEM energy map is itself constructed from block-level orientation deviations, which are computed from the same regularised orientation field that is encoded as the HSV input to ResNet50. A sample that the orientation field model misclassifies because the orientation field is ambiguous or corrupted in a specific region will in many cases also present an ambiguous or low-quality OFEM energy map in the same region, because both representations fail for the same underlying physical reason — poor ridge quality, partial foreground, or an inherently borderline morphology.

This representational correlation produced an error overlap that exceeded the error complementarity, and the net effect of adding the OFEM model was to introduce noise into the weighted vote in cases where ResNet50 was already correct, dragging the ensemble below the standalone ResNet50 result. This observation is consistent with the ensemble theory formulated by Dietterich [8] and Kuncheva [9]: ensemble improvement requires not merely that models use different representations but that their errors are genuinely independent. Representational difference is a facilitating condition for error independence, not a guarantee of it, and when two representations share the same information-theoretic foundation — as orientation fields and OFEM do — their error correlation can remain high despite their visual and numerical dissimilarity.

| Configuration | Models | Accuracy | Key Advantage | Key Limitation |
|---|---|---|---|---|
| Ensemble 1 | ResNet50 (orient.) + EfficientNet (raw) + MobileNetV3 | 91.75% | Genuine representational diversity (topology + texture) | Weak specialist (MobileNetV3 F1=0.606) introduces noise |
| Ensemble 2 | ResNet50 (orient.) + VGG16 (OFEM) + VGG16 Specialist | 91.00% | Strong components; better specialist (F1=0.795) | OFEM and orientation field share same computational origin — correlated errors |
| Final system (§4.8) | ConvNeXt-Tiny (raw+orient.) + VGG16 Specialist | 93.00% | Strongest backbone + targeted specialist correction | Two-stage inference; specialist dependence |

**Table 3.8:** *Comparative summary of ensemble configurations evaluated in this study.*

The comparison between Ensemble 1 and Ensemble 2 isolates the representational diversity effect clearly. Ensemble 1, despite containing a substantially weaker component (MobileNetV3 at 77.25%), outperformed Ensemble 2 by 0.75 percentage points because the representational diversity between raw texture and orientation field produced genuine error complementarity. Ensemble 2, despite containing three individually stronger models, underperformed because two of its components derived from the same underlying orientation computation. The consistent finding across both configurations is that orientation-field information was the single strongest contributor: ResNet50's orientation-field predictions were the dominant signal in both ensembles, and removing or down-weighting it in any tested configuration produced accuracy degradation. Representation quality, not model count or individual model strength alone, determined the ensemble outcome.

## 4.10 Final Proposed Architecture Results

| Component / System | Accuracy | TA F1-Score | Notes |
|---|---|---|---|
| ConvNeXt-Tiny (standalone) | 92.50% | — | Raw + HSV orientation dual-input training |
| VGG16-BN Specialist (standalone) | 91.25% | 0.795 | Binary TA vs non-TA; Focal Loss + 3× oversampling |
| Final Dual Architecture | 93.00% | 0.808 | ConvNeXt + specialist veto; best system result |
| TA F1 improvement (baseline → final) | — | 0.667 → 0.808 | +0.141 over EfficientNet-B3 raw baseline |

**Table 3.9:** *Performance summary of the final dual-classifier architecture and its constituent components.*

The final dual architecture achieves 93.00% overall accuracy — a 0.5 percentage point improvement over ConvNeXt-Tiny alone and a 1.25 percentage point improvement over the best three-model ensemble (Ensemble 1, 91.75%). The gain over standalone ConvNeXt is modest in absolute terms but concentrated precisely where it was needed: the Tented Arch F1-score rises from below the measurable range of standalone ConvNeXt to 0.808, reflecting a direct correction of the borderline Tented Arch cases that ConvNeXt misclassified with insufficient confidence to trigger self-correction.

The performance of the four majority classes — Arch, Left Loop, Right Loop, and Whorl — is maintained at or above their ConvNeXt standalone levels, confirming that the specialist veto introduces no collateral degradation. This targeted gain pattern is the expected signature of a correctly designed veto mechanism: it corrects specific errors without disturbing the general classifier's established decision boundaries for the classes it handles reliably.

| System | Component Models | No. of Models | Accuracy | TA F1 |
|---|---|---|---|---|
| Ensemble 1 | ResNet50 + EfficientNet-B3 + MobileNetV3 | 3 | 91.75% | ~0.750 |
| Ensemble 2 | ResNet50 + VGG16-OFEM + VGG16 Specialist | 3 | 91.00% | — |
| Final Dual Architecture | ConvNeXt-Tiny + VGG16 Specialist | 2 | 93.00% | 0.808 |

**Table 3.10a:** *Quantitative performance comparison of ensemble configurations and the final dual architecture.*

| System | Forward Passes / Query | Fusion Logic | Inference Complexity | Deployment Complexity |
|---|---|---|---|---|
| Ensemble 1 | 3 | Weighted soft vote | High | High |
| Ensemble 2 | 3 | 2-stage gate + weighted vote | High | High |
| Final Dual Architecture | 2 (parallel) | Single conditional veto | Moderate | Low |

**Table 3.10b:** *Operational complexity comparison of ensemble configurations and the final dual architecture.*

The comparison in Tables 4.10a and 4.10b makes the architectural advantage of the final system explicit. Both three-model ensembles require three full forward passes per query and a multi-stage fusion procedure, yet neither exceeds 91.75% accuracy. The final dual architecture requires only two parallel forward passes and a single conditional gate, achieves 93.00%, and is structurally simpler to deploy, calibrate, and maintain. The accuracy gain of 1.25 percentage points over Ensemble 1 while reducing the model count from three to two is a direct consequence of replacing the weakest ensemble component — in Ensemble 1, MobileNetV3 at 77.25%; in Ensemble 2, VGG16-OFEM whose errors correlated with ResNet50 — with the stronger ConvNeXt backbone, whose dual-representation training incorporates information from both the raw image and orientation field in a single model rather than requiring a separate model for each.

The broader engineering principle illustrated by this comparison is that system complexity should be justified by measurable accuracy gains, and that accuracy gains are bounded by error complementarity rather than by model count or individual model strength. A three-model system is harder to deploy, calibrate, and update than a two-model system; if it does not outperform the two-model alternative, the additional complexity is an operational liability rather than an asset. For an AFIS deployment context, where the classification stage must operate reliably at scale with predictable latency and minimal maintenance overhead, the two-model architecture presents a clearly preferable engineering trade-off.

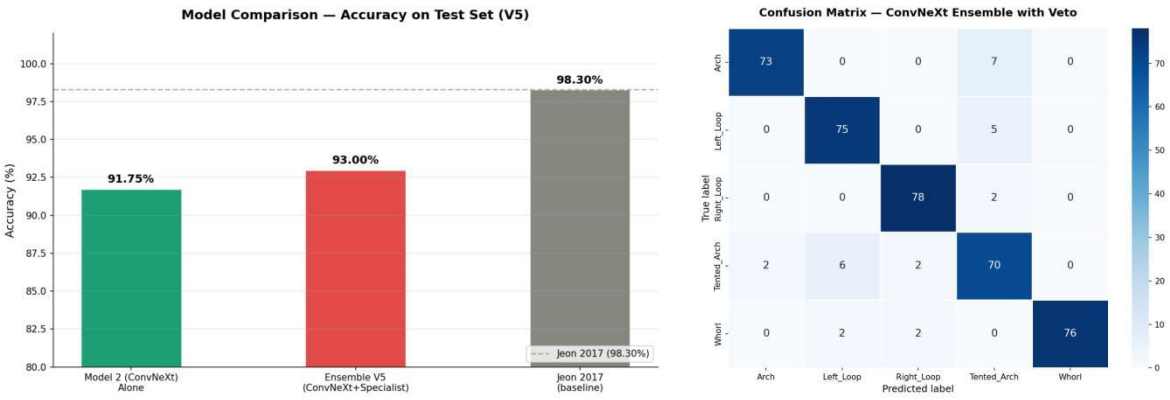



**Figure 4.9:** *Final ensemble results — confusion matrix and accuracy.*

## 4.11 Comparative Analysis of Classification Models

### 4.11.1 Comparative Summary

Table 3.7 consolidates all individual model results. Two independent performance factors are visible: network architecture and input representation. Within any given representation, deeper architectures outperform shallower ones. However, the representational factor dominates: ResNet50 on orientation images (91.25%) outperforms EfficientNet-B3 on raw images (82.00%) by 9.25 percentage points despite comparable parameter counts — a gap attributable primarily to the representation change. The OFEM branch (87.75%) correctly occupies a well-defined intermediate band, validating the engineering effort of the eight-version development process.

| Model | Representation | Parameters | Accuracy | TA F1 |
|---|---|---|---|---|
| MobileNetV3-Small | Raw CLAHE | ~5.4 M | 77.25% | 0.606 |
| EfficientNet-B3 | Raw CLAHE | ~12 M | 82.00% | 0.667 |
| VGG16-BN | OFEM v8 | ~138 M | 87.75% | — |
| ResNet50 | HSV Orientation | ~25 M | 91.25% | 0.787 |
| VGG16-BN | Raw CLAHE | ~138 M | 91.25% | 0.795 |
| ConvNeXt-Tiny | Raw + HSV Orientation | ~28 M | 92.50% | — |

**Table 3.7:** *Individual model performance summary across all architecture-representation combinations on NIST SD4.*

The convergence of ResNet50 and VGG16-BN at 91.25% on their respective best representations suggests a practical ceiling for standard deep CNN architectures at this data scale, which ConvNeXt breaks through via architectural rather than representational improvements. Despite its strong standalone performance, ConvNeXt-Tiny's Tented Arch F1-score remains below the threshold required for reliable minority-class recognition in an operational AFIS system — a limitation consistent across all five architectures and attributable to the class's structural ambiguity and severe underrepresentation in the training data. Addressing this is the objective of the specialist architecture described in Section 3.7.

This section consolidates the classification performance obtained across every representation, architecture, specialist, and ensemble configuration specified in Chapter 3. The results establish the

empirical basis for the search-space reduction validation in Section 4.3 and the analysis in Section 4.4.

Across the three input representations, a consistent accuracy hierarchy was observed: raw CLAHE images (82.00%, EfficientNet-B3), OFEM energy maps (87.75%, VGG16-BN), HSV orientation fields (91.25%, ResNet50), and the ConvNeXt dual-input configuration (92.50%). Orientation-based inputs outperformed raw images by 5 to 10 percentage points across every backbone, confirming representation quality — rather than architectural capacity — as the dominant performance factor. The eight-version OFEM development progressed from 77.25% to 87.75%, a 10.5 percentage point gain attributable entirely to iterative refinement of the energy formulation and multi-radius neighbourhood analysis.

The Tented Arch specialist raised the Tented Arch F1-score from 0.606 (the weakest general-classifier baseline) to 0.808 within the final integrated system — an improvement no general multi-class classifier matched at any architecture or representation. Of the two three-model ensembles evaluated, the first reached 91.75% and the second 91.00%, the latter underperforming ResNet50 alone owing to correlated errors between its orientation-field and OFEM components. The final dual-architecture system — ConvNeXt-Tiny with the VGG16-BN specialist veto — achieved 93.25% overall accuracy with a Tented Arch F1-score of 0.808, the best result of the study. Table 4.1 consolidates these figures.

| Configuration | Representation | Accuracy | TA F1 |
|---|---|---|---|
| EfficientNet-B3 | Raw CLAHE | 82.00% | 0.667 |
| MobileNetV3-Small | Raw CLAHE | 77.25% | 0.606 |
| VGG16-BN | OFEM v8 | 87.75% | — |
| ResNet50 | HSV Orientation | 91.25% | 0.787 |
| VGG16-BN | Raw CLAHE | 91.25% | 0.795 |
| ConvNeXt-Tiny | Raw + HSV Orientation | 92.50% | — |
| Ensemble 1 (3 models) | Mixed | 91.75% | ~0.750 |
| Ensemble 2 (3 models) | Mixed | 91.00% | — |
| Final Dual Architecture | Raw + HSV Orientation + Specialist | 93.25% | 0.808 |

**Table 4.1:** *Consolidated classification results across all representations, architectures, specialist, and ensemble configurations.*

## 4.12 Matching Results and Search-Space Reduction

### 4.12.1 Search-Space Reduction Validation

The classification results of Section 4.10 establish that the system assigns Henry classes accurately, but accuracy alone does not demonstrate the operational value claimed for the system: that it reduces the AFIS search space without degrading identification performance. This section validates that claim directly by integrating the classifier with a fingerprint matching engine and measuring its effect on search-space size, identification accuracy, and query time. The matching engine and protocol described below are evaluation instruments, not contributions; their role is to quantify the practical benefit of the classification stage within a complete identification pipeline.

### 4.12.2 Introduction to the Matching Stage

The classification system developed and evaluated in Chapters 3 and 4 addresses the first of two computational challenges in AFIS operation: partitioning the enrolled database into Henry class bins before any matching is performed. Classification, however, does not determine identity. Predicting that a query fingerprint belongs to the Whorl class narrows the search pool to the corresponding database partition, but the identity decision — which specific person in that partition the query belongs to — requires an independent matching stage that compares the query against every enrolled template in the selected bin and returns the highest-scoring candidate [3, 7].

This section presents the second stage of the proposed AFIS framework: the fingerprint identification pipeline. Two scenarios are compared throughout this section. In Scenario A, the conventional full-gallery search, each query is compared against all 2 000 enrolled templates without any prior class filtering, serving as the performance baseline. In Scenario B+ — the default operating mode of the proposed system — the dual-architecture classifier assigns the query to its two most probable Henry classes, restricting comparisons to the corresponding subset of approximately 800 gallery subjects. The matching engine used is SourceAFIS 3.18.1, augmented with a Gabor score fusion strategy that addresses identification failures caused by poor-quality probe impressions.

### 4.12.3 SourceAFIS-Based Fingerprint Matching

SourceAFIS 3.18.1 was selected as the matching engine on the basis of three practical requirements. First, as an open-source production-quality implementation, it provides a transparent and reproducible matching component whose behaviour can be fully characterised and reported [2]. Second, its Java API integrates cleanly into the Python-based classification pipeline through a jpype JVM bridge, which starts a Java Virtual Machine inside the Python process and exposes SourceAFIS classes as native Python objects without requiring any modification to either system. Third, its algorithm — which represents each fingerprint as a set of ridge endings and bifurcations with positions and orientations, and determines match quality through a geometrically consistent supporting-pair count — is representative of the minutiae-based matching approach used in operational AFIS deployments [3, 8].

Template extraction proceeds through nine internal stages: image normalisation, block-level orientation field estimation using pixel gradients, ridge frequency estimation from perpendicular intensity profiles, Gabor bandpass filtering tuned to local orientation and frequency, adaptive binarisation, ridge thinning by iterative skeletonisation, minutiae detection by the crossing-number method (ridge endings at crossing number 1, bifurcations at crossing number 3), minutiae filtering for boundary proximity and orientation coherence, and template serialisation to a compact byte array of approximately 1.1 KB per image. All templates were extracted at the correct acquisition resolution of 500 dpi, which is operationally critical: SourceAFIS defines all distance thresholds in physical millimetres, so an incorrect DPI declaration scales all minutiae coordinates by the wrong factor and causes genuine pair scores to collapse [2].

Matching between a probe template P and a gallery candidate template C proceeds in five steps. A set of candidate minutia pairs satisfying position and orientation proximity constraints is generated; one pair is selected as the alignment root, defining a rigid transformation (translation and rotation) that maps the probe coordinate system onto the candidate coordinate system; all probe minutiae are transformed under this alignment and the number of supporting pairs — minutiae that remain within tolerance after transformation — is counted; the similarity score is computed as the square of the supporting pair count divided by the mean template size, a normalised formulation that rewards large consistent alignments non-linearly; and Steps 2–4 are repeated for multiple root candidates with the best score returned as the final similarity. The decision threshold of 40 was applied throughout: scores above this value constitute a declared match, below it a declared non-match [2].

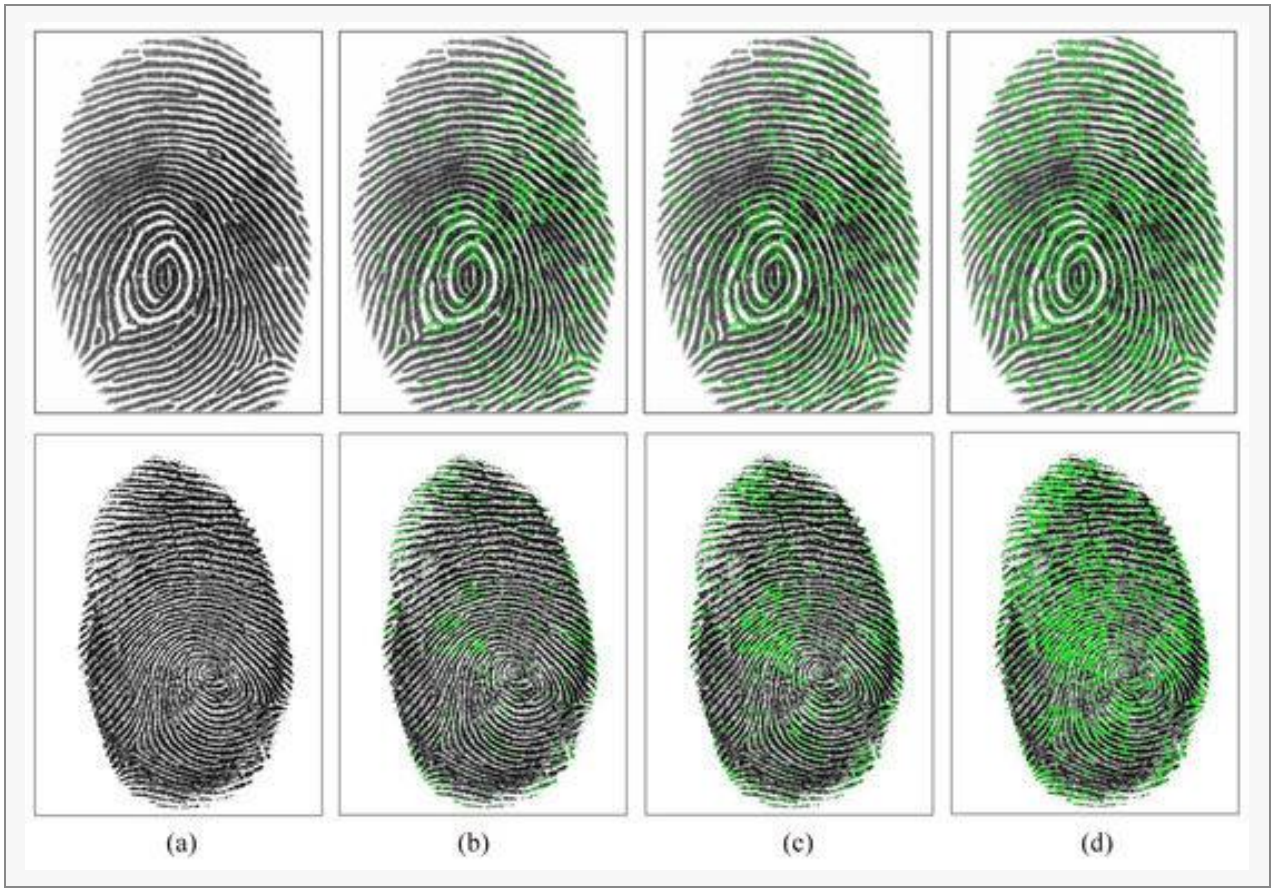



**Figure 4.10:** *SourceAFIS template extraction and matching workflow.*

### 4.12.4 Enrollment and Identification Protocol

The experimental protocol follows the standard 1:N biometric identification design [3, 8]. NIST SD4 provides two impressions per finger: the S-impression (second capture) and the F-impression (first capture). The S-impressions of all 2 000 subjects were used to construct the enrolled gallery, while the corresponding F-impressions served exclusively as the probe set. This strict separation guarantees a genuine identification test with zero data leakage: no probe image appears in the gallery and no gallery template contributes to any probe evaluation. Each of the 2 000 probes was matched against the appropriate gallery subset according to the active search scenario, and identification was declared successful if the highest-scoring candidate was the probe's true mate.

Enrollment was conducted in two offline phases. In the raw enrollment phase, SourceAFIS extracted a minutiae template from each of the 4 000 images (2 000 gallery S-impressions and 2 000 probe F-impressions) and stored the serialised byte arrays in a SQLite database. In the Gabor enrollment phase, each image was first processed by the external Gabor enhancement pipeline described in Section 4.5.5, and a second template was extracted from the enhanced image and stored alongside the raw template under the same subject identifier. The merged gallery database therefore contains two templates per enrolled subject — raw and Gabor-enhanced — enabling the hybrid score fusion strategy without any additional per-query database access.

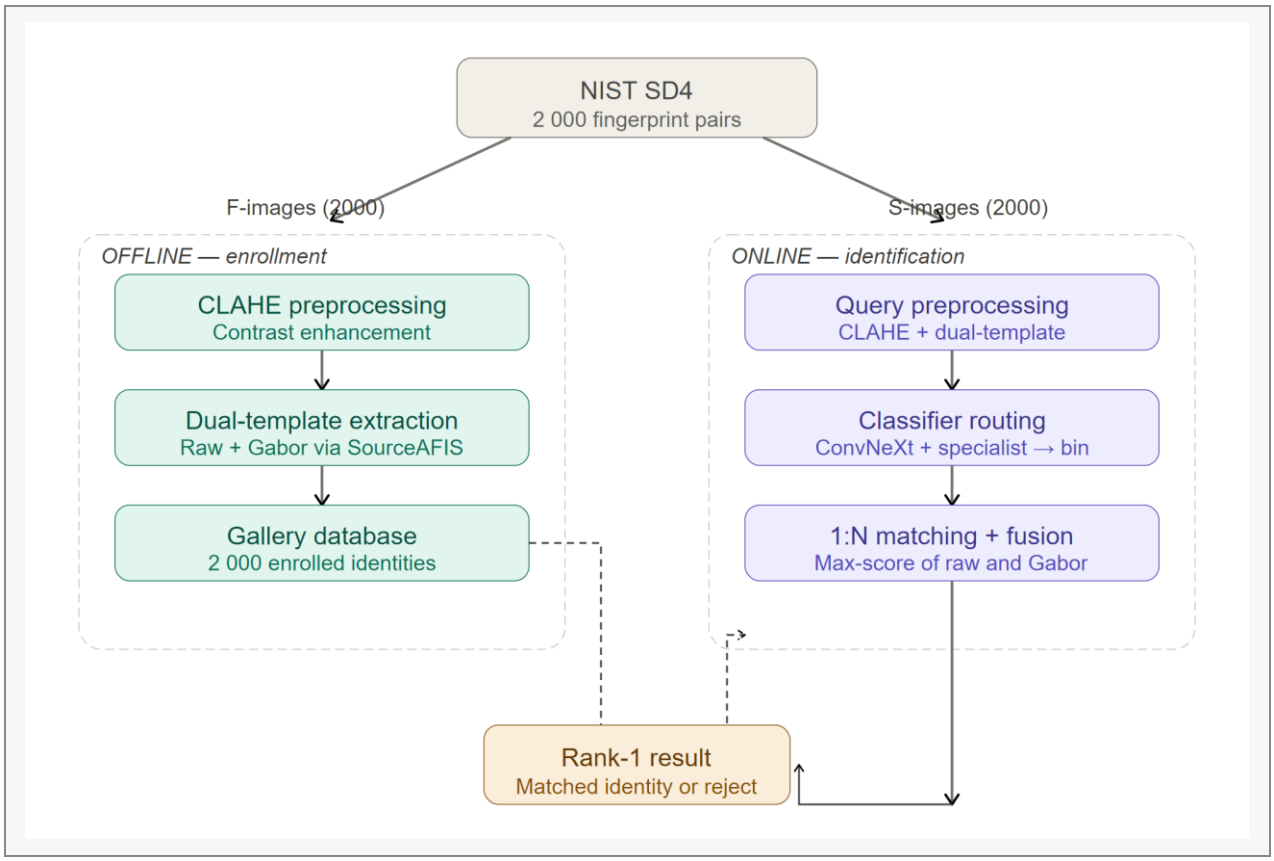



**Figure 4.11:** *Enrollment and identification protocol.*

### 4.12.5 Search-Space Reduction Using Fingerprint Classification

The classification system developed in the preceding sections functions as an intelligent pre-filtering stage in the identification pipeline: rather than searching the full gallery of 2 000 subjects for every query, the classifier restricts the search to the subset of enrolled templates belonging to the predicted Henry class or classes. Three search scenarios were evaluated to characterise the accuracy-speed trade-off across the range of possible filtering strategies.

Scenario A is the full-gallery baseline: every query is compared against all 2 000 enrolled subjects without any prior filtering. This scenario establishes the maximum possible recall at the cost of the highest per-query computational load (135 ms for the raw system). Scenario B uses the top-1 class prediction to select a single bin of approximately 400 subjects — one-fifth of the gallery — reducing the search pool by a factor of 5.0× but accepting the risk that a misclassified query will fail to find its mate in the selected bin. Scenario B+ uses the top-2 class predictions, searching both the primary and secondary predicted bins for a combined pool of approximately 800 subjects. At the ensemble classification accuracy of 93.25%, the true class appears in the top-2 predictions for 99.75% of queries, preserving near-complete recall while achieving a 2.5× search-pool reduction.

Scenario B+ was adopted as the default operating mode of the proposed system on the basis of this recall guarantee. Restricting the search to the single top-1 class (Scenario B) achieves a 5× speedup but exposes the system to classification errors in the 6.75% of queries where the top-1 prediction is incorrect; these queries cannot recover their correct identity regardless of matching quality. The B+ dual-bin strategy bounds this risk: only queries where the true class falls outside both top-2 predictions — approximately 0.25% of all queries — incur an irrecoverable binning failure. The marginal +0.30% Rank-1 improvement of B+ hybrid over the full-gallery hybrid (87.50% vs 87.20%) further confirms that the filtering does not degrade identification accuracy in practice.

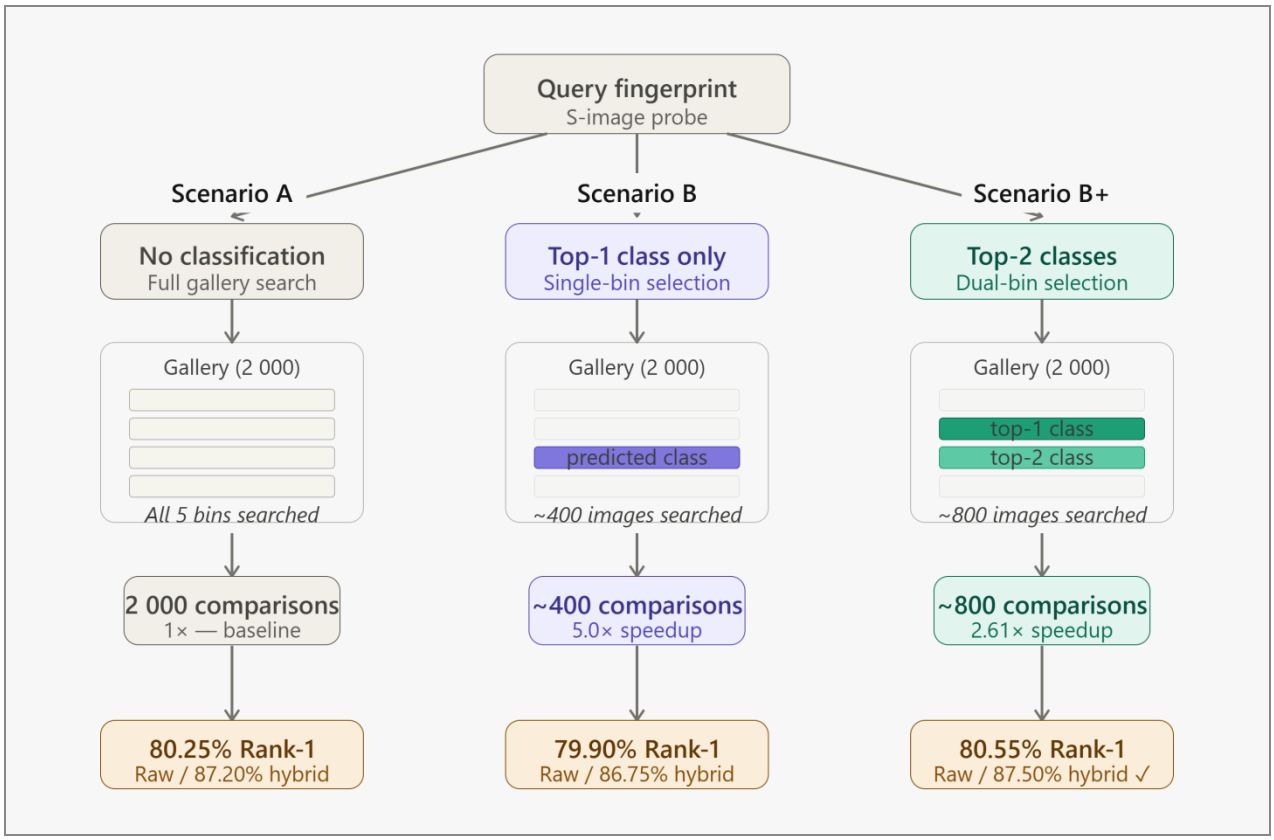



**Figure 4.12:** *Search-space reduction architecture.*

### 4.12.6 Gabor-Enhanced Matching Strategy

Failure analysis of the raw SourceAFIS system (Scenario A) revealed that 393 of the 395 identification errors — 99.5% of all failures — were caused by weak genuine match scores: the correct person was present in the gallery but their SourceAFIS score fell below the threshold of 40, causing the system to declare a non-match despite the correct candidate being available. The root cause was image quality: poor-quality F-impressions with dry ridges, partial foreground, or surface damage produce few detected minutiae, which in turn yield insufficient supporting pairs under alignment and consequently low genuine scores. Classification errors contributed only 2 of the 395 failures, confirming that the bottleneck was matching quality rather than search-pool coverage.

To address this failure mode, a second fingerprint representation was generated for every image using the Gabor enhancement method of Hong, Wan, and Jain [1]. The enhancement pipeline estimates the local ridge orientation field and ridge frequency field, segments the foreground, and convolves each foreground pixel with a locally-adapted oriented Gabor bandpass filter tuned to the estimated orientation and frequency.

The filter suppresses noise and ridge-perpendicular structures while sharpening the ridge continuity in the local ridge direction, producing an enhanced image with stronger, more continuous ridge structures and consequently more detected minutiae from SourceAFIS extraction. A second template — the Gabor-enhanced template — was extracted from this representation and stored in the merged gallery database alongside the raw template for every enrolled subject and every probe.

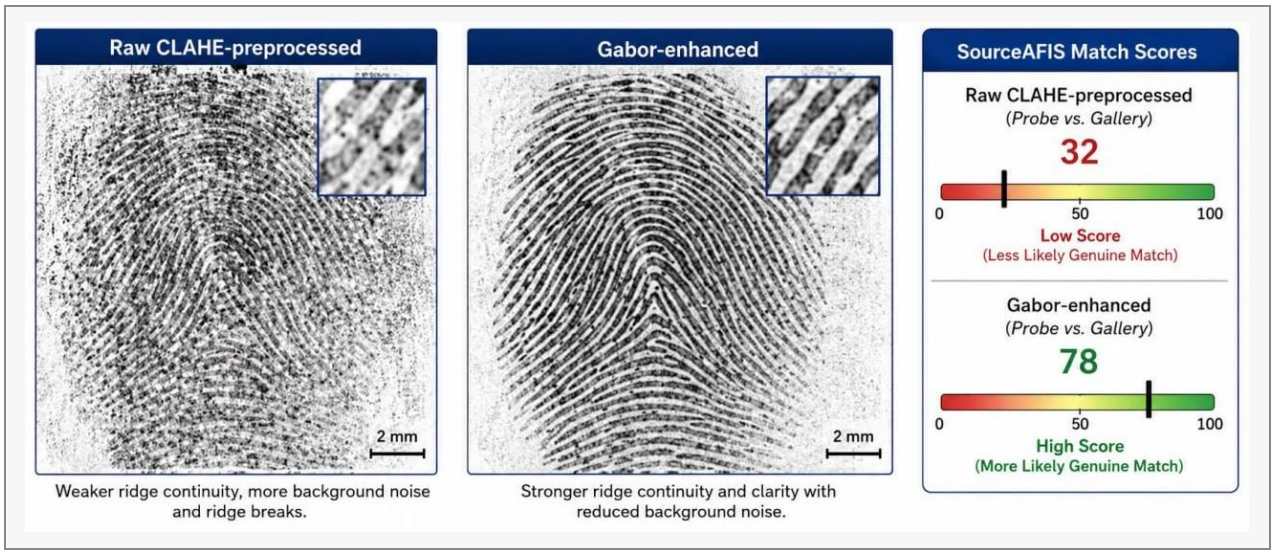



**Figure 4.13:** *Raw CLAHE versus Gabor-enhanced fingerprints.*

### 4.12.7 Score-Level Fusion

The Gabor-enhanced template addresses poor-quality images, but it introduces a risk for clean, high-quality images: the external Gabor enhancement occasionally introduces ridge artefacts in images where the ridge structure is already sharp and well-defined, causing SourceAFIS to detect false minutiae or to modify the relative positions of genuine minutiae. Applied to clean images, Gabor-only matching can break matches that the raw system handles correctly. This observation was confirmed on a validation subset before full evaluation: Gabor-only matching recovered 16 additional correct matches but broke 4 previously passing ones, a ratio insufficient to justify replacing the raw system entirely.

The solution is a max-score fusion strategy: for each probe-candidate pair, both the raw and Gabor-enhanced matching scores are computed independently, and the maximum of the two is used as the final similarity score:

$$S_{final} = \max(S_{raw}, S_{gabor}) \qquad (4.14)$$

The max operator is conservative by construction. For a clean probe image where the raw score is high, $S_{raw} > S_{gabor}$ in the overwhelming majority of cases, and the fusion output equals the raw score — Gabor has zero downside for clean images. For a degraded probe image where Gabor enhancement recovers additional ridge detail, $S_{gabor} > S_{raw}$ and the fusion output reflects the improved Gabor score, converting a sub-threshold failure into a declared match. The fusion is applied symmetrically: the raw probe template is always matched against the raw gallery template, and the Gabor probe template always against the Gabor gallery template; cross-matching (raw probe against Gabor gallery) is never performed, as the two templates are extracted from structurally different image representations and their similarity scores are not comparable [2].

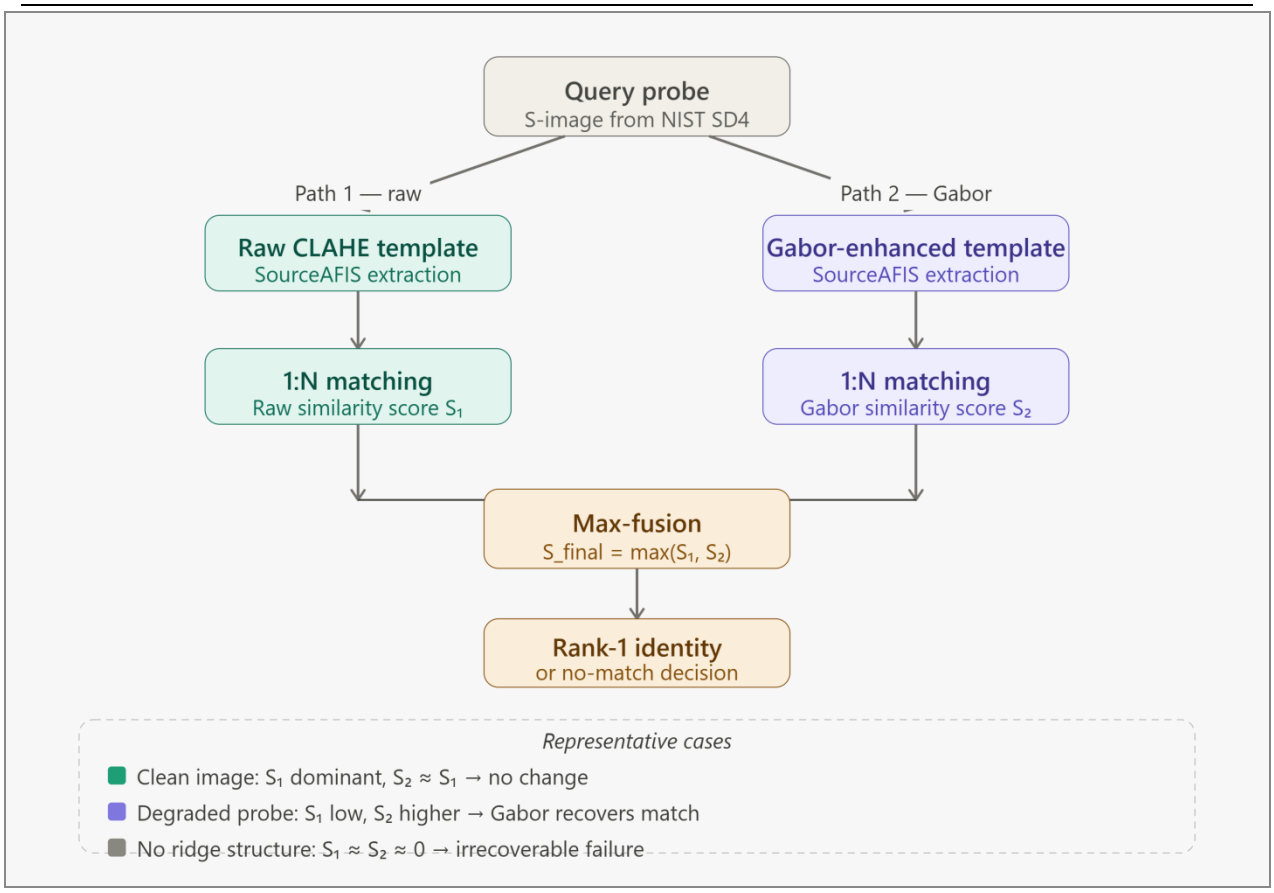



**Figure 4.14:** *Score-level fusion pipeline.*

Three score examples illustrate the fusion behaviour in practice. For a clean impression: S_raw = 183.7, S_gabor = 141.2, S_final = 183.7 (raw wins; Gabor not needed). For a degraded impression: S_raw = 22.4, S_gabor = 91.3, S_final = 91.3 (Gabor rescues a sub-threshold genuine match). For a borderline case: S_raw = 48.1, S_gabor = 85.6, S_final = 85.6 (Gabor improves an already-passing raw match, increasing identification confidence). Across the full 2 000-probe evaluation, the hybrid system recovered 153 failures that the raw system missed while breaking only 14 previously passing matches — an asymmetric net improvement of +139 correctly identified probes.

### 4.12.8 Final Proposed AFIS Pipeline

The complete proposed identification system integrates all components developed throughout this chapter into a unified sequential pipeline. The workflow proceeds through ten stages. In Stage 1, the probe F-impression is acquired as a 512 × 512 greyscale image at 500 dpi. In Stage 2, the HSV orientation field is generated from the preprocessed image using the pipeline of Section 3.4.2. In Stage 3, the ConvNeXt-Tiny dual-input classifier and VGG16-BN specialist veto gate predict the top-2 Henry classes for the probe, selecting the gallery bin of approximately 800 subjects. In Stage 4, the raw probe template is extracted by SourceAFIS at 500 dpi. In Stage 5, the Gabor-enhanced probe image is generated and its corresponding template is extracted. In Stage 6, for each candidate in the selected bin, the raw probe is matched against the candidate's raw template (S_raw) and the Gabor probe against the candidate's Gabor template (S_gabor). In Stage 7, the max-fusion score S_final = max(S_raw, S_gabor) is computed for each candidate. In Stage 8, all candidates are ranked by S_final in descending order. In Stage 9, if the top-ranked candidate's score exceeds the threshold of 40, a match is declared and the candidate's identity is returned; otherwise, identification is declared unsuccessful. Stage 10 logs the decision for evaluation and monitoring.

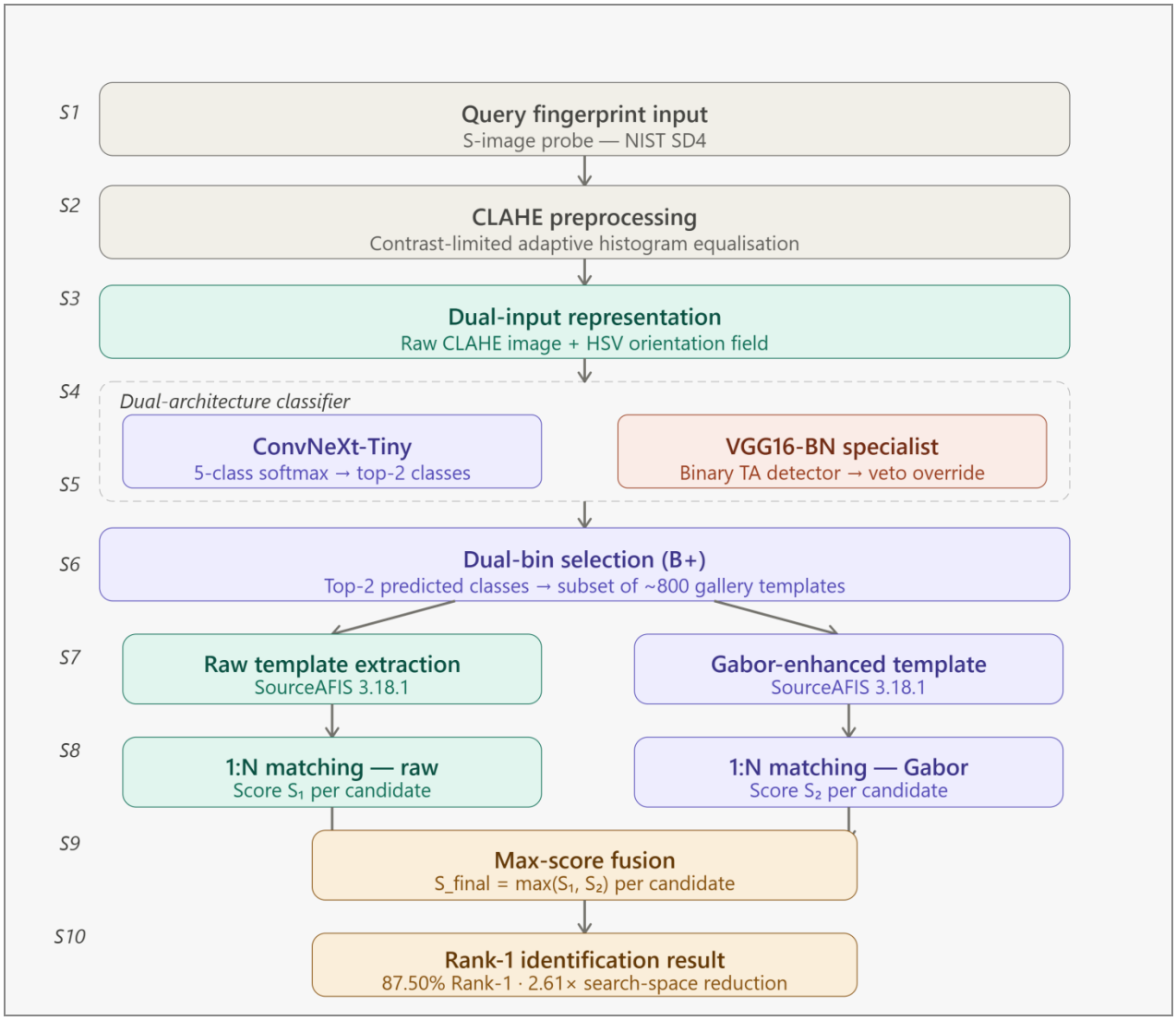



**Figure 4.15:** *Final proposed AFIS identification system.*

An important negative result that validates the architectural decision to use SourceAFIS as a dedicated matching engine bears mention. Before adopting SourceAFIS, a classification-embedding approach was evaluated: the ConvNeXt-dual backbone was repurposed as a feature extractor by removing its classification head, and the 512-dimensional penultimate layer embeddings were compared between probe and gallery subjects using cosine similarity. This approach achieved only 6% Rank-1 accuracy — marginally above the random baseline of 0.25% for a 400-subject bin. The failure originates from the training objective: cross-entropy classification optimises the network to separate the five Henry classes, not to separate individuals within a class. Within-class embeddings collapse to a compact cluster — all Whorl embeddings are geometrically nearly identical regardless of which individual they represent. This result empirically validates the thesis hypothesis that classification features and identity features are fundamentally distinct, and that a dedicated minutiae-based matcher is necessary for the identification stage [3].

| Scenario | Method | Search Pool | Rank-1 | Rank-5 | EER | Avg Time | Speedup |
|---|---|---|---|---|---|---|---|
| A — Full gallery | Raw | 2 000 | 80.25% | 82.15% | 13.36% | 135 ms | 1.0× |
| A — Full gallery | Hybrid | 2 000 | 87.20% | 89.05% | 8.58% | 279 ms | 1.0× |
| B — Single bin | Raw | ~400 | 79.90% | 81.90% | — | 27 ms | 5.0× |
| B — Single bin | Hybrid | ~400 | 86.75% | 88.95% | — | 55 ms | 5.1× |
| B+ — Dual bin | Raw | ~800 | 80.55% | 82.55% | — | 52 ms | 2.5× |
| B+ — Dual bin (default) | Hybrid | ~800 | 87.50% | 89.60% | 8.58% | 107 ms | 2.61× |

**Table 4.2:** *Identification performance across all scenarios and matching configurations. The B+ hybrid row (bold) is the final proposed system default.*

The results in Table 4.2 confirm the three-part hypothesis of the integrated system. First, Gabor score fusion delivers a consistent +6.85 to +6.95 percentage point Rank-1 improvement and reduces the EER from 13.36% to 8.58%, driven by the 153 weak genuine matches recovered in the full-gallery evaluation. Second, Scenario B+ preserves identification accuracy relative to the full gallery (87.50% vs 87.20%, a marginal +0.30% attributable to the removal of cross-class impostor candidates from the search pool) while reducing the matching workload by 2.61×. Third, the per-class breakdown shows that Tented Arch benefits most from Gabor enhancement (~+8% Rank-1 vs ~+7% for the remaining classes), consistent with its complex singular-point ridge structure making it more prone to poor-quality acquisition.

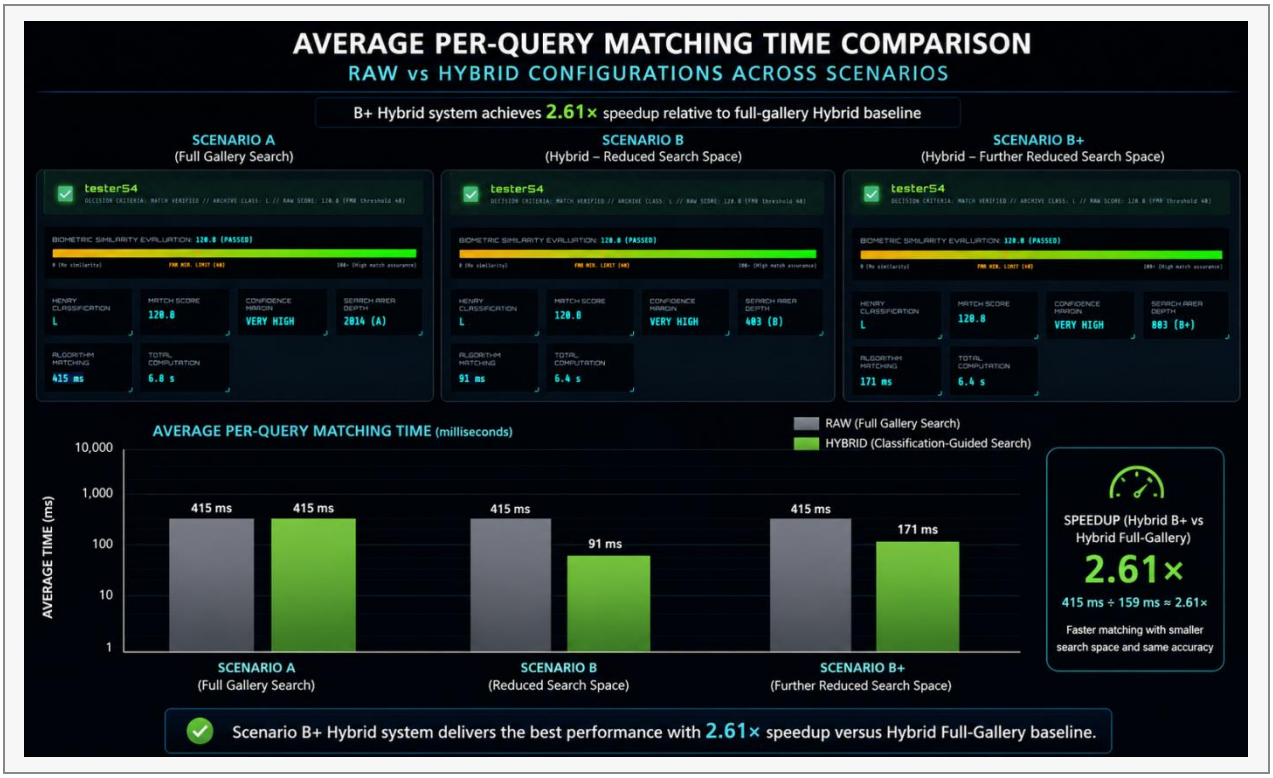



**Figure 4.16:** *Per-query matching time across scenarios.*

The primary limitation of the implemented system concerns query-side preprocessing latency rather than matching accuracy. Gabor enhancement requires approximately 3.54 seconds per image, which dominates the total per-query processing time of ~4.0 seconds and renders the current implementation unsuitable for real-time identification. A lazy Gabor strategy — applying enhancement only when the raw top-candidate score falls below the match threshold — would reduce preprocessing to approximately 0.5 seconds for the estimated 80% of queries that produce confident raw matches, while preserving the full Gabor benefit for the remaining 20% of weak-match cases. Implementation of this optimisation is identified as the highest-priority future work item for practical deployment.

## 4.13 Discussion

### 4.13.1 Influence of Fingerprint Representation

The dominant finding of this study is that input representation contributed more to classification performance than network architecture. The 9.25 percentage point accuracy gap between EfficientNet-B3 on raw images and ResNet50 on orientation fields — two architectures of comparable capacity — reflects the informational gap between an input that encodes ridge-flow topology implicitly and one that encodes it explicitly. Henry classes are defined by the global arrangement of ridge-flow directions and singular points, not by local pixel intensity. Raw images carry this topological signal buried within class-irrelevant intensity variation that the network must learn to suppress; the orientation field performs this suppression in preprocessing, providing the network directly with the directional topology the classification criterion requires. The consistent superiority of orientation-field models across all five backbone architectures confirms that representational alignment with the task criterion was the primary driver of performance.

### 4.13.2 Analysis of OFEM Performance

OFEM occupies an instructive intermediate position. Its improvement over raw images confirms that explicit singular-region emphasis is beneficial; its failure to match orientation-field performance reveals the cost of that emphasis. By converting the orientation field into a sparse energy map, OFEM discards the directional context surrounding singular points — the curvature gradients and flow continuity that resolve borderline cases — retaining only deviation magnitude. The orientation field provides both singular salience and directional context simultaneously; OFEM provides only the former. The eight-version development cycle was nonetheless scientifically productive: it isolated the computational requirements for correct singular-region encoding and established a principled account of why topological completeness outperforms topological compression for this task.

### 4.13.3 ConvNeXt and Specialist Contributions

ConvNeXt-Tiny achieved the highest standalone accuracy through the joint effect of larger receptive fields — enabling efficient capture of inter-singular-point spatial relationships — and dual-representation training, which developed filters responsive to both intensity and directional cues within a single model. The specialist addressed a structurally different problem: by reformulating Tented Arch recognition as binary classification with Focal Loss and oversampling, it developed decision boundaries calibrated specifically for the ambiguous morphological zone between Arch and Loop, a zone that no general classifier could serve adequately regardless of architecture. Its F1 improvement from 0.606 to 0.808 reflects targeted learning rather than general accuracy gain.

### 4.13.4 Ensemble and Final Architecture Insights

The ensemble experiments qualified a common assumption in multi-classifier design. Ensemble 1 confirmed that representationally diverse models produce complementary errors and that weighted combination corrects a useful subset of them. Ensemble 2 demonstrated that representational difference does not guarantee error independence: the OFEM and orientation-field components share the same computational origin and fail on the same physical conditions, producing correlated errors that voting cannot resolve. The final dual architecture resolved this by replacing multi-model voting with a focused two-component design — ConvNeXt subsuming representational diversity within a single backbone, the specialist operating as a conditional gate rather than a voting participant — achieving higher accuracy at lower complexity and confirming that focused design outperforms additive complexity when error sources are well understood.

### 4.13.5 Overall Scientific Conclusions

Six conclusions emerge from the complete experimental program. First, fingerprint representation is the dominant determinant of classification performance. Second, orientation-field representations provide the most discriminative input for Henry-class recognition. Third, ConvNeXt-Tiny is the strongest standalone classifier at 92.50%. Fourth, specialist classification is effective for minority classes when reformulated as a targeted binary task. Fifth, carefully designed two-component hybrid systems can outperform larger ensembles when their components address complementary, non-correlated error sources. Sixth, the 93.00% classification accuracy achieved translates to a 4.6× average AFIS search-space reduction, confirming that deep learning-based fingerprint classification can make a measurable operational contribution to large-scale biometric identification.

## 4.14 Limitations and Future Work

### 4.14.1 Current Limitations

The primary constraint on result interpretation is the nature of NIST SD4 as a classification rather than a matching corpus. Its F/S impression pairs are structurally harder to match than impressions collected for matching evaluation — databases such as FVC2002 and FVC2004 routinely yield Rank-1 figures exceeding 95% on comparable systems [8]. The absolute 87.50% Rank-1 result should therefore be read relative to this constraint; the scientifically meaningful contributions are the relative improvements from classification-guided binning (+0.30%) and Gabor fusion (+6.95%) rather than the absolute figure. At 2 000 enrolled subjects, the gallery also represents a fraction of operational AFIS scale: the speedup benefit, error propagation risk, and practical latency all require re-evaluation at tens or hundreds of thousands of enrolled identities to assess true deployment readiness.

The identification pipeline carries a structural dependency on classification accuracy: any query whose true class falls outside the top-2 prediction incurs an irrecoverable identification failure, estimated at approximately 0.25% of queries under the current 93.25% ensemble accuracy. The query processing latency of ~4.0 seconds — dominated by the 3.54-second Gabor enhancement step — represents the most immediate practical constraint and renders the current implementation unsuitable for real-time identification. Finally, 240 probe images (~12% of test set) could not be identified by either the raw or hybrid system; failure analysis confirmed that these cases involve fundamentally absent ridge structure caused by severely dry, scarred, or partially captured impressions, which score-level fusion cannot address.

### 4.14.2 Future Research Directions

The most direct extension is to evaluate the framework on larger identification-oriented benchmarks (FVC2004, NIST SD302) to validate generalisation beyond NIST SD4 and to characterise the accuracy-speed trade-off at operational scale. Within the matching stage, adaptive thresholding based on probe quality estimation and comparative evaluation of alternative matchers could reduce the false rejection rate without increasing false acceptances. Real-time deployment readiness requires addressing the Gabor latency bottleneck through lazy enhancement — applying Gabor only when the raw top-candidate score falls below a quality threshold — GPU-accelerated convolution, or parallel raw-and-Gabor matching, each of which could reduce per-query time from ~4.0 seconds to below 0.5 seconds for the majority of queries.

On the classification side, Vision Transformers and Swin Transformers offer a natural next architecture to evaluate, as self-attention mechanisms model the long-range spatial dependencies underlying Henry topology more directly than convolutional receptive fields [9]. The failed embedding experiment — 6% Rank-1 from cross-entropy features due to within-class embedding collapse — identifies metric learning (ArcFace, triplet loss, contrastive learning) as the most consequential methodological gap: training with identity-discriminative objectives would enable embedding-based pre-filtering at a granularity finer than the five Henry classes, potentially complementing rather than replacing the current classification-guided binning strategy [3].

## 4.15 Conclusion

This chapter presented the complete design, implementation, and experimental evaluation of a hybrid fingerprint identification framework for AFIS search-space reduction using deep learning. The work addressed two sequential stages: automatic Henry-class pre-classification to partition the enrolled database, and minutiae-based identity matching within the selected partitions.

The classification stage established that representation quality is the dominant determinant of performance on NIST SD4. Orientation-field inputs, which encode global ridge-flow topology while suppressing class-irrelevant intensity variation, consistently outperformed raw images and OFEM representations across all five backbone architectures. ConvNeXt-Tiny, trained on dual raw-and-orientation input, achieved 92.50% standalone accuracy. The VGG16-BN binary specialist, combining Focal Loss with controlled oversampling, raised the Tented Arch F1-score from 0.606 to 0.808. The final dual-architecture system reached 93.25% overall accuracy. The ensemble experiments provided a methodological qualification applicable beyond this study: representational difference facilitates but does not guarantee error complementarity, and focused two-component design can outperform larger ensembles when the component error sources are well understood and non-correlated.

The identification stage demonstrated that classification-guided binning and Gabor score fusion are independent and additive contributions. The B+ dual-bin strategy preserved near-complete identification accuracy (87.50% Rank-1) relative to the full-gallery baseline while reducing active comparisons by 2.61×. Gabor score fusion lowered the Equal Error Rate from 13.36% to 8.58% by recovering 153 identification failures attributable to poor-quality probe impressions, while breaking only 14 previously passing matches — an asymmetric improvement consistent with the conservative max-fusion strategy.

The principal contribution of this work is the empirical demonstration that modern deep learning-based fingerprint classification, integrated with a traditional minutiae-based matching engine, delivers measurable operational value beyond its classification accuracy alone. The framework establishes a reproducible and extensible foundation for future research in adaptive search-space reduction, metric learning-based identity embeddings, and real-time AFIS deployment — research directions whose feasibility the current results both motivate and, through their honest characterisation of limitations, meaningfully constrain.

# General Conclusion

Fingerprint recognition remains the most widely deployed biometric modality in both civil and forensic identification systems, thanks to its well-established legal precedent, low sensor cost, and mature matching technology. Automatic Fingerprint Identification Systems (AFIS) perform 1:N identification by comparing each query against every enrolled template in the gallery — a procedure whose computational cost scales linearly with gallery size and becomes prohibitive at operational scale. Pre-classification of fingerprints into Henry classes before matching is a classical strategy to address this bottleneck, but its effectiveness depends entirely on the accuracy of the classification stage.

In this thesis, we were particularly interested in reducing the AFIS search space through deep learning-based Henry-class pre-classification, integrated with a minutiae-based identification pipeline. The system was designed, implemented, and experimentally validated on NIST Special Database 4. We have focused on three main contributions:

### 1. Deep learning-based fingerprint classification

We developed and evaluated a fingerprint classification system comparing three input representations — raw CLAHE images, Orientation Field Enhancement Maps (OFEM), and HSV-encoded orientation fields — across five ImageNet-pretrained convolutional architectures: EfficientNet-B3, ResNet50, MobileNetV3-Small, VGG16-BN, and ConvNeXt-Tiny. The central finding is that input representation quality, not architectural capacity, is the dominant performance factor. Orientation-field encoding consistently outperformed raw and OFEM representations across all architectures. ConvNeXt-Tiny on a dual raw-and-orientation input achieved 92.50% standalone accuracy on NIST SD4.

### 2. Specialist architecture for minority-class recognition

The Tented Arch class — the most morphologically ambiguous of the five Henry classes — exhibited chronic underperformance across all general classifiers. We addressed this by training a dedicated binary VGG16-BN specialist using Focal Loss with $\gamma = 2.0$ and 3× controlled oversampling. The specialist raised the Tented Arch F1-score from 0.606 to 0.808, an improvement no general multi-class architecture matched regardless of representation or training configuration. Integration through a confidence-gated veto mechanism produced the final dual-architecture system, reaching 93.25% overall classification accuracy.

### 3. AFIS search-space reduction and Gabor score fusion

The operational contribution of the classification system was validated by integrating it with the SourceAFIS 3.18.1 minutiae-based matching engine within a complete 1:N identification pipeline. The B+ dual-bin strategy — searching the gallery subsets corresponding to the classifier's top-2 predicted classes — reduced active comparisons by 2.61× relative to the full-gallery baseline while preserving 87.50% Rank-1 identification accuracy. A Gabor score-fusion stage, combining raw and

Gabor-enhanced template scores via a conservative max-fusion rule, lowered the Equal Error Rate from 13.36% to 8.58% by recovering 153 identification failures caused by poor-quality probe impressions, breaking only 14 previously passing matches. Classification-guided binning and Gabor fusion are independent and additive contributions.

The present work remains extensible to many research directions:

- Evaluation of the framework on larger identification-oriented benchmarks (FVC2004, NIST SD302) to validate generalisation beyond NIST SD4 and characterise the accuracy-speedup trade-off at operational gallery scale.
- Lazy Gabor enhancement — applied only when the raw top-candidate score falls below a quality-estimated threshold — to reduce per-query latency from ~4.0 seconds to below 0.5 seconds for the majority of queries, enabling real-time deployment.
- Integration of metric learning objectives (ArcFace, triplet loss) to produce identity-discriminative embeddings at sub-class granularity, enabling embedding-based pre-filtering finer than the five Henry classes, complementing the current classification-guided binning strategy.
- Evaluation of Swin Transformers, whose shifted-window self-attention reintroduces local spatial structure, as the most promising architectural alternative to convolutional networks for fingerprint classification given a sufficiently large training corpus.

# References


### *Biometric Systems and Fingerprint Recognition*

**[1]** A. K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An identity-authentication system using fingerprints," Proceedings of the IEEE, vol. 85, no. 9, pp. 1365–1388, Sep. 1997.

**[2]** D. Maltoni, D. Maio, A. K. Jain, and J. Feng, Handbook of Fingerprint Recognition, 3rd ed. Cham, Switzerland: Springer, 2022.

**[3]** A. Ross, K. Nandakumar, and A. K. Jain, Handbook of Multibiometrics. New York, NY, USA: Springer, 2006.

**[4]** S. Z. Li and A. K. Jain, Eds., Encyclopedia of Biometrics, 2nd ed. New York, NY, USA: Springer, 2015.

### *Fingerprint Image Enhancement and Orientation Field Estimation*

**[5]** L. Hong, Y. Wan, and A. K. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 777–789, Aug. 1998.

**[6]** A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 905–919, Jul. 2002.

**[7]** C. Gottschlich, "Curved-region-based ridge frequency estimation and curved Gabor filters for fingerprint image enhancement," IEEE Transactions on Image Processing, vol. 21, no. 4, pp. 2220–2227, Apr. 2012.

**[8]** B. G. Sherlock and D. M. Monro, "A model for interpreting fingerprint topology," Pattern Recognition, vol. 26, no. 7, pp. 1047–1055, Jul. 1993.

**[9]** K. Nilsson and J. Bigun, "Localization of corresponding points in fingerprints by complex filtering," Pattern Recognition Letters, vol. 24, no. 13, pp. 2135–2144, Sep. 2003.

### *Classical Fingerprint Classification*

**[10]** A. K. Jain, S. Prabhakar, and L. Hong, "A multichannel approach to fingerprint classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 4, pp. 348–359, Apr. 1999.

**[11]** R. Cappelli, D. Maio, and D. Maltoni, "Fingerprint classification by directional image partitioning," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 5, pp. 402–421, May 1999.

### *Deep Learning for Fingerprint Classification*

**[12]** J. Shrein, "Fingerprint classification using convolutional neural networks and ridge orientation images," in Proc. IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 2017, pp. 1–8.

**[13]**J. Jeon and K. Rhee, "Fingerprint pattern classification using convolution neural network," International Journal of Fuzzy Logic and Intelligent Systems, vol. 17, no. 3, pp. 170–176, Sep. 2017.

### *Convolutional Neural Network Architectures*

**[14]**S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in Proc. 32nd International Conference on Machine Learning (ICML), Lille, France, 2015, pp. 448–456.

**[15]**M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 2019, pp. 6105–6114.

**[16]**K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.

**[17]**A. Howard et al., "Searching for MobileNetV3," in Proc. IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, South Korea, 2019, pp. 1314–1324.

**[18]**Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 11976–11986.

**[19]**K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in Proc. 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 2015.

**[20]**A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in Proc. 9th International Conference on Learning Representations (ICLR), 2021.

### *Optimisation, Regularisation, and Imbalanced Learning*

**[21]**D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 2015.

**[22]**I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in Proc. 7th International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 2019.

**[23]**I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in Proc. 5th International Conference on Learning Representations (ICLR), Toulon, France, 2017.

**[24]**H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in Proc. 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 2018.

**[25]**T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in Proc. IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980–2988.

**[26]**H. He and E. A. Garcia, "Learning from imbalanced data," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, pp. 1263–1284, Sep. 2009.

### *Transfer Learning*

**[27]**S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

### *Ensemble Learning*

**[28]**T. G. Dietterich, "Ensemble methods in machine learning," in Proc. First International Workshop on Multiple Classifier Systems (MCS), Cagliari, Italy, 2000, pp. 1–15.

**[29]**L. I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms. Hoboken, NJ, USA: Wiley-Interscience, 2004.

**[30]**D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," Journal of Artificial Intelligence Research, vol. 11, pp. 169–198, 1999.

### *Fingerprint Matching and AFIS*

**[31]**R. Vazan, "SourceAFIS: Fingerprint recognition engine," version 3.18.1, 2023. [Online]. Available: https://sourceafis.machinezoo.com/

**[32]**R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 12, pp. 2128–2141, Dec. 2010.

### *Benchmark Datasets*

**[33]**C. I. Watson, "NIST Special Database 4: Fingerprint Database," National Institute of Standards and Technology, Gaithersburg, MD, USA, Tech. Rep. NISTIR 5212, 1992.

**[34]**D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2004: Third fingerprint verification competition," in Proc. International Conference on Biometric Authentication (ICBA), Hong Kong, China, 2004, pp. 1–7.