Advancements and Challenges In Fingerprint Recognition Using Minutiae Matching : A Comprehensive Survey

Author: Manha Garg
(Email, Orcid ID)
Supervisor: Gregory Wagner
College of Engineering and Computer Science
Syracuse University, Syracuse, NY USA - 13244

ABSTRACT

Fingerprint recognition through minutiae matching represents a significant domain within biometrics and pattern recognition. The central goal is to formulate precise and efficient algorithms for individual identification and authentication, leveraging the distinctive fingerprint patterns. This authentication process hinges on matching minutiae points, such as ridge endings and bifurcations. In this survey, we explore diverse methodologies, including the Onion Peeling Approach, Contact vs Contactless Fingerprint extraction, and the influence of Deep Learning Neural Networks, to automate fingerprint matching.

Within this domain, the primary challenge pertains to the precise extraction and representation of minutiae features, alongside the development of effective matching algorithms for inter-fingerprint comparison. The efficacy of fingerprint recognition systems is intrinsically linked to the quality of fingerprint image capture, the robustness of feature extraction, and the efficiency of matching algorithms. This survey aims to scrutinize the contemporary landscape by investigating state-of-the-art approaches, advancements in minutiae matching techniques, and strategies employed to enhance fingerprint recognition system performance.

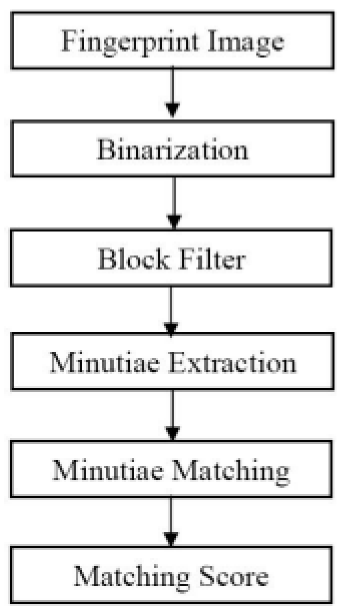
Furthermore, this survey identifies prevalent challenges faced by researchers in this arena and offers potential solutions and future research trajectories. By delving into this evolving field, I seek to provide a comprehensive overview for researchers, practitioners, and policymakers, offering insights into the latest methodologies, challenges, and promising avenues for the advancement of fingerprint recognition technology.

KEYWORDS

Fingerprint recognition Minutiae matching Biometrics Automation Matching algorithms Convolutional Neural Network(CNN) Ridges Valleys Pattern Matching Nested Convex Polygon(NCP) Minutiae Cylinder Code(MCC) Automated FingerPrint Recognition(AFR) Minutiae texture cylinder codes (MTCC). Automatic Fingerprint Identification System(AFIS)

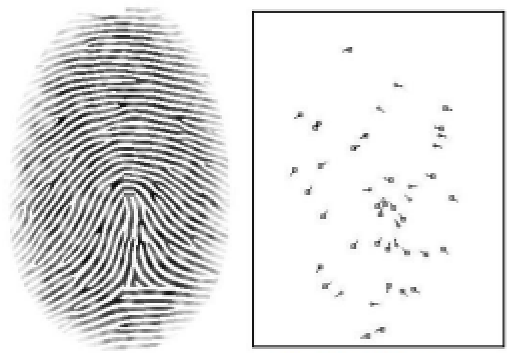
INTRODUCTION

Minutiae matching is a fundamental concept and technique within the field of fingerprint recognition, a biometric authentication method. Fingerprint recognition relies on capturing and analyzing the unique patterns of ridges and valleys on the surface of an individual's fingertip. Minutiae are specific points on these ridges where the ridge lines end (ridge endings) or bifurcate (ridge bifurcations). These minutiae points serve as distinct and stable landmarks that can be used to compare and match different fingerprint images.
Minutiae matching involves the comparison of these minutiae points between a captured fingerprint (input) and a stored reference fingerprint in a database. The goal is to determine whether the two fingerprints belong to the same individual. [12]



1. Minutiae Matching : Let’s Understand the different steps in the process of Minutiae Matching

1.1 Minutiae Extraction : Automated algorithms are employed to locate and extract the minutiae points from fingerprint images. These algorithms identify the locations, orientations, and types of minutiae (ridge endings or bifurcations)[12].



(a) (b)

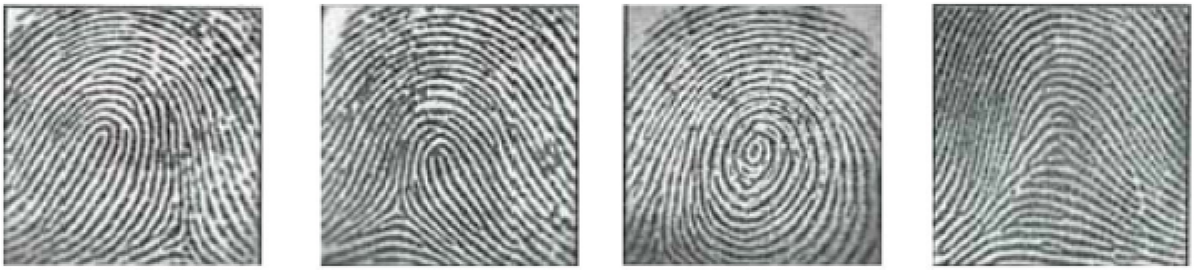
: (a) Gray-scale Fingerprint (b) Minutiae points.

The effectiveness of fingerprint matching techniques lies in their ability to capture and compare the unique ridge patterns that characterize each individual's fingertips. Among the innovative approaches in this field, the Onion Peeling Approach and Turning Function have emerged as intriguing methodologies that offer new dimensions to the art of fingerprint matching.

1.1.1 The Onion Peeling Approach [11]: The Onion Peeling Approach is a novel technique that involves the iterative thinning of fingerprint ridges, layer by layer, akin to peeling layers of an onion. By progressively thinning the ridges, the approach unveils underlying patterns and structures that might remain hidden in the original image. The method capitalizes on the concept that the core of a fingerprint contains essential information for matching, which becomes more discernible as surrounding ridges are peeled away. The resulting core information, combined with minutiae points, contributes to a more comprehensive and accurate matching process.

1.1.2 The Turning Function [11]: The Turning Function is an innovative concept that focuses on capturing the curvature and orientation changes along the fingerprint ridges. Fingerprint ridges exhibit distinct patterns of curvature and orientation, akin to the unique contours of a terrain. By analyzing these curvature changes, the Turning Function extracts distinctive features that can enhance the discriminative power of fingerprint matching. This technique complements traditional minutiae-based matching by incorporating additional ridge-level information, thus contributing to a more holistic representation of fingerprints.

1.2. Minutiae Representation [11]: The extracted minutiae are often represented using numerical coordinates and orientation angles. This representation allows for efficient storage and comparison.



(a) left loop (b) right loop (c) whorl and (d) arch

1.3. Matching Algorithm [4]: Various algorithms are utilized to compare the minutiae of the input fingerprint with those of the reference fingerprint. These algorithms assess the spatial relationships, angles, distances, and other characteristics of the minutiae to determine the degree of similarity.

1.3.1 Geometric Fingerprint Recognition via Oriented Point-Set Pattern Matching[4] is a research topic covered in this paper that involves the application of geometric and pattern recognition techniques to fingerprint identification. This approach focuses on extracting geometric features from fingerprints and using oriented point-set patterns to match and compare fingerprints for identification purposes.Oriented point-set pattern matching is a technique used to align and match two sets of points based on their relative orientations. In the context of fingerprint recognition, this technique is applied to compare the geometric patterns formed by the ridges and valleys of a fingerprint.

Minutiae matching is favored for its reliability and robustness. It is not influenced by the overall size, position, or rotation of the fingerprint, focusing instead on the specific minutiae points that remain relatively stable across different fingerprint impressions of the same individual. However, the accuracy of minutiae matching can be affected by factors such as image quality, noise, variability in minutiae extraction, and the chosen matching algorithm.

1.3.2 Types of Matching : Various approaches can be used to perform minutiae matching techniques. Here are the different types of Matching :

- Point-to-Point Matching[11]: This fundamental approach involves directly comparing individual minutiae points, such as ridge endings and bifurcations, between two fingerprint images. A similarity score is computed based on the spatial proximity and orientation of the minutiae points.
- Local Ridge Orientation Matching [6]: Instead of comparing individual minutiae points, this method considers the local ridge orientation patterns around each minutia. The orientation information is used to establish correspondences and calculate a similarity measure.

- Pairwise Minutiae Matching [12]: In this technique, pairs of minutiae are compared to assess their geometric relationship, including distance, angle, and orientation. Pairwise matching enables the consideration of multiple minutiae in combination.Fingerprint recognition using minutia score matching is a biometric identification technique that focuses on comparing and matching the minutiae points present in fingerprints to establish the identity of individuals.
- Triplet-based Matching[6]: This approach extends pairwise matching by incorporating a third minutia point. The relative angles and distances between the three minutiae points are analyzed to establish a more robust matching criterion.
- Graph-based Matching: Fingerprint minutiae are represented as nodes in a graph, and edges represent potential associations between minutiae. Graph theory algorithms are then employed to find the best matching configuration that optimizes the overall graph structure.
- Topological Minutiae Matching: This technique considers the topological arrangement of minutiae points, focusing on the relative positions and sequences of ridge endings and bifurcations. It emphasizes the structural relationships within the fingerprint.
- Pattern-based Matching[4]: Here, minutiae are grouped or clustered based on their spatial arrangement, forming distinctive patterns. Matching involves identifying and comparing these patterns, enhancing robustness against noise and variability.
- Ridge Flow Matching[1: This approach considers the overall ridge flow direction between minutiae points. By analyzing the paths of ridges between corresponding minutiae, a more comprehensive understanding of fingerprint similarity is achieved.
- Contextual Minutiae Matching: Contextual information, such as ridge count, curvature, and ridge shape, is used alongside minutiae data to enhance the matching process. This approach exploits additional fingerprint characteristics beyond minutiae alone.
- Hybrid Matching[6]: Hybrid approaches combine minutiae matching with other fingerprint recognition techniques, such as texture analysis or spectral analysis, to create a more comprehensive and accurate matching system.

These different types of minutiae matching techniques represent a diverse array of strategies aimed at capturing the unique attributes of fingerprints and enhancing the accuracy and robustness of fingerprint recognition systems. This paper will discuss the advancements made in different stages of minutiae matching and different matching techniques and comparing the results.

1.4. Similarity Scoring [12]: A similarity score is calculated based on the degree of resemblance between the two sets of minutiae. Different algorithms may use different scoring methods, such as Euclidean distance, angular difference, or more complex mathematical models.

1.5. Decision Threshold [14]: A decision threshold is applied to the similarity score to determine whether the fingerprints are a match or not. If the score exceeds a certain threshold, the fingerprints are considered a match; otherwise, they are considered non-matching.

2. Paradigms of Matching : As technology advances, distinct paradigms for fingerprint acquisition have emerged too: contact-based and contactless fingerprint recognition. Both methodologies aim to capture the intricate ridge patterns and minutiae landmarks that characterize each individual's fingerprints. However, they differ in their approach to acquisition and subsequent minutiae matching.

> **2.1 Contact-based fingerprint recognition** entails direct physical interaction between the fingertip and a sensor surface. This method, characterized by historical significance and established reliability, involves pressing the finger against a surface to acquire a high-resolution fingerprint image.
> **2.2 Contactless fingerprint recognition** dispenses with physical touch, capturing fingerprint data from a distance through technologies like optical sensors, infrared illumination, or capacitive sensing.

We explore minutiae extraction techniques for both paradigms, considering factors like image quality, accuracy, and robustness and explore both the Minutiae Based Approach and the Deep Learning Based Approach using Convolutional Neural Network(CNN) on Contactless FingerPrint Recognition system, in detail in the upcoming sections.The objective of this work is to understand the development of a contactless fingerprint recognition system (CFRS)[1], incorporating both deep learning and standard fingerprint matching algorithms.

3. Automated Fingerprint Recognition with Minutiae Matching [10] : Automated Fingerprint Recognition (AFR) is a biometric technology that involves the automatic identification and verification of individuals based on their fingerprint patterns. It is widely used for various applications, including access control, law enforcement, forensic investigations, and border security. One of the key techniques used in AFR is minutiae matching.

> **3.1 Minutiae Matching**: Minutiae matching is a method within AFR that focuses on capturing and comparing specific points of interest, known as minutiae, present in fingerprint patterns. Minutiae points include ridge endings (where ridges terminate) and bifurcations (where ridges split). These minutiae points serve as unique landmarks that can be used to establish the identity of an individual. The process involves extracting minutiae from a captured fingerprint image and then comparing them with minutiae from a reference fingerprint template stored in a database.
>
> **3.2 Large Fingerprint Databases**: With the proliferation of biometric systems and the need for accurate identification, the size of fingerprint databases has grown significantly. Large fingerprint databases contain vast numbers of fingerprint records from various sources, such as criminal databases, immigration records, and civil registrations. Managing and searching through these databases efficiently and accurately is a challenge.

4. Deep Networks for Fingerprint Recognition [3]: Deep neural networks, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have demonstrated remarkable capabilities in capturing intricate patterns and features from complex data. In the context of fingerprint recognition, deep networks can be customized to learn discriminative representations from fingerprint images, thereby enhancing the system's ability to differentiate between individuals. This paper investigates the design and architecture of deep networks optimized for fingerprint recognition, considering aspects such as network depth, convolutional layers, pooling strategies, and activation functions.

4.1 Incorporating Fingerprint Domain Knowledge: Fingerprint domain knowledge encapsulates insights into the structural characteristics of fingerprints, including ridge orientation, minutiae distribution, and ridge count. Integrating this domain knowledge into deep networks can bolster their feature learning capabilities, resulting in more informative and robust fingerprint representations. Techniques for incorporating fingerprint domain knowledge range from designing specialized network layers to pre-processing fingerprint images to highlight relevant features [2]. This integration fosters a synergy between data-driven feature learning and domain-specific insights, leading to enhanced recognition accuracy and resilience.

4.2 Experimental Evaluations and Real-World Applications: To validate the efficacy of integrating deep networks and fingerprint domain knowledge, experimental evaluations are conducted across diverse datasets and scenarios. Recognition accuracy, robustness against variations, and adaptability to real-world conditions are quantified and analyzed. Real-world applications spanning access control, forensic identification, and identity verification underscore the practical relevance of this integration. These applications showcase how the combined strengths of deep networks and fingerprint domain knowledge result in systems that outperform traditional methods and adapt effectively to dynamic operational contexts.

5. Genetic programming (GP)[7] is a computational approach inspired by the principles of natural selection and evolution. In the context of fingerprint matching, a GP framework offers a unique and innovative methodology for enhancing the accuracy and effectiveness of fingerprint recognition systems. The essence of GP lies in its ability to evolve and optimize complex solutions through iterative generations, thereby addressing the challenges posed by the intricacies of fingerprint patterns and variations.

Fingerprint matching using a GP framework involves the following key components:

5.1 Representation: Genetic programming represents potential solutions, known as individuals or candidate programs, as structured trees. These trees consist of functions, terminals, and other components that encode fingerprint matching strategies. The genetic material within these trees evolves over generations.

5.2 Fitness Function: A fitness function evaluates the performance of each individual in the population. In the context of fingerprint matching, the fitness function assesses the accuracy and robustness of the candidate solution in comparing and matching fingerprint features.

5.3 Genetic Operators: Genetic programming employs genetic operators such as crossover and mutation to create new individuals. Crossover involves combining genetic material from two parent individuals to produce offspring, while mutation introduces random changes to an individual's genetic makeup.

5.4 Evolutionary Process: Through successive generations, the genetic programming framework evolves the population of candidate solutions. Over time, the framework selects individuals with higher fitness values, leading to the propagation of more effective fingerprint matching strategies.

5.5 Advantages of Genetic Programming for Fingerprint Matching are Adaptability,Complex Relationships,Feature Fusion,Reduced Manual Tuning and Robustness.

5.6 Challenges and Considerations are Computational Complexity, Interpretability and Overfitting.

Ongoing research in this field is discussed in the paper that aims to refine and optimize the application of genetic programming for fingerprint matching, ultimately contributing to the evolution of biometric authentication technology.

Therefore, through a systematic exploration of existing literature, this survey paper synthesizes the latest advancements, prevailing trends, and unexplored research avenues within the realm of fingerprint matching using minutiae analysis.

1. MINUTIAE MATCHING

Minutiae matching is a fingerprint recognition technique that focuses on identifying and comparing specific points of interest within a fingerprint. These points, known as minutiae, include ridge endings (where ridges in the fingerprint end) and bifurcations (where ridges split into two). This section will discuss innovative research that has been done recently to enhance each step of the process and increase the effectiveness and reliability at each step.

1.1 Minutiae Extraction and Representation

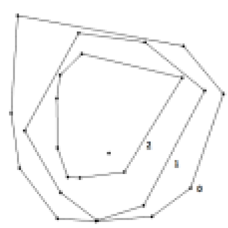
This section presents a new minutiae-based fingerprint matching using the onion peeling approach, proposed in Oct 2021 by N.Padkan and M.Reza [11]. In the proposed method, fingerprints are aligned to find the matched minutiae points. Then, the nested convex polygons of matched minutiae points are constructed and the comparison between peer-to-peer polygons is performed by the turning function distance.The performance of the proposed algorithm is evaluated on the database FVC2002. The results show that fingerprints of the same fingers have higher scores than different fingers.

1.1.1 Onion Peeling Method -

The onion peeling approach in minutiae matching is a fingerprint recognition technique that aligns and compares two fingerprint images for authentication or identification. It involves aligning the images which involves involves translation, rotation, and scaling of one of the fingerprint images to match the orientation and size of the other, comparing minutiae points in concentric circles around a central point , resembling the layers of an onion (hence the name) , scoring the matches, applying a threshold, and making a final decision based on matching scores.

ALGORITHM 1:

Consider S = {P1, P2, ..., Pn} as n points in two dimensional space. The convex layers of S are a sequence of nested convex polygons (NCPs) having the points as their vertices while the outer layer is the convex hull of the points and the rest is formed in a recursive manner. The inner layer may consist of one or two points.

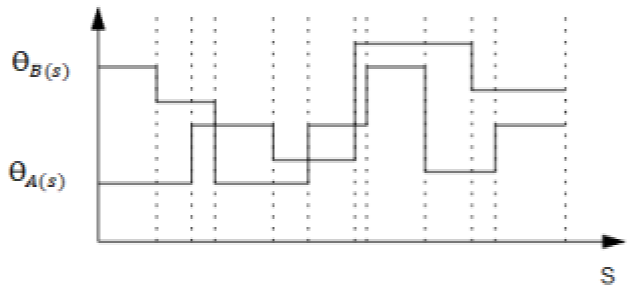


The constructed NCPs of a set point in two-dimensional pace

The onion peeling, by itself as a computational geometry algorithm is not widely used in computer vision issues such as image retrieval and image matching.The turning function however is invariant to scaling, since the polygons is re-scaled and also invariant under translation and rotation because of having no orientation information . These properties make it ideal for measuring the similarity between shapes and it can be used in computer vision and specially image retrieval, to enhance the results of Onion Peeling Algorithm.

1.1.2 Turning Function

The turning function is an easy and standard tool for representing and describing shapes and in particular polygons. The turning function starts from a certain point, called reference, that is arbitrarily selected on the shape's boundary. The function measures the counterclockwise tangent as a function of arc s . Θ(0) is the angle that reference point makes with x-axis. $\Theta_A(s)$ keeps track of the turning that takes place along the shape, increasing with counterclockwise turns and decreasing with clockwise turns.The distance between two polygons A and B is defined as distance between their turning functions $\Theta_A(s)$ and $\Theta_B(s)$.



The distance between two polygons is calculated using rectangles enclosed by $\theta_A(s)$, $\theta_B(s)$, and dotted lines

Rotating the polygons and choosing different points as the reference point affects the distance. Therefore, the minimum distance over all different reference points and rotation is computed.

1.1.3 Combined Algorithm

The proposed algorithm is divided into two processes. In the first process, minutiae points are extracted using MinuE(). Minutiae are considered as triplets m = {(x, y), θ, type} with three elements which are the minutiae location, orientation, and type. The type can be ridge ending or ridge bifurcation. Then, minutiae pairs are found in the input and template fingerprints using MM(). In the next step, the nested convex polygons (NCP) of matched minutiae points are constructed and the distance between corresponding turning functions is computed.

Algorithm 2 The Proposed Fingerprint Matching Algorithm

Input: Input and template fingerprint images
Output: Matching score over the average of turning distances and minutiae matching

Extract minutiae points of the input and template fingerprints:
I = **MinuE**(Input);
T = **MinuE**(Template);
Find matched minutiae points of the input and template fingerprints:
for i in I **do**
  for j in T **do**
    I_k , T_k = **MM**(I,T);
  end for
end for
Construct the NCPs of matched minutiae points:
for i in I_k **do**
  L_i = **NCP**(I_k);
end for
for j in T_k **do**
  L_j = **NCP**(T_k);
end for
Compute the turning functions of the NCPs in the input and template fingerprints:
$TF1$ = **TF**(L_i);
$TF2$ = **TF**(L_j);
Compute the average of distances between **TD**($TF1$,$TF2$);
Compute the similarity score.

Step 1 : Minutiae Matching

The aim is to maximize the number of correct matched minutiae. So, aligning the input and template fingerprints is an obligatory step.Minutiae mi in the input fingerprint and minutiae m0 j in the template fingerprint will be matched, if their orientation difference D and spatial distance S are smaller than specific tolerances r0 and θ0, respectively.Consider (s,∆θ,∆x,∆y) is the set of transformations that is needed to be estimated to find the best matching alignment:

Implementation of minutiae matching on two fingerprints of the same finger from the FVC2002 db2.

Step 2 : Nested Convex Polygons of Matched Minutiae Points

After detecting the matched minutiae points, their NCPs are constructed in the fingerprints. Let I_k = {m_1, m_2, ..., m_k} and T_k = {m'_1 , m'_2 , ..., m'_k } be the sets of matched minutiae points on the input and template fingerprint images. Using the onion peeling approach, the nested convex polygons are constructed for the sets I_k and T_k . The depth and vertices of each polygon are stored.

(a)

The nested convex polygons of minutiae points in two fingerprints of same finger.

Turning function TF() is used here for representing and describing nested convex polygons of minutiae points. After creating all the turning functions of the nested convex polygons, it needs to check the distance in the optimal orientation between the two peer-to-peer convex polygons using TD(). The number of matched minutiae are related to the average of turning function to have a better score for comparing fingerprints, using Matching Score alpha (Scaled to lie between 0 and 1)

$$\alpha = \frac{Minutiae \;\; score}{Average}$$

1.1.4 System Evaluation and Results

The result of comparing the input and template fingerprint images is an amount between 0 and 1 called "matching score" that shows the similarity between those images. Closer the score to 1 shows more similarity between two fingerprint images. The decision of considering two fingerprints as matching or non-matching pairs is defined by regulating a threshold. Two fingerprints are considered as matching pairs if the matching score is higher than or equal to the threshold t and fingerprints whose matching score is lower than t are regarded as non-matching pairs. If the input fingerprint is matched to an enrolled fingerprint, this pair is called a genuine match. When the comparison is between samples from different users, this is called an impostor match.

Table 1: Results of Implementing the Proposed Algorithm to Same Fingers of FVC 2002 DB2

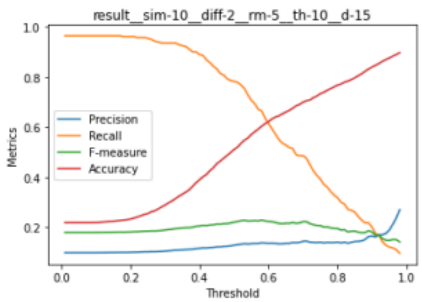
Fingerprint Names	Minutiae Matching Score	the number of convex polygons	average	Final Score
101_1 101_2	0.44	5 5	26.78	0.98
101_1 101_3	0.33	3 4	52.97	0.75
101_5 101_6	43.54	3 3	43.54	0.77
103_2 103_1	0.38	4 4	45.68	0.88
103_6 103_7	0.17	3 2	33.3	0.73
105_1 105_4	0.25	2 2	73.2	0.57
107_3 107_6	0.26	3 3	63.53	0.65

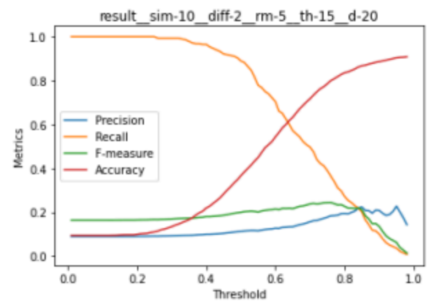
Table 2 : Results of Implementing the Proposed Algorithm to Different Fingers of FVC 2002 DB2

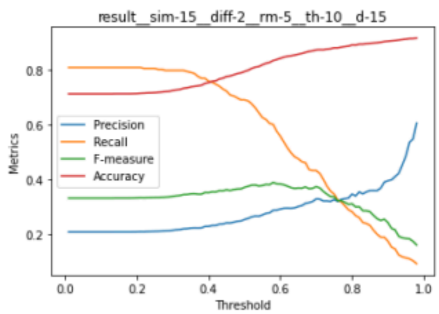
Some results of implementing the proposed algorithm to different fingers of $FVC2002\ DB2\_b$.

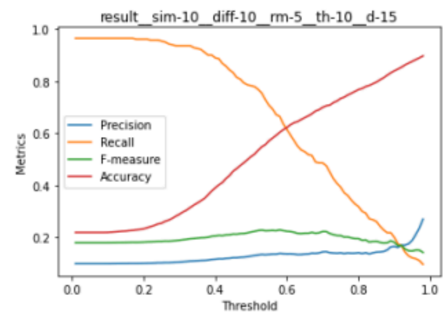
Fingerprint Names	Minutiae Matching Score	The Number of Convex polygons	Average	Final Score
104_2 110_1	0.16	2 2	0.146	0.54
109_8 102_7	0.10	2 3	0.20	0.31
109_5 107_7	0.12	1 1	0.11	0.51
108_6 110_4	0.13	1 1	0.13	0.49
102_1 101_1	0.15	2 2	0.24	0.36
103_6 102_7	0.16	2 2	0.23	0.39
104_5 103_3	0.13	2 2	0.147	0.49
110_4 101_1	0.21	3 3	0.19	0.53

As Observed, fingerprints of the same finger (Table 1) have higher minutiae matching score in comparison with those from different fingers (Table 2) and as a result, the number of nested convex layers is also higher. The comparison of the last two columns of tables shows a higher score in Table 1. Regulating an appropriate threshold t is a vital step that should be regarded. Fingerprints whose score is higher than t are regarded as matched and those with lower score considered as non-match pairs. Comparing the Performance Metrics for different values of distance, similarity and threshold, we see that fewer error rates are obtained when sim = 15, diff = 2, rm = 5, th = 10, and d = 15.









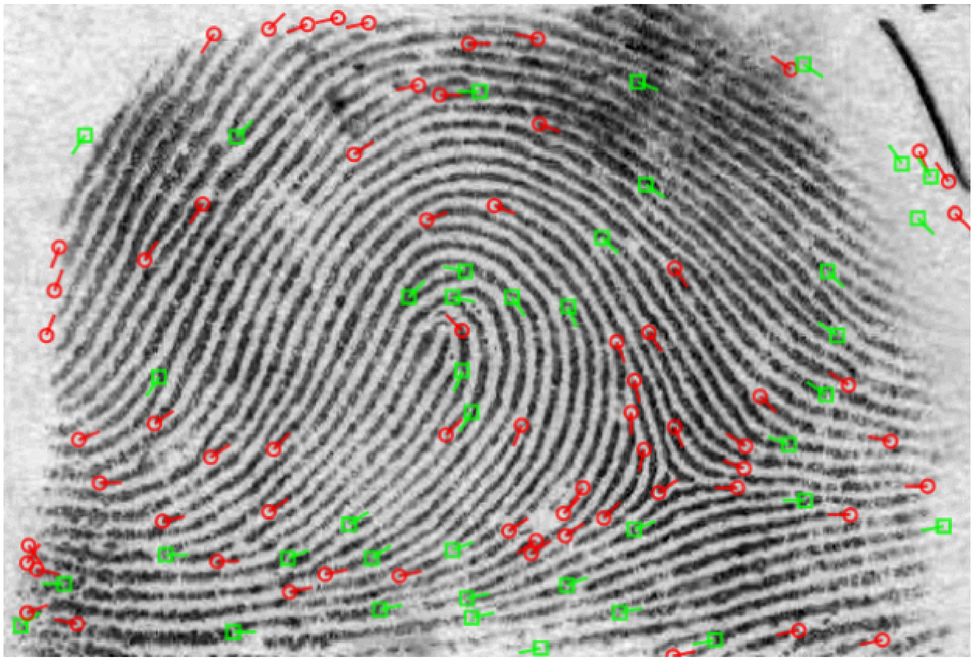
In Conclusion,the results show that the proposed algorithm is efficient for matching fingerprints. The score of matching fingerprints of the same finger is approximately higher than the score of different fingers. The best results are obtained when fingerprints with minutiae matching score lower than 0.15 and layer differences more than 2 are ignored. In this case,the best threshold for comparing fingerprints is 0.72 and accuracy and precision are 0.85 and 0.31, respectively.

2. MATCHING ALGORITHMS

The evolving landscape of applications, data variability, security concerns, performance requirements, and privacy considerations continues to drive the development of innovative fingerprint matching algorithms, some of which are covered in this section.

2.1 Oriented Point Set Pattern Matching

An important consideration while minutiae matching is that the minutiae are not pure geometric points: besides having geometric positions, defined by (x, y) coordinates in the respective images, each minutiae point also has an orientation (the direction of the associated ridges), and these orientations should be taken into consideration in the comparison.



Screenshot of the display of fingerprint minutiae in NIST's Fingerprint Minutiae Viewer

In a 2018 paper [4] by David Epstien and his colleagues, a set of new minutiae matching algorithms are proposed using Oriented Point Set Pattern Matching to counter this problem using computational geometry.
All instances of pointset pattern matching problems are considered, where given a "pattern" set, P, of m points in R^2 and a "background" set, B, of n points in R^2 , a transformation of P that best aligns the points of P with a subset of the points in B is needed.
In other words, the problem is to find a transformation of P that minimizes the farthest any point in P is from its nearest neighbor in B. Rather than only considering the positions of the points in P and B, however, in this paper they consider instances in which each point in P and B also has an associated orientation defined by an angle, as in the fingerprint matching application. This is done using an underlying distance,which combines information about both the locations and the orientations of the points, Hausdorff Distance.

$$h(P', B) = \max_{p \in P'} \min_{q \in B} \rho(p, q)$$

These new algorithms are needed because optimal solutions calculated by existing algorithms generally have high polynomial running times.

2.1.1 Base Algorithm

Each of these algorithms uses as a subroutine a base algorithm that selects certain points of the pattern, P, and "pins" them into certain positions with respect to the background, B. This choice determines a transformed copy P' of the whole point set P. We then compute the directed Hausdorff distance for this transform by querying the nearest neighbor in B for each point of P' To find nearest neighbors for a suitably-defined metric on oriented points that combines straight-line distance with rotation amounts, we adapt balanced box decomposition (BBD) trees to oriented point sets, which may be of independent interest.

Algorithm BaseTranslate(P, B):

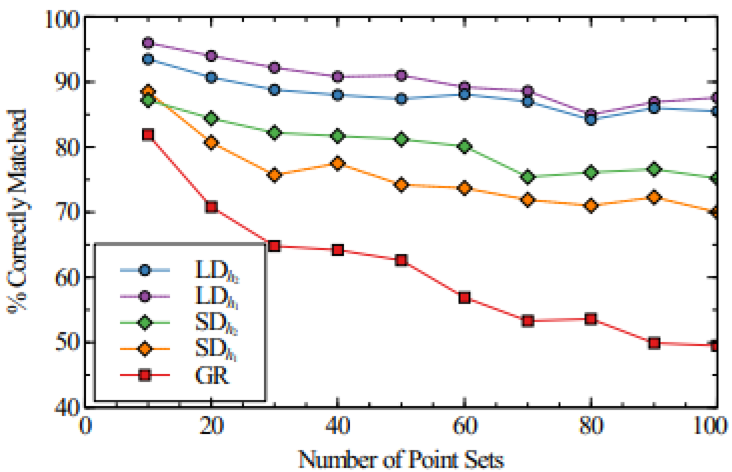
Choose some $p \in P$ arbitrarily.
for every $b \in B$ **do**
  Pin step: Apply the translation, $T_v \in \mathcal{T}$, that takes p to b.
  for every $q \in T_v(P)$ **do**
    Query step: Find a nearest-neighbor of q in B using the μ_i metric, and update a candidate Hausdorff distance for T_v accordingly.
  end for
  return the smallest candidate Hausdorff distance found as the smallest distance, $h_i(T_v(P), B)$.
end for

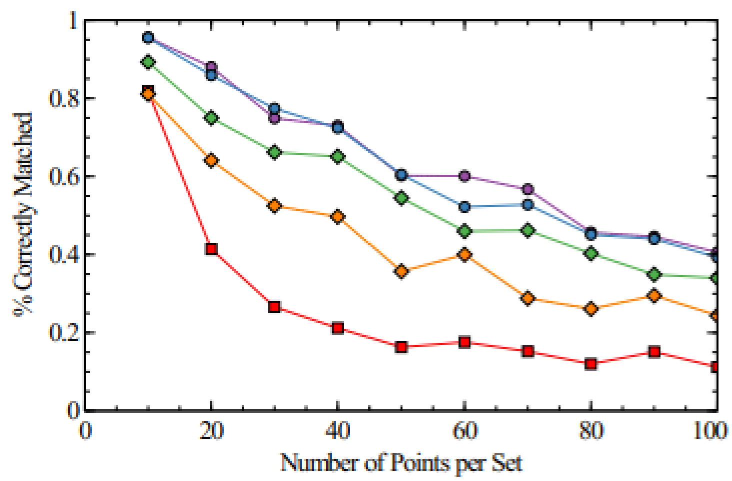
2.1.2 Results

Several Modified Versions of the Base Algorithm were experimented with by changing the Translation, Rotation and Scaling with Big and Small Diameter and their accuracy results were compared. Each algorithm was used to match the modified pattern of points against each of the background point sets and it was considered a success if the background set from which the pattern was derived had the smallest distance.

Below Figure shows the results of this experiment under two variables: the number of background sets from which the algorithms could choose, and the size of the background sets. Each data point is the percentage of successes across 1000 different pattern sets.In reporting the results of our experiments, we use the following labels for the algorithms:

• GR: the non-oriented translation and rotation algorithm.
• LDh1/h2 : the base version of the large diameter algorithm using either the h1 or h2 distance metric.
• SDh1/h2 : the base version of the small diameter algorithm using either the h1 or h2 distance metric.





In every case, the oriented algorithms are more successful at identifying the origin of the pattern than GR. LD was also more successful for each distance metric than SD.

2.2 Minutiae Matching using Descriptor and Clustering

This section discusses fingerprint indexing which is used to reduce the identification time without the loss of accuracy of the fingerprint recognition system. Since a search with the fingerprint indexing is performed by comparing the indexing features, we can regard the fingerprint indexing as a kind of matching process.
The first stage of matching is the fingerprint indexing and it has a low computational cost, while the second stage is the original 1:1 matching and it has a high accuracy, however, the matching time is long.

In the paper[6], authors Guang Ri and Chong Ri propose a novel fingerprint indexing approach for speeding up in the fingerprint recognition system. The kind of features used for indexing and employing the extracted features for searching are crucial for indexing. In this approach, a minutia descriptor is selected, which has been used to improve the accuracy of the fingerprint matching, as a local feature for indexing and a fixed-length feature vector is constructed which will be used for searching from the minutiae descriptors of the fingerprint image using a clustering. A fingerprint searching approach that uses the Euclidean distance between two feature vectors as the similarity between two indexing features is proposed.

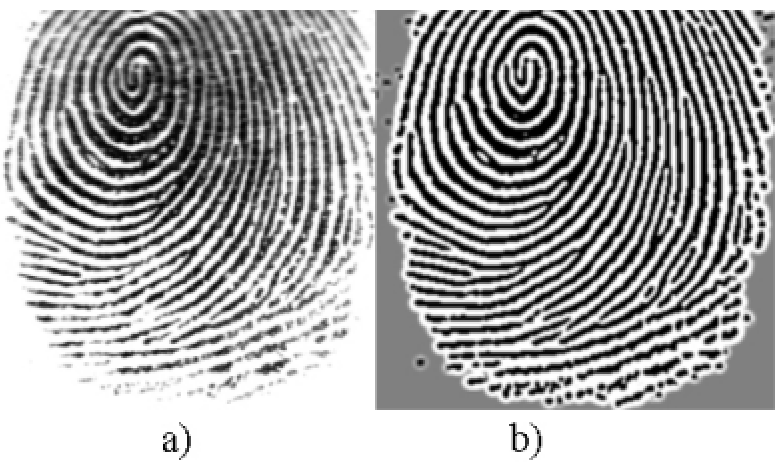
This indexing approach has several benefits. It reduces searching time significantly and is irrespective of the existence of singular points and robust even though the size of the fingerprint image is small or the quality is low. And the constructed indexing vector by this approach is independent of the features which are used for indexing based on the geometrical relations between the minutiae, like one based on the minutiae triplets.

2.2.1 Construction of an indexing feature vector

The proposed minutia descriptor is a vector of a fixed length. Below is the procedure for constructing the minutia descriptor:

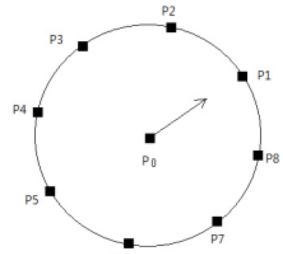
Step 1 : Extracting minutiae

Step 2 : Enhancement of the fingerprint image through DOG(Difference of Gaussian) Filtering.



B is Enhanced Image

Step 3 : Calculation of Gabor wavelet coefficients: For every minutia, they took 9 sampling points adopting the minutia as the center and the direction of the minutia as the reference direction.



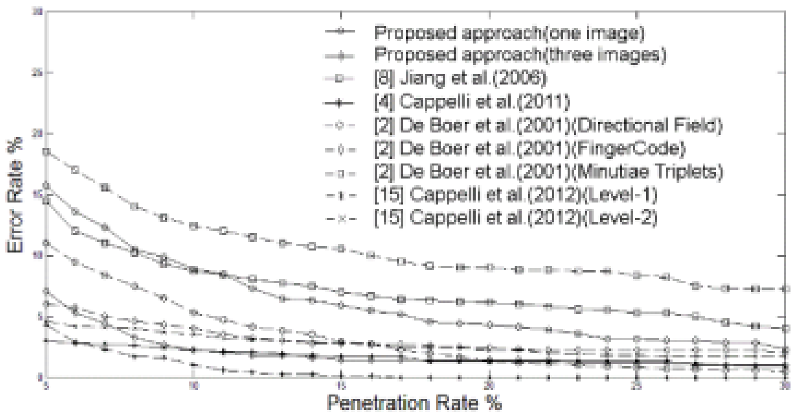
Step 4 : Construction of the Linear Discriminant Analysis transform

Step 5 : Performing a Linear Discriminant Analysis to the Gabor wavelet feature vectors

Step 6: Construction of the feature vector of a fixed length for the search.
Step 7: Clustering in the local feature space: All the calculated minutiae descriptors are collected to get a training set for the clustering. 200 central vectors of the clusters are obtained by performing the K-Means algorithm on the training set. Every central vector is a 25-D vector.
Step 8: Feature vector for the search

2.2.2 Observations and Results

The proposed Algorithm is compared with various other algorithms on Database FVC 2000 DB2 and has the lowest search time and a low error rate.



Method	**Search Time (ms)**
[8] Jiang et al.(2006)	**67**
[15] Cappelli et al. (2012) Level-1	**1.6**
[15] Cappelli et al. (2012) Level-2	**14**
[15] Cappelli et al. (2012) Fusion	**16**
Proposed method	**0.3**

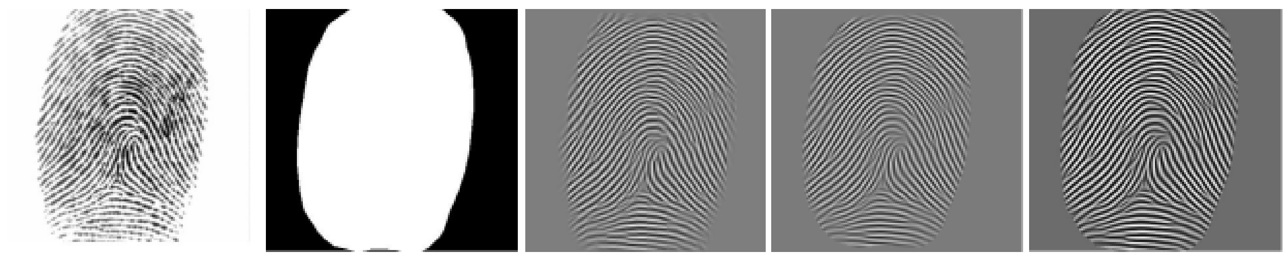
2.3 Minutiae Matching Using Cylinder Codes

Minutia Cylinder Codes (MCC) are minutiae based fingerprint descriptors that take into account minutiae information in a fingerprint image for fingerprint matching. In a 2018 paper by W.Baig and U.Munir [14], a modification to the underlying information of the MCC descriptor is presented which shows using different features that the accuracy of matching is highly affected by such changes. MCC originally being a minutia only descriptor is transformed into a texture descriptor. The transformation is from minutiae angular information to orientation, frequency and energy information using Short Time Fourier Transform (STFT) analysis. The minutia cylinder codes are converted to minutiae texture cylinder codes (MTCC).

The main contributions by the proposed method are as follows:

• Five variants to the MCC descriptor by incorporating texture features. STFT analysis lays the foundation for texture features. That is, the orientation, frequency and energy features are extracted during STFT analysis.

• Incorporating Successive Mean Quantize Transform (SMQT) enhancement of fingerprint images. Light and faded ridges are made prominent using SMQT. This technique seems to be a better approach than traditional techniques like simple contrast enhancements or histogram normalization. Even when viewed during visualization of SMQT results, the images produced show highly clear ridges.

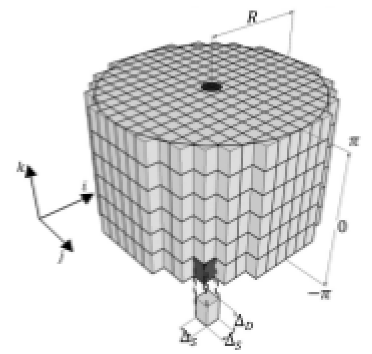


(a) Input image (b) Segmented mask (c) STFT enhanced image. (d) Gabor enhanced image (after STFT analysis). (e) SMQT normalized image. Final ouput

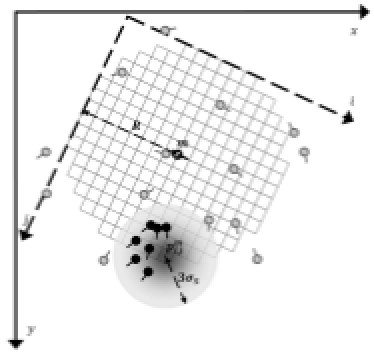
Fig. 1: **Fingerprint enhancement in proposed method.** Using prominent steps STFT, Gabor and SQMT, the proposed method produces very fine enhanced images. STFT analysis of the fingerprint image tends to make the ridges rough after joining broken ridges. The roughness is removed by Gabor filtering. SQMT helps in enhancing light ridges.

2.3.1 MCC/MTCC BASIS STEPS :

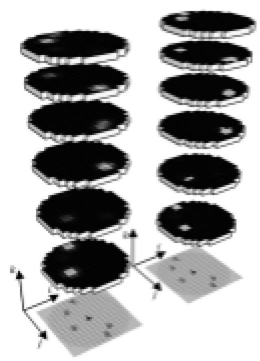
Step 1: Fingerprint image segmentation
Step 2: Fingerprint image enhancement
Step 3 : STFT Analysis - orientation, frequency and energy image generation
Step 4: Minutiae extraction
Step 5: Minutiae Cylinder Codes



(a) A 3D cylinder

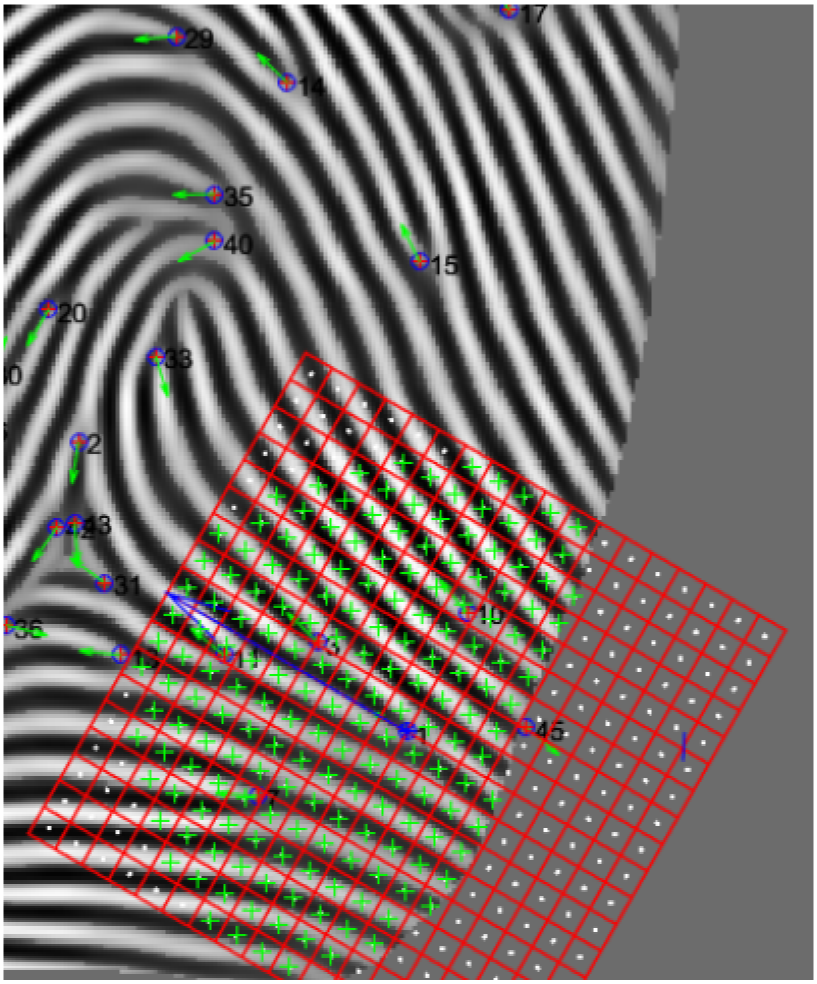


(b) A cylinder slice with cell and its neighbouring minutiae



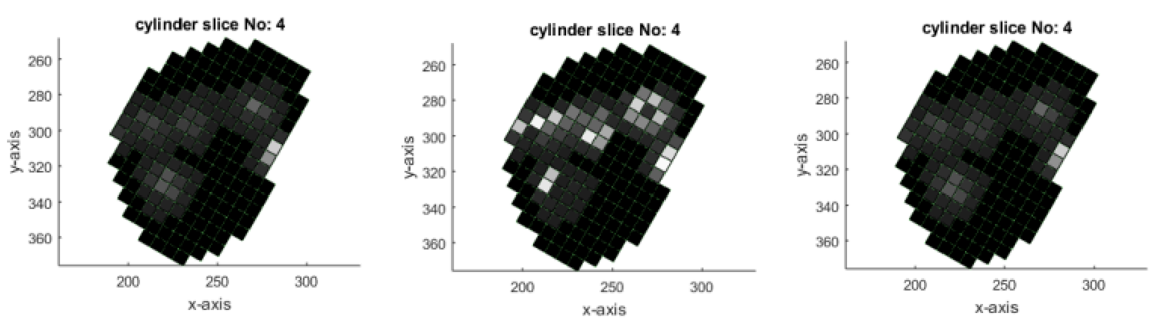
(c) A Cylinder split into sections.

Step 6: Minutiae Texture Cylinder Codes



2.3.2 Observations and Results

Based on a fixed set of parameters like Orientation,Frequency and Energy of a Cylinder, performance of different MCCs are compared.



(a) MCC_{co} cell centered orientation cylinder slice.

(b) MCC_{cf} cell centered frequency cylinder slice.

(c) MCC_{ce} cell centered energy cylinder slice.

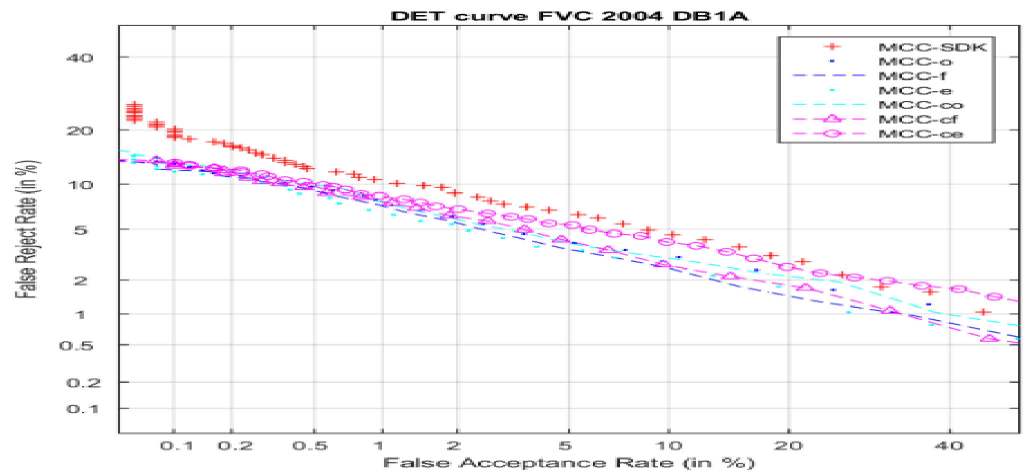
The proposed changes to MCC show improved performance on FVC 2002 and 2004 data sets and surpass the traditional MCC performance.This shows that orientation is a better feature when the database comprises fingerprint images that are of low quality just.

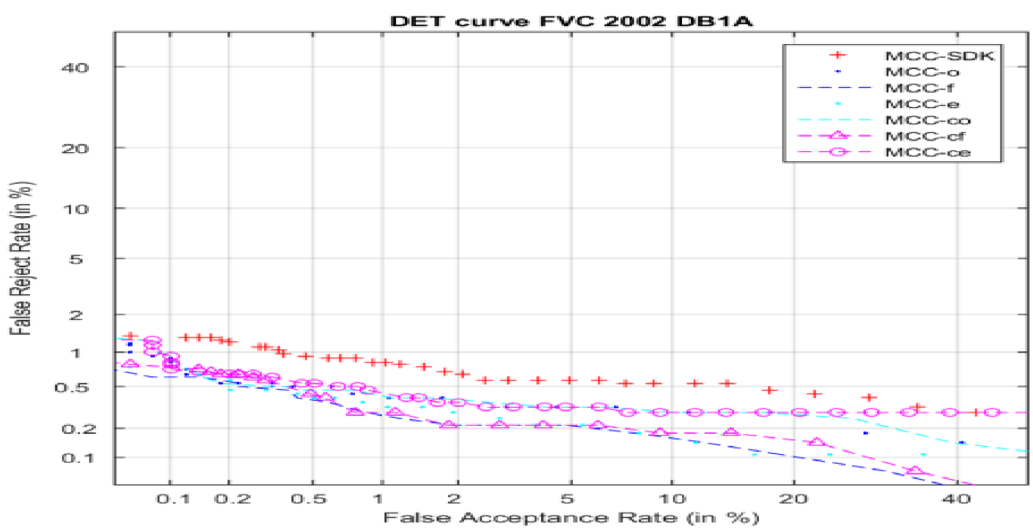
TABLE V: EERs for FVC 2002 DBs.

Feature	DB1A	DB2A	DB3A	DB4A
MCC_{SDK}	0.89	0.9	4.92	1.61
MCC_o	0.54	0.85	4.68	2.03
MCC_f	0.46	0.36	5.89	1.63
MCC_e	0.46	0.38	4.67	1.71
MCC_{co}	**0.42**	0.38	**4.42**	**1.49**
MCC_{cf}	**0.42**	**0.28**	4.24	1.61
MCC_{ce}	0.5	0.39	5.6	1.67

TABLE VII: EERs for FVC 2004 DBs.

Feature	DB1A	DB2A	DB3A	DB4A
MCC_{SDK}	6.07	5.75	4.42	3.57
MCC_o	5.21	6.38	6.00	3.75
MCC_f	4.32	6.00	6.15	2.68
MCC_e	4.25	6.78	6.72	3.20
MCC_{co}	**3.85**	**5.35**	**3.78**	**2.38**
MCC_{cf}	4.00	5.71	5.63	2.39
MCC_{ce}	4.24	6.46	6.74	3.07





3. Contactless FingerPrint Identification

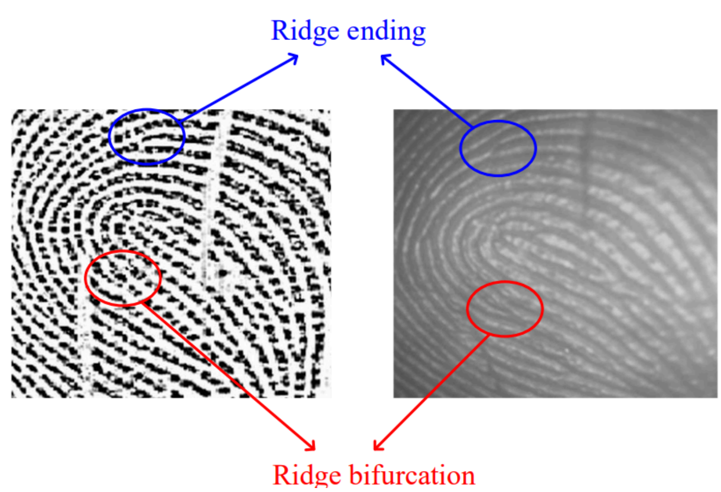
Contact-based fingerprint systems, while widely used and generally effective, have certain limitations and drawbacks such as hygiene concerns, wear and tear on sensors, latency, false rejections, privacy issues, limited coverage, susceptibility to environmental factors, and the need for physical access. These limitations can affect their suitability for specific applications and user preferences.

It is however quite known from the literature that the contactless fingerprint images deliver remarkably low matching accuracies as compared with those obtained from the contact based fingerprint sensors. The 2018 paper by R. Vyas and A. Kumar [13] therefore develops a new approach to significantly improve contactless fingerprint matching capabilities available today. They systematically analyze the extent of complimentary ridge-valley information and introduce new approaches to achieve significantly higher matching accuracy over common fingerprint matchers. They also investigate least explored options for the fingerprint color-space conversions, which can play a key-role for more accurate contactless fingerprint matching.

In Contact based system, when a finger touches the sensor, the pixels corresponding to the light rays reflected from the low topographic regions, or the valleys of the finger skin surface, have higher intensity (bright) in a fingerprint image. The spatial locations from high areas or ridges are rendered as darker regions in the fingerprint image.

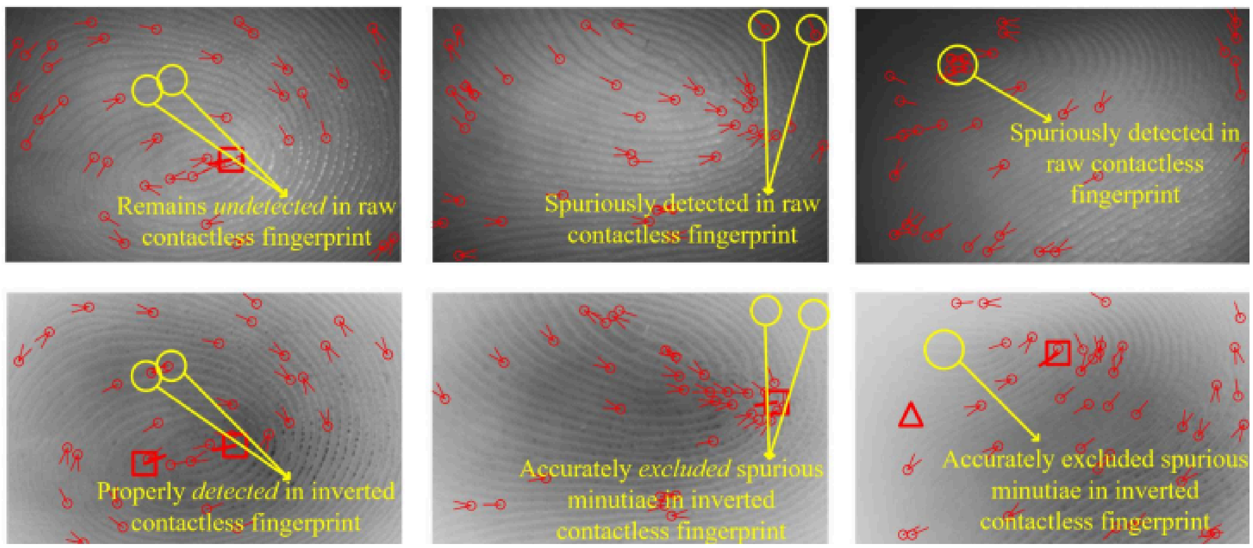
These gray-level alterations can be easily observed from the raw fingerprint images acquired in contactless manner. However, if the images are represented as grayscale, or even binarized to detect the potential minutiae points, this polarity reversal effect is completely lost. This is also the key reason for poor performance when the contactless images are matched against the legacy contact-based fingerprints.

The ridge endings of contact-based fingerprint become ridge bifurcations in the contactless fingerprint and vice-versa. Left Image below is Contact Based and the image on the Right is Contactless.



3.1 Accurate Minutiae Detection from Contactless Fingerprints

It can also be noticed from below images that many spurious minutiae, which are falsely detected from the raw fingerprints, remain undetected in their inverted counterparts. Additionally, the inverted fingerprint facilitates the detection of additional cores and deltas, which otherwise remain undetected from the raw fingerprints.



For instance, the first image sample in the top row of abo shows a sample fingerprint having two cores (i.e. u-shaped ridges) in the lower half of the image. However, the popular commercial off the-shelf (COTS) tool, namely VeriFinger, is only able to detect single core (shown by red colored rectangle), while the other one remains unnoticed because of poor focus of the image. However, when the inverted fingerprint is provided as input to the same COTS tool, both the cores get detected (highlighted by two red-colored rectangles in the first image in bottom row). Similar observations can be made from the second and third image of Figure, where the cores of fingerprints are detected only when their inverted version is presented as input (see the red-colored rectangles in second and third images of bottom row).

3.2 Grayscale Representation for Colored Contactless Fingerprints

In order to study the effect of different grayscale representation, the unprocessed fingerprint images from PolyU Contactless to Contact-based fingerprint database were selected.These ordinary grayscale images are generally formed by weighted combination of linear intensities observed in individual 6 channels, employing the linear intensities in the red, green and blue channels, respectively, in the colored images.

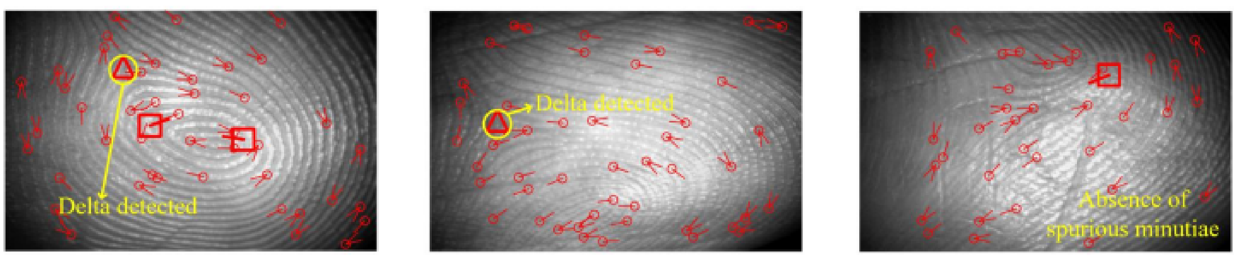
$$I_{Ordinary} = 0.3I_R + 0.59I_G + 0.11I_B$$

Ordinary GrayScale Representation

To reveal additional details from the output images, The Luma grayscale representation is used, as it employs non-linear gamma corrected versions of all channels in colored images, rather than their linear intensities.

$$I_{Luma} = 0.2126\, I'_R \; + \; 0.7152 \;\, I'_G + 0.0722 \;\, I'_B$$

where, IR' , IG' and IB' are the gamma-corrected versions of red, green and blue channels, respectively. It was noted that The Luma representation of colored images enabled larger contrast as compared to the ordinary grayscale images, which facilitates the enhanced detection of minutiae features.



Luma GrayScale Representation

The recognition experiments were also performed on the inverted versions of Luma grayscale images to present a comprehensive evaluation in the currently selected framework.

3.3 Observations and Results

FingerPrints were evaluated using three popular fingerprint matchers, namely NBIS, MCC and COTS on 4 different databases namely PolyU EER, GAR and Multiview EER and GAR.

Databases \ Matchers		NBIS					MCC					COTS				
		Ridge	Valley	Combined	Comparison w.r.t. Ridge	Comparison w.r.t. Valley	Ridge	Valley	Combined	Comparison w.r.t. Ridge	Comparison w.r.t. Valley	Ridge	Valley	Combined	Comparison w.r.t. Ridge	Comparison w.r.t. Valley
PolyU Contactless to Contact-based Database	EER (%)	13.33	14.14	12.50	↑6.22%	↑11.59%	13.25	14.07	11.96	↑9.73%	↑14.99%	0.81	0.29	0.26	↑67.90%	↑10.34%
	GAR (%)	65.52	61.61	69.43	↑5.96%	↑12.69%	58.12	52.52	65.82	↑13.25%	↑25.32%	91.11	98.68	98.91	↑8.56%	↑0.23%
Multiview Contactless Fingerprint Database	EER (%)	16.25	17.71	16.00	↑1.54%	↑9.66%	12.74	12.66	10.91	↑14.36%	↑13.82%	3.60	2.23	1.45	↑59.72%	↑34.97%
	GAR (%)	52.49	53.13	55.35	↑5.44%	↑4.18%	48.82	50.65	60.21	↑23.33%	↑18.87%	90.43	95.52	95.74	↑5.87%	↑0.23%

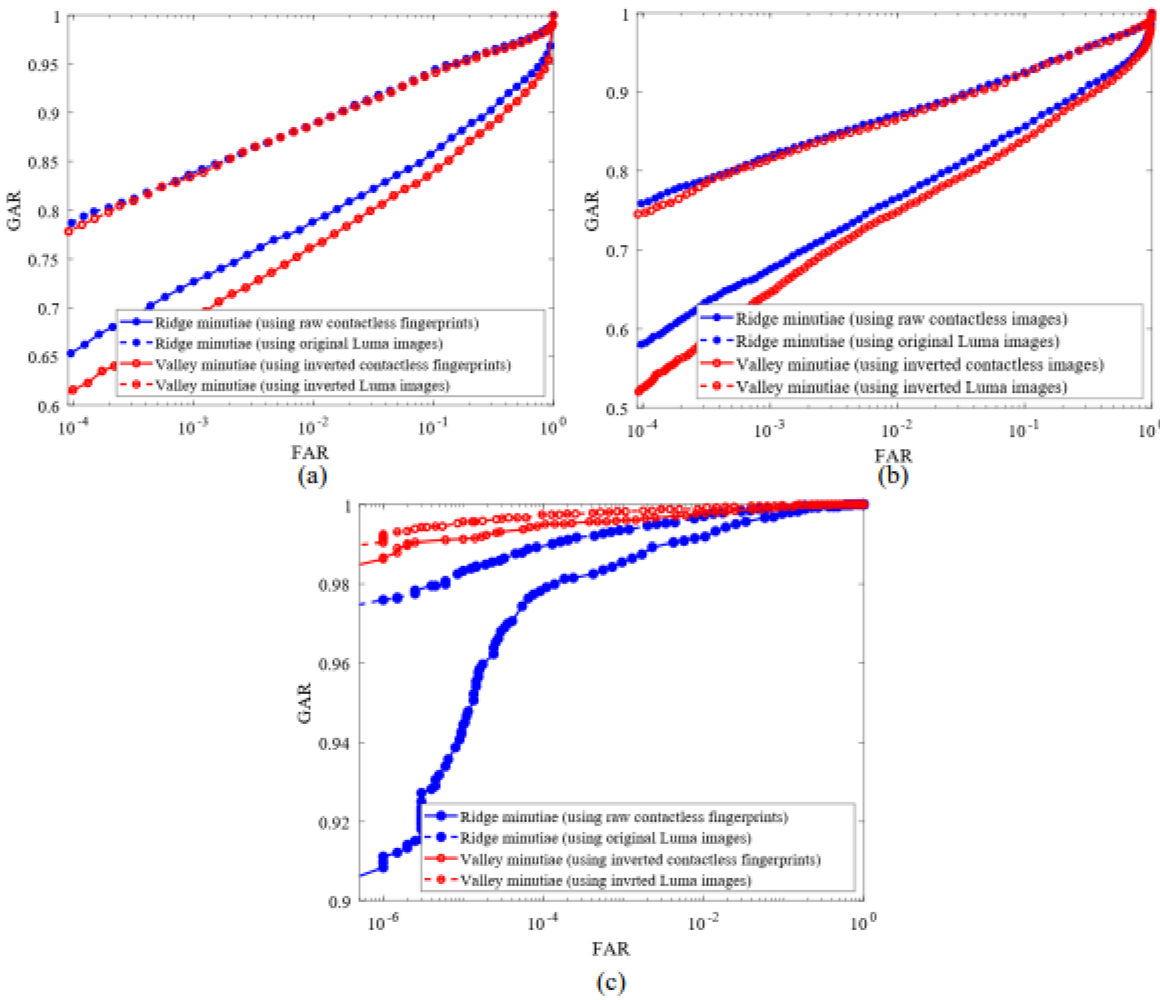


Figure 9: Comparative ROC curves for original and inverted contactless fingerprints using ordinary and Luma grayscale images, (a) using NBIS, (b) using MCC, (c) using COTS matcher.

On one hand, NBIS and MCC yield comparable performance for both ridge (raw fingerprints) and valley (inverted fingerprints) minutiae matching approaches. However, the combination of scores from raw and inverted fingerprints achieves significantly improved performance with both of these matchers as well. Significant performance improvement, in EER and ROC or GAR, due to such a combination strongly validates the proposed arguments. On the other hand, COTS matcher clearly illustrates improved performance with inverted fingerprints alone, which can further be improved through the score combination. This work also considered the effective conversion of contactless color fingerprint representation to its grayscale representation which has received almost nil attention in the literature. In this context, a more diversified grayscale representation, namely Luma grayscale, was introduced with quite encouraging results.

4. Automated Fingerprint Recognition(AFR) with Minutiae Matching

Automated Fingerprint Recognition with Minutiae Matching offers several advantages over manual fingerprint analysis, namely speed, accuracy, consistency, scalability, 24*7 availability, enhanced security, objectivity, cost- effectiveness and consistent record keeping that manual analysis cannot match. It has become an indispensable tool in various fields, including law enforcement, border control, banking, healthcare, and access control, where the rapid and reliable identification of individuals is essential and is further explored in paper by S.M.Mohsen[10].

4.1 The Proposed System

The proposed system is based on three main phases: the image processing phase, the minutiae extraction phase and the matching phase. An image processing technique is used to erase noisy background textures of the fingerprint. The minutiae extraction phase aims to extract fingerprint image characteristics. Finally, the matching phase determines whether the fingerprints are impressions of the same finger. The matching stage also defines a threshold to decide whether a given pair of representations is of the same finger (mated pair) or not. The proposed AFR system has 3 main components.

Step 1: Image Processing

Image processing consists of three stages:

- Image Filtering : In Image Filtering every image is a collection of pixels, where each pixel contains three colors in different ratio, range over 0 to 255 RGB values, which has three parameters, such as RED, GREEN and BLUE as constant figure. For an example, return value as RGB of all 0 contains BLACK and return value as RGB of all 255 contains WHITE. For filtering RGB values of every pixel is replaced by maximum or minimum, with comparing the given threshold.

- Image Enhancement : In Image Enhancement, the black colors mean the ridgelines, which haven’t the same thickness all over the image.

- Image Shaping : In Image Shaping, it requires lining the image which is the initial step to overcome the limitation of smoothness of the image.

Step 2: Minutiae extraction and Template Generation

Minutiae extraction is just a trivial task of extracting singular points in a thinned ridge map.After extracting the minutiae points a predetermined position is clipped to generate a template which bears some sophisticated data to generate a corresponding data file used as template, which file contains the Total number of minutiae points, coordinates of each point. AFIS is a proposed minutiae finding technique used to find out the fingerprint feature named minutiae points from processed images artificially.

Table 1: Classifications of Minutiae point of an image.

Image		1	2	3	4
Minutiae	Contained in image	54	37	32	30
	Selected by AFIS	44	43	38	32
	Dropped by AFIS	22	12	12	9
	False by AFIS	12	18	18	11
	Correct selection	32	25	20	21

Step 3 : Fingerprint Matching

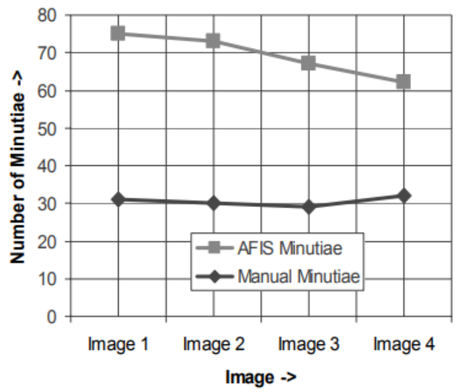
Matching is the final stage of AFR. It comprises of 2 steps:

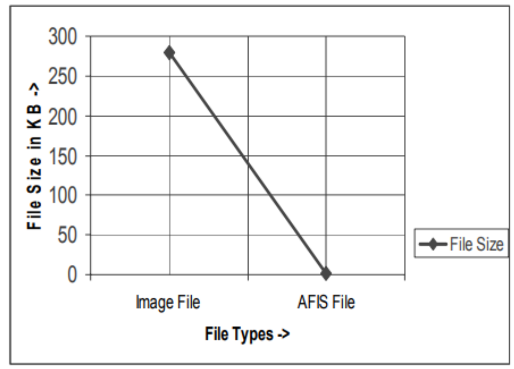
- Registration: In this process, the process takes the one's fingerprint as an image format and processes that image in a few steps such as filtering, enhancing, lining and shaping. Then it requires selection of minutiae points as a feature then generates a template and stores it. The authenticate template contains the total number of minutiae points selected by proposed feature selection technique AFIS. The number of minutiae points limited by a bindings named limited region to improve the flexibility of verification, and then find out the correlation among the features bounded by limited region.

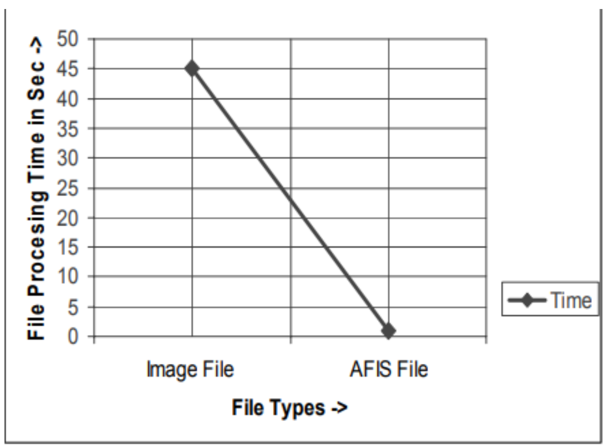
- Verification: To verify one, the process takes one's fingerprint as an image format through fingerprint acquisition hardware and processes that. Then it requires technique to detect minutiae points and selects features, and then to verify, load the templates and compare with the information gathered from verifying one. If it obtains any template matched with that of verifying one, it makes a decision that one was authenticated, or not.

4.2 Experimental Results

A collection of images with manually detected minutiae points contained in the image are used to detect minutiae , dropped minutiae points , False minutiae points , and correct minutiae points acquired by AFIS. Total minutiae points in the image calculated manually are used to find out the accuracy.Dropped minutiae points are those which are present in manual image but not in the image processed by the AFIS, i.e., failed to recognize that point as minutiae points or as feature. False minutiae points are those which are not present in the manual image but the AFIS selects them as minutiae points.







It is clearly visible that Automated System detects much more minutiae points , has lower File Size and Processing Time than a Manual Detection System.The proposed approach thus shows more efficiency in the case of memory and time requirements.

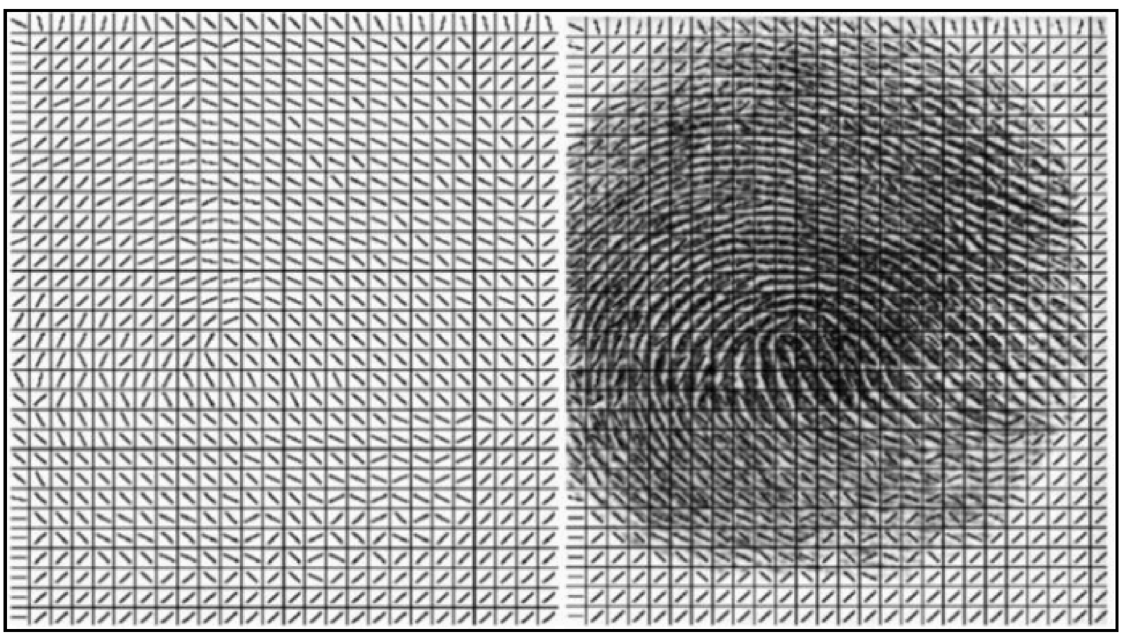
5. Fingerprint Liveness Detection using Convolutional Neural Networks.

Fingerprint recognition and matching is a common form of user authentication. While a fingerprint is unique to each individual, the authentication is vulnerable when an attacker can forge a copy of the fingerprint (spoof). To combat these spoofed fingerprints, spoof detection and liveness detection algorithms are currently being researched as countermeasures to this security vulnerability. An insightful paper by Riley Kiefer[8] introduces a fingerprint anti-spoofing mechanism using machine learning. Specifically, the proposed algorithm builds upon existing methods of local patch-based fingerprint liveness detection by applying a dense and overlapping local patch sampling method and applying algorithm reduction principles to reduce algorithm complexity for embedded platforms. These minutiae-independent local patches are uniformly rotated according to the patch's intensity gradient. Once the preprocessing methods are completed, the patches are passed into a shallow Convolutional Neural Network (CNN) and an aggregate of the patch scores determines the fingerprint classification.

5.1 Proposed Approach

The proposed approach for the fingerprint liveness detection algorithm involves three steps: local patch extraction and preprocessing, local patch classification, and an aggregate fingerprint classification based on the local patch classifications.

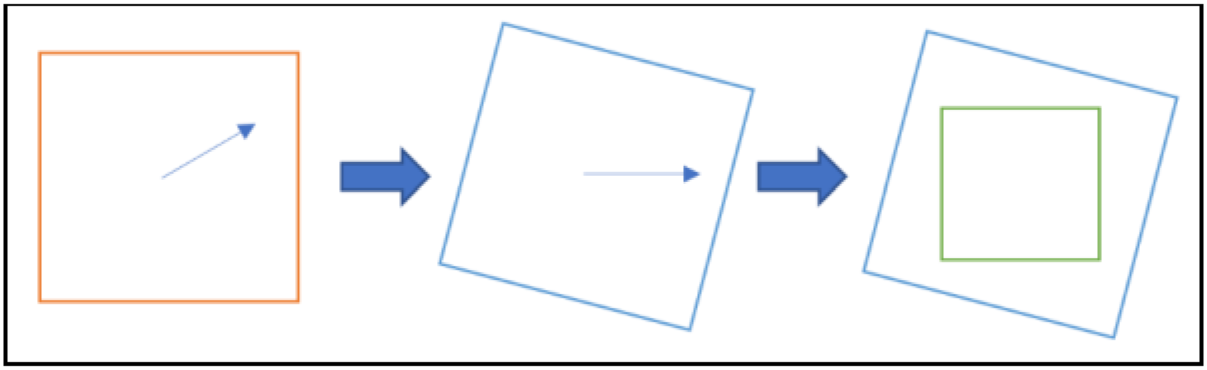
- **Local Vector Orientation Field :** The first step in the proposed algorithm's preprocessing stage is computing a matrix of the 2D local vector orientations based on pixel gradients for all grid cells in an image. A grid cell, τ, is defined as a σ x σ pixel region of the fingerprint image, where σ is equal to 12 pixels. This value was chosen arbitrarily as a base unit of measure. For illustrative purposes, the local vector orientation field would generate vectors for each grid cell of variable angle and magnitude as depicted with lines in Figure Below



  Then the magnitude is computed for every grid cell in the image and saved into a matrix for future access. Each grid cell's unit magnitude is stored at an index in the matrix for each axis x and y.

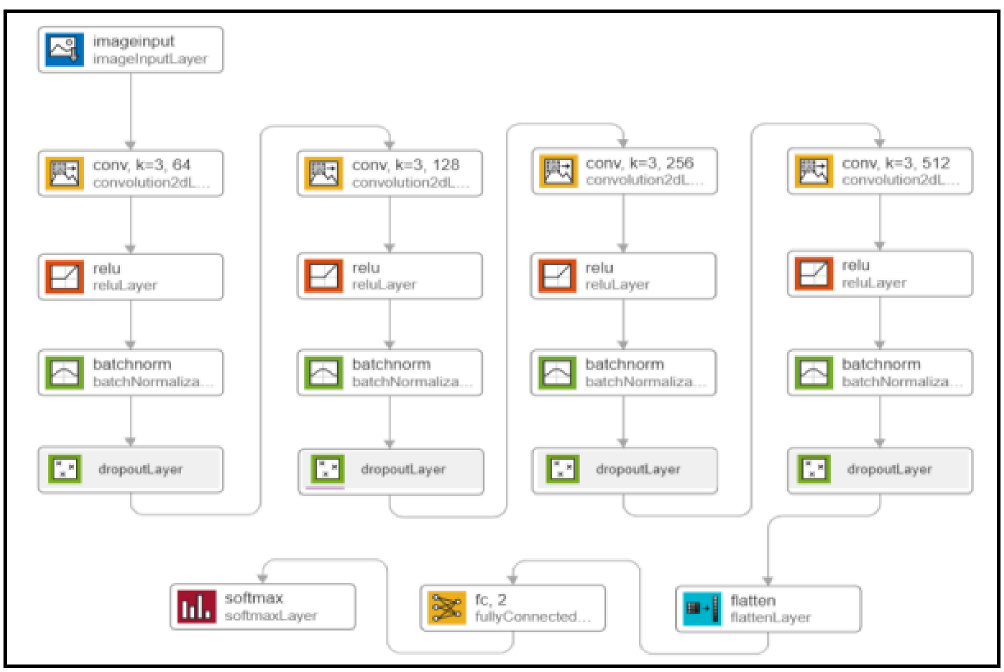
- **Local Patch Extraction:** A local patch is defined as a collection of grids cells . With the grid cell magnitudes computed for all grid cells and saved in a matrix , the local overlapping fingerprint patches can use the grid cell's magnitude to compute the local patch angle. Once the angle of a patch is found, the patch is rotated to 0 degrees (from

the right), cropped to remove any background generated from rotating, and saved to the storage.
This can also be used as a robust minutiae extractor with compact embedding of minutiae features. [3]



5.2 Classification Model

- **Local Patch Classification:** The classification model employs the use of shallow Convolutional Neural Network (CNN) instead of a transfer learning approach from the MobileNet-v1 architecture.



CNN for local patch classification.

The model is optimized using Adam, sparse categorical cross-entropy is used for the loss, and accuracy is the model metric. Model training uses batches of 32 patch images at a time. Several models were tested on different patch sizes using the same model, but larger patch sizes were trained for a longer number of epochs. Since the patches are all preprocessed to the same orientation angle, no transformative cropping or rotation data augmentation techniques are needed to be applied on the training set before model training unlike the traditional Spoof Buster Algorithm. A non-maximum suppression can also be proposed to get precise locations for candidate patches, studied by D.Nguyen in 2017 [3]

- **Aggregate Fingerprint Classification:** The result of the output soft-max layer from the shallow CNN gives a liveness score percentage and a spoof score percentage for each patch. The sum of all live score percentages and spoof score percentages are computed for all generated patches for a given fingerprint image.If a given fingerprint image has a larger global liveness score than global spoof score, the image is classified as real. Otherwise, the image is classified as fake.

5.3 Observations and Results

This proposed method is tested on the public fingerprint liveness detection dataset: LivDet-2009 Biometrika, CrossMatch, and Identix as well as LivDet-2011 Biometrika, Digital, and Sagem. Intra-sensor models are generated, tested, and compared to other top algorithms created by researchers using the Average Classification Error (ACE) metric. A graphic user interface tailored to the proposed method is also presented to visualize the classifier results at the local level.

	Training		Testing		
Dataset	**Live Samples**	**Spoof Samples**	**Live Samples**	**Spoof Samples**	**Spoof Material**
Biometrika	1000	1000	1000	1000	EcoFlex, Gelatin, Latex, Silgum, Wood Glue
Digital	1004	1000	1000	1000	Gelatin, Latex, Playdoh, Silicone, Wood Glue
Italdata	1000	1000	1000	1000	EcoFlex, Gelatin, Latex, Silgum, Wood Glue
Sagem	1008	1007	1000	1036	Gelatin, Latex, Playdoh, Silicone, Wood Glue

Figure 9. LivDet-2011 dataset training and testing details for each scanner. Note that these counts represent the total number of fingerprint images, not the number of patches generated after preprocessing.

Scanner	**FRR**	**FAR**	**ACE**	**Accuracy**
Biometrika	2.30%	4.40%	3.35%	96.04%
Digital	2.30%	1.00%	1.65%	98.35%
Sagem	1.06%	4.50%	2.78%	97.08%

Figure 11. Intra-sensor fingerprint classification performance results after taking the aggregate score of all patches for a given fingerprint for the LivDet-2011 dataset.

Rank	**Reference**	**Algorithm Name**	**ACE (%)**
1	11	DCNN and SVM, RBF Kernel	0
2	12	Local Uniform Comparison Image Descriptor (LUCID)	0.14
3	11	DCNN and SVM, Polynomial Kernel Order 2	0.38
4	11	DCNN and SVM, Polynomial Kernel Order 3	0.57
5	-	**PROPOSED MODEL**	**0.61**

Figure 12. Top 5 intra-sensor models ranked by the ACE metric from the results presented in the survey [1][2] for LivDet-2009 Biometrika. Our proposed model ranks 5th place.

Rank	Reference	Algorithm Name	ACE (%)
1	-	**PROPOSED MODEL**	**0.53**
2	13	CNN-VGG-227	0.6
3	14	MvDA: G5, SID RICLBP LCPD DSIFT	1
4	13	CNN-Alexnet-224x224	1.1
5	15	Deep Triplet Embedding (Tnet)	1.57

Figure 13. Top 5 intra-sensor models ranked by the ACE metric from the results presented in the survey [1][2] for LivDet-2009 CrossMatch. Our proposed model ranks 1st place.

The proposed algorithm LivDet-2009 Biometrika model is placed 5th best overall algorithm and the LivDet-2009 CrossMatch placed 1st overall based on the results published by R. Kiefer et al.

5.4 Optimization of Proposed Algorithm : Feature Extraction using CNN

While the Proposed Algorithm gives good results, it is not completely safe. Recent Studies in 2023 done by Ai Takahashi et.all [2], further enhances the Minutiae Matching algorithm by employing fingerprint feature extraction by combining Texture, Minutiae, and Frequency Spectrum Using Multi-Task CNN.

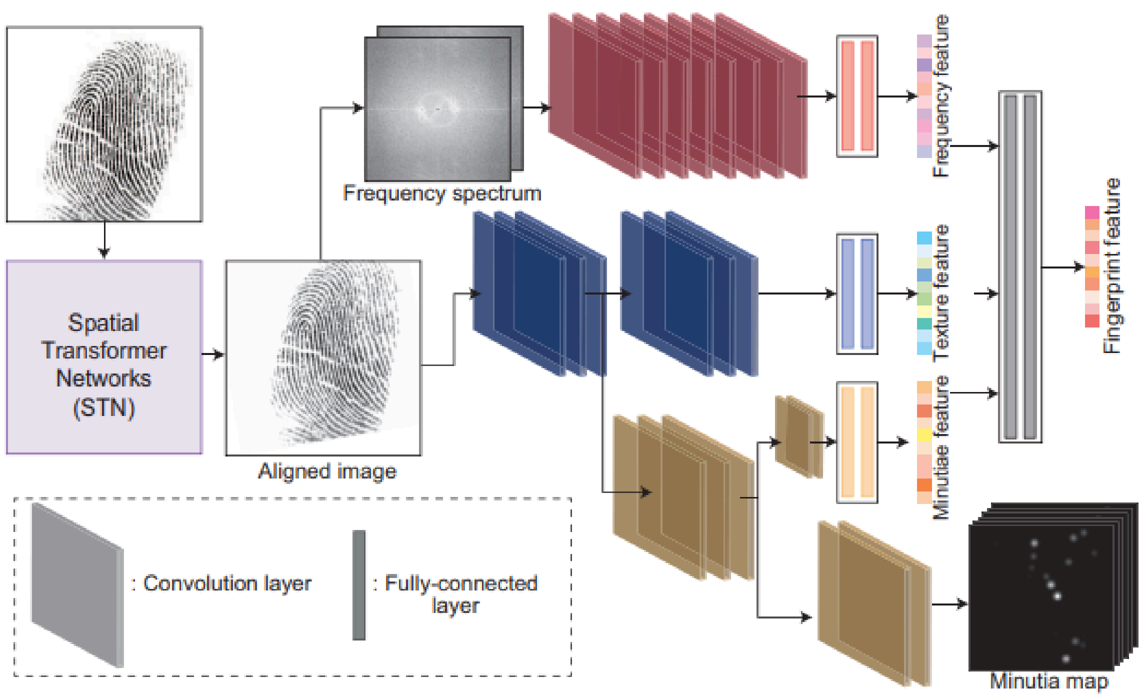


Figure 1. Overview of the network architecture used in the proposed method.

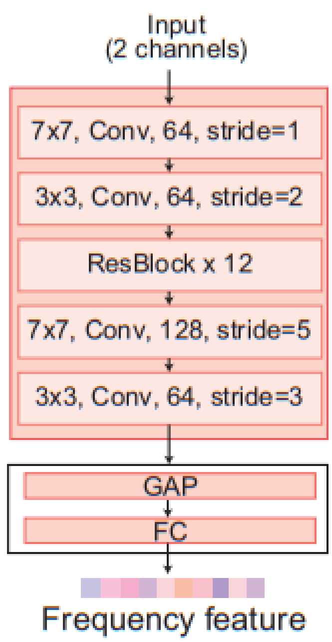
Frequency, texture and minutiae features are extracted from the rotation-corrected fingerprint image using ResNet . The frequency features are extracted using CNN where the input is two channels of the real and imaginary parts of the frequency spectrum obtained by Discrete Fourier Transform (DFT) of the fingerprint image.

In addition, the following preprocessing is applied to extract the frequency features that represent the features of the fingerprint image. The DC component is much larger than the other frequency components and represents the gain inherent to the sensor. In order to reduce the effect of the DC component, we apply normalization so that the average of the pixel values is zero, and then perform DFT.

Since the energy of the fingerprint image is concentrated in an ellipse of the low-frequency band, the high-frequency region contains only perturbations such as noise and aliasing. In order to consider only the inherent frequency band of fingerprints similar to the BLOC , only the region containing the elliptical frequency band is extracted as an input.

The CNN network architecture for extracting texture and minutiae features. Note that the CNNs that extract texture and minutiae features share weights up to the middle. The gray-scale fingerprint image is input to the CNN as one channel. In order to extract minutiae features using CNN, we introduce a minutia map which can represent the positions and angles of minutiae. The minutia map is obtained with a minutiae map generator as shown in , which is branched from a minutiae feature extractor.

Figure 3. Architecture of CNN for extracting frequency-based feature.



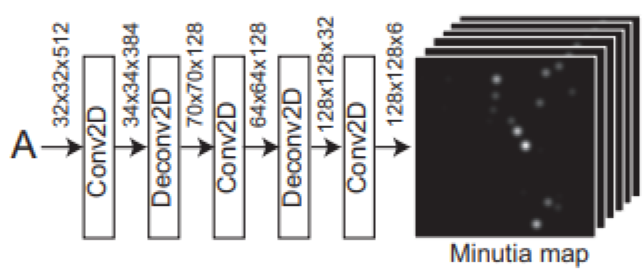
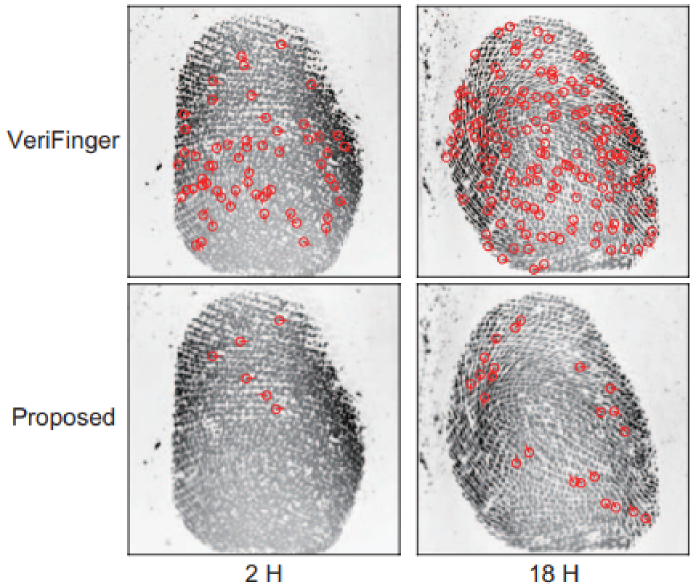


Figure 5. Architecture of minutia map generator.

The proposed CNN model is trained as follows. We use 12,000 fingerprint images (DB1, DB2, and DB3) out of 19,200 fingerprint images in the experiments.

The effectiveness of the proposed method is verified using a saliency map that visualizes the effect of each pixel in the input image on the extracted features.Through a set of experiments using FVC2004 DB1 and DB2, we demonstrated that the matching accuracy is higher than that of VeriFinger and the conventional method.



6. Genetic Programming Framework for Minutiae Matching

Genetic Programming (GP) is a type of evolutionary algorithm and a machine learning technique that is inspired by the process of natural selection. It is used to automatically evolve computer programs to solve problems or optimize tasks. GP is particularly useful when you don't know the exact form of a solution in advance but have a set of building blocks or functions that can be combined to create a solution.Genetic programming (GP) is a computational technique used in minutiae matching for automated fingerprint recognition systems. It is needed in Minutiae Matching for several reasons [15]:

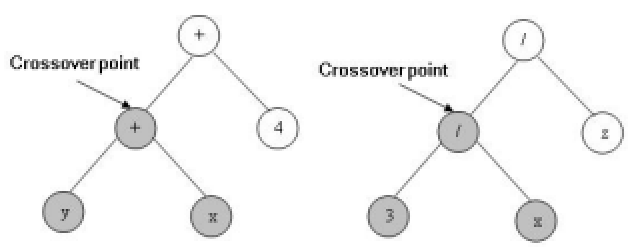
- Complex Search Space: Fingerprint matching involves searching for correspondences between minutiae points (e.g., ridge endings and bifurcations) in two different fingerprints. This creates a complex search space with many possible alignments and correspondences.
- Variability in Fingerprints: Fingerprints can vary significantly in terms of orientation, position, and distortion. This variability makes it challenging to design a fixed algorithm that works well for all fingerprint pairs.
- Optimization: The goal of minutiae matching is to find the best alignment or correspondence between minutiae points while minimizing the dissimilarity (or maximizing the similarity) between the two fingerprints. GP can be used to optimize the matching process by exploring different combinations and alignments of minutiae.

Genetic Programming(GP) in minutiae matching provides a flexible and adaptive approach to designing fingerprint recognition algorithms. It allows the system to evolve and improve its matching capabilities, making it better suited to handle real-world fingerprint data with all its variations and complexities.[15]:

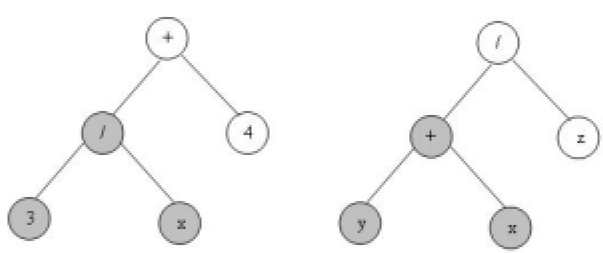
6.1 Proposed Algorithm

Baseline Algorithm for GP is :

- Initialization: Create an initial population of random computer programs. Each program is represented as a tree-like structure, with nodes representing functions or operators and leaves representing operands or variables.Define the function and terminal sets. Functions are operations like addition, subtraction, multiplication, etc., while terminals are variables or constants.Set parameters such as population size, maximum tree depth, and the number of generations.
- Evaluation: Evaluate each program in the population using a fitness function. The fitness function quantifies how well each program solves the problem.Programs that perform better (have higher fitness) are more likely to be selected for reproduction.
- Selection: Use a selection method to choose programs from the current population to create the next generation. Common selection methods include tournament selection, roulette wheel selection, or rank-based selection.
- Recombination (Crossover): Select pairs of parent programs from the current population.Apply crossover operators to exchange subtrees between the parents to create offspring.The crossover point is chosen randomly in the parent programs.

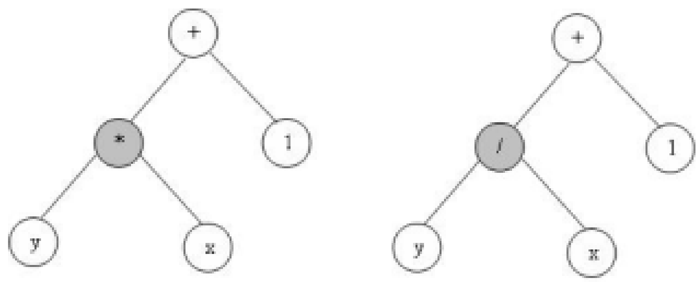


(a) Before Crossover



(b) After Crossover

- Mutation:Randomly select some programs from the population.Apply mutation operators to introduce small random changes to the selected programs. Mutations can involve altering operators, constants, or variables.



(a) Before Mutation (b) After Mutation

- Termination:Repeat the evaluation, selection, crossover, and mutation steps for a fixed number of generations or until a termination condition is met.Termination conditions might include reaching a fitness threshold, a maximum number of generations, or a runtime limit.
- Result: Once the algorithm terminates, the best-evolved program represents the solution to the problem or the optimized task.

A study published in International Journal of Computer Science by Ismail A.[7] et.all has used genetic programming with minutiae points to devise the mathematical equation that defines fingerprints and can be used in matching between fingerprints. These equations are obtained by enhancing and filtering the fingerprint after extracting the minutiae points in the fingerprint using crossing number (CN) method.

- To match a query fingerprint to a matching fingerprint, we extract the minutiae points for the two fingerprints first from enhanced images.They used the crossing number (CN) method to perform the minutiae points extraction. This method extracts the ridge endings and bifurcation from the skeleton image by examining the local neighborhood of ridge pixel using a (3*3) window. After using the CN method they extracted all data for minutiae points (end points and bifurcation points) as summarized in table I and II.

TABLE I: End Points Data

x	angle	y
147	-1.05	48
40	1.57	101
133	-1.57	111
50	1.57	112
63	1.57	115
49	-2.09	117
67	2.36	124
119	-2.09	127
88	1.05	143
126	1.05	154

Ending points were determined by three variables x, y, and angle, but bifurcation points were determined by five variables x, y, angle 1, angle 2, and angle 3.

TABLE II: Bifurcation points Data

x	angle1	angle2	angle3	y
109	3.14	-1.05	.52	50
74	2.09	-2.09	.52	55
98	-2.62	1.05	-.52	60
100	3.14	1.57	-1.05	82
107	-2.36	1.57	-.79	89
39	-2.36	1.05	-.79	94
45	-2.09	1.57	0	101
92	-2.36	1.05	-.79	103
39	3.14	-1.57	1.05	110
71	-2.36	-1.05	.79	119
64	3.14	1.05	-1.05	120
128	-2.36	1.05	-1.05	135
92	-2.36	1.05	-1.05	138
48	2.09	-2.09	0	152
59	-2.36	1.57	0	179

- All the parameters are manually fed as input to the GP Algorithm and the fitness function, to train the CNN and get the final equation for finding Minutiae Points for any parameters.

TABLE III: Genetic programming parameters

Maximum number of Generations	1700
Size of Population	2500
Maximum depth of new individuals	6
Maximum depth of new sub trees for mutants	4
Maximum depth of individuals after crossover	18
Fitness-proportionate reproduction fraction	0.1
Crossover at any point fraction	0.2
Crossover at function points fraction	0.2
Number of fitness cases (end points)	10
Number of fitness cases (bifurcation points)	15
Selection method	fitness- proportionate with- over-selection
Generation method	Full

Two equations are generated for the Minutiae Points

```
(+ (- (% (+ (% (% -8 X) (*(% (+ (% ANGLE X) -8) -2) ANGLE))
      (% (% ANGLE -1) (% (+ -9 ANGLE) (- 3 -2))))
    (- (+ (% (+ (+ 7 ANGLE) -2) X) (- -5 2))
      (- (% -5 (* X -4)) (+ ANGLE 9))))
  (- (*(% (- X ANGLE) (+ X - 1))
      (- (% (+ (+ X (+ -2 (- (% X ANGLE) ANGLE))) X) -6)
        (*(% (- X ANGLE) (+ (* ANGLE X) -1))
          (- (% (% (+ ANGLE X)
                (*(% (*(*10 (+ ANGLE (+ 1 -3)))
                    (% (*ANGLE ANGLE) 2))
                  (+ (+ 9 -3)
                    (% (- (+ (*(+ X X) X))))))
                        .
                        .
                        .
```

(a) End Points formula

```
(+ (% (- (+ (% (*10 (% (% (* -9 ANGLE2) ANGLE3) -2)) (% ANGLE3 -1))
     (*(*(- ANGLE3 3) (+ ANGLE2 ANGLE1))
       (+ (- -7 ANGLE3) (- ANGLE2 3))))
   (+ (- (- 10 -4) (*-7 ANGLE3))
     (- 9
       (% (*(*6 -8) ANGLE3)
         (% (- (+ (% (*10 (% (% (*-9 ANGLE2) ANGLE3) -2))
                 (% ANGLE3 -1))
               (*(*(- ANGLE3 3) (+ ANGLE2 ANGLE1))
                 (+ (- ANGLE2 3) (- ANGLE2 3))))
             (+ (- (- 10 -4) (*-7 ANGLE3))
               (% (- (% (* ANGLE3 -3)
                     (- (+ (% ANGLE1
                           (+ (- (- (+ ANGLE3
                                 (% (- (% (- (+ (*(+ (% (+ (% X
                                                    3)
                                                  (% (% (% (- -5
                                                        X)
                                                      (% ANGLE2
                                                        6))
                                                    (- 2
                                                      (- ANGLE1
                                                        -5)))
                                                  3))
                                                -3)
                                              2))))))))
                              .
                              .
                              .
```

(b) Bifurcation points formula

All the query minutiae points are now applied to these formulas obtained from GP and if the output of both Ends Point Formula and Bifurcation Points Formula matches the target minutiae points, then it's a match.

6.2 Results and Observations

Three different images were used with their end points and bifurcation points, as an input to the GP.

(a) image1

(b) image2

(c) image3

TABLE IV: End points of three fingerprints.

image1		image2		image3	
x	angle	x	angle	x	angle
86	-2.62	147	-1.05	104	3.14
158	-.52	40	1.57	98	0
156	-.52	133	-1.57	121	2.09
93	.52	50	1.57	65	2.36
111	.79	63	1.57	130	-2.09
24	-2.36	49	-2.09	133	1.57
112	.52	67	2.36	88	-2.09
161	-2.09	119	-2.09	107	1.05
103	.52	88	1.05	128	-1.57
151	.52	126	1.05	144	-1.57
				69	1.57
				99	-2.62
				73	-2.09
				62	0.79
				120	2.62

TABLE VI: Bifurcation points of image 1

x	angel1	angel2	angel3
109	3.14	.79	-.52
96	2.62	-2.09	0
149	2.62	-1.57	0
110	2.62	-2.09	0
122	2.62	-1.57	0
80	3.14	-1.57	1.05
116	2.36	-2.62	-.79
171	2.36	-2.36	-.79
154	-2.62	1.57	-1.05
167	-2.62	1.05	-.52

TABLE VII: Bifurcation points of image 2

x	angel1	angel2	angel3
109	3.14	-1.05	.52
74	2.09	-2.09	.52
98	-2.62	1.05	-.52
100	3.14	1.57	-1.05
107	-2.36	1.57	-.79
39	-2.36	1.05	-.79
45	-2.09	1.57	0
92	-2.36	1.05	-.79
39	3.14	-1.57	1.05
71	-2.36	-1.05	.79
64	3.14	1.05	-1.05
128	-2.36	1.05	-1.05
92	-2.36	1.05	-1.05
48	2.09	-2.09	0
59	-2.36	1.57	0

TABLE VIII: Bifurcation points of image 3
X Angle 1 Angle 2 Angle 3

x	angel1	angel2	angel3
52	2.09	-2.09	.79
88	-2.62	1.05	0
132	-2.36	2.09	-1.05
75	-2.62	1.05	-1.05
106	-2.36	1.57	-.79
123	2.09	-1.57	1.05
115	3.14	1.05	-.79
108	-2.62	-1.05	.79
66	-2.62	1.05	-1.05
62	-2.62	-1.05	.79
137	2.36	.79	-.79
77	2.09	-2.09	0
75	-2.09	1.57	0
78	-2.62	-1.05	.79

After Evaluations, the final results were obtained as below-

TABLE V: The results of evaluating them in the mathematical formula of the query fingerprint (end points formula).

image1	image2	image3	query image
111.62	48.00	150.47	48
96.73	101.00	224.74	101
96.62	111.00	149.19	111
160.05	112.00	122.94	112
153.90	115.00	129.03	115
98.99	117.00	150.43	117
172.66	124.00	121.71	124
134.89	127.00	152.10	127
166.66	143.00	109.83	143
199.05	154.00	112.52	154
		117.86	
		114.43	
		119.19	
		125.93	
		129.56	

TABLE IX: The results of evaluating the bifurcation formula of the query fingerprint on the 3 fingerprints.

image 1	image 2	image 3	query image
97.06	50.00	70.79	50
157.71	55.00	178.81	55
153.76	60.00	-341.05	60
155.82	82.00	182.66	82
156.58	89.00	89.06	89
134.07	94.00	-81.82	94
325.16	101.00	98.94	101
314.5	103.00	115.07	103
227.62	110.00	177.93	110
65.89	119.00	115.95	119
	120.00	115.19	120
	135.00	149.14	135
	138.00	133.63	138
	152.00	115.70	152
	179.00		179

From the above results, it is concluded that the number of bifurcation points is ten points in Image 1, fifteen bifurcation points in Image 2, and fourteen bifurcation points in Image 3. Using the mean square error method,it was found that image 2 is a match to the query image in bifurcation points, whereas Image 1 and Image 3 are not a match. Finally, Fingerprint Image 2 is matched with the query fingerprint Image and fingerprints Image 1 and Image 3 are not.
In future work Genetic Programming can be used to classify fingerprints to decrease the time Complexity and Space Complexity of other Algorithms.

7. Conclusion

In this survey, We have explored the extensive body of research surrounding Minutiae Matching for Fingerprint Recognition. The field of fingerprint recognition has witnessed significant advancements over the years, and minutiae-based methods remain a cornerstone in this domain.
We began by delving into understanding what exactly is Minutiae Matching,the various types of matching techniques, including point pattern matching, ridge-based matching, and orientation-based matching. These methods, with their unique strengths and weaknesses, continue to play a vital role in enhancing fingerprint recognition systems. We then compared Contact- Based Matching vs Contactless matching and dived into active research on improving the accuracy of Contactless matching.
Furthermore, we examined the incorporation of Convolutional Neural Networks (CNNs) into the minutiae matching process. The utilization of deep learning techniques has opened new avenues for feature extraction and matching, resulting in improved accuracy and robustness in challenging scenarios.
One of the noteworthy highlights of this survey is the discussion on Genetic Programming (GP) applied to Minutiae Matching. GP has proven to be a promising approach in automating the design of minutiae matching algorithms, reducing the need for manual feature engineering, and improving adaptability to different fingerprint datasets.

7.1 Challenges in Minutiae Matching Literature

While minutiae matching is a fundamental technique in fingerprint recognition, the research literature does have some weaknesses and challenges. Here are some common weaknesses identified in the minutiae matching research literature:

- Sensitivity to Noise: Minutiae-based methods can be highly sensitive to noise and variations in fingerprint images. This sensitivity can lead to false positives and false negatives, particularly in low-quality or distorted fingerprint images.
- Limited Robustness: Minutiae matching algorithms may struggle with variations in fingerprint impressions due to factors like pressure, moisture, and skin conditions. These variations can affect the consistency and accuracy of matching.
- Dependency on Feature Extraction: Traditional minutiae matching heavily relies on manual feature extraction, which can be time-consuming and may not capture all relevant information in a fingerprint. Feature extraction methods can vary, impacting the performance of the matching algorithm.
- Scalability Issues: Some minutiae matching algorithms may not scale well to handle large databases of fingerprint templates. As the size of the database increases, the computational and memory requirements can become a significant challenge.
- Lack of Standardization: There is a lack of standardized datasets and evaluation metrics in the minutiae matching literature. This can make it difficult to compare the performance of different algorithms and draw meaningful conclusions.
- Vulnerability to Spoofing: Minutiae-based methods may be vulnerable to spoofing attacks, where fake fingerprints or replicas are used to gain unauthorized access. Enhancing the robustness of minutiae matching against such attacks is an ongoing challenge.
- Limited Adaptability: Minutiae matching algorithms may not be easily adaptable to different fingerprint sensor technologies, which vary in terms of resolution, image quality, and sensor type. This limits their applicability across diverse devices.
- Real-Time Processing: Achieving real-time performance for minutiae matching in large-scale systems remains a challenge. As the demand for quick and efficient recognition systems grows, there is a need for faster matching algorithms.
- Privacy Concerns: Storing and matching minutiae templates can raise privacy concerns, as they can potentially be reverse-engineered to reconstruct a fingerprint image. Protecting the privacy of biometric data is an ongoing concern.
- Ethical and Legal Issues: The use of minutiae matching in law enforcement and security applications raises ethical and legal questions regarding privacy, consent, and potential misuse of fingerprint data.

7.2 Potential Future Research in Minutiae Matching

Future research in minutiae matching for fingerprint recognition could focus on some of the following paradigms:

- Enhanced Feature Extraction: Develop more robust and automated methods for extracting fingerprint features, reducing sensitivity to noise.

- Noise Handling: Research techniques to effectively handle noisy and low-quality fingerprint images.
- Adversarial Attacks: Improve resistance to spoofing attacks through anti-spoofing mechanisms.
- Scalability: Develop algorithms for efficient matching in large fingerprint databases.
- Cross-Device Compatibility: Ensure compatibility across various fingerprint sensor technologies.
- Real-Time Processing: Enhance speed and efficiency for real-time applications.
- Template Protection: Securely store and transmit fingerprint templates while protecting privacy.
- Multimodal Biometrics: Combine fingerprint matching with other biometric modalities.
- Privacy-Preserving Matching: Explore privacy-preserving techniques for secure matching.
- Ethical and Legal Considerations: Address ethical and legal aspects of fingerprint matching.
- Standardization: Promote standardized datasets and evaluation metrics.
- Human-Machine Collaboration: Leverage human expertise in minutiae matching tasks.
- Explainable AI: Develop interpretable algorithms for user trust.
- Cross-Domain Applications: Adapt minutiae matching to other fields like medical imaging and forensics.

In closing, this survey has provided a comprehensive overview of Minutiae Matching for Fingerprint Recognition, highlighting its evolution, current trends,limitations and potential directions for future research.

REFERENCES:

1. Aman Attrish, Nagasai Bharat, Vijay Anand, Member, IEEE, and Vivek Kanhangad, Senior Member, IEEE. A Contactless Fingerprint Recognition
[PDF] A Contactless Fingerprint Recognition System | Semantic Scholar
2. Ai Takahashi , Yoshinori Koda , Koichi Ito , and Takafumi Aoki. Fingerprint Feature Extraction by Combining Texture, Minutiae, and Frequency Spectrum Using Multi-Task CNN.
[2008.11917] Fingerprint Feature Extraction by Combining Texture, Minutiae, and Frequency Spectrum Using Multi-Task CNN (arxiv.org)
3. Dinh-Luan Nguyen, Kai Cao and Anil K. Jain. Robust Minutiae Extractor: Integrating Deep Networks and Fingerprint Domain Knowledge.
https://arxiv.org/abs/1712.09401
4. Eppstein D, Goodrich MT, Jorgensen J, Torres MR. Geometric fingerprint recognition via oriented point-set pattern matching. arXiv.org. 2018.
https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/geometric-fingerprint-recognition-via-oriented/docview/2092756459/se-2.
5. Grosz SA, Engelsma JJ, Liu E, Jain AK. C2CL: Contact to contactless fingerprint matching.
https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/c2cl-contact-conta ctless-fingerprint-matching/docview/2509915118/se-2.
6. Gwang-Il Ri, , Chol-Gyun Ri and Su-Rim Ji. A Fingerprint Indexing Method Based on Minutia Descriptor and Clustering.
[1811.08645] A Fingerprint Indexing Method Based on Minutia Descriptor and Clustering (arxiv.org)
7. Ismail A. Ismail, Nabawia A. ElRamly , Mohammed A. Abdelwahid , Passent M. ElKafrawy and Mohammed M. Nasef. Genetic Programming Framework for Fingerprint Matching.
[0912.1017] Genetic Programming Framework for Fingerprint Matching (arxiv.org)
8. Kiefer R, Stevens J, Patel A. Fingerprint liveness detection using minutiae-independent dense sampling of local patches..
https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/fingerprint-liveness-detection-using-minutiae/docview/2799917745/se-2.
9. Kim J. A method of data augmentation to train a small area fingerprint recognition deep neural network with a normal fingerprint database. arXiv.org. 2022.
https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/method-data-augmentation-train-small-area/docview/2642601079/se-2.
10. Mohsen SM, Zamshed Farhan ,S.M., Hashem MMA. Automatic fingerprint recognition using minutiae matching technique for the large fingerprint database. arXiv.org. 2013.
https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/automatic-fingerpr int-recognition-using-minutiae/docview/2084939990/se-2.
11. Padkan N, B SB, Faraji MR. Fingerprint matching using the onion peeling approach and turning function.

https://libezproxy.syr.edu/login?url=https://www.proquest.com/working-papers/fingerprint-matchi ng-using-onion-peeling-approach/docview/2579215567/se-2

12. RAVI. J , K. B. RAJA, VENUGOPAL. K. R. FINGERPRINT RECOGNITION USING MINUTIA SCORE MATCHING. https://arxiv.org/abs/1001.4186.
13. Ritesh Vyas, Ajay Kumar. A Collaborative Approach using Ridge-Valley Minutiae for More Accurate Contactless Fingerprint Identification. [1909.06045] A Collaborative Approach using Ridge-Valley Minutiae for More Accurate Contactless Fingerprint Identification (arxiv.org)
14. Wajih Ullah Baig, Umar Munir, Waqas Ellahi, Adeel Ejaz, Kashif Sardar. Minutia Texture Cylinder Codes for fingerprint matching. [1807.02251] Minutia Texture Cylinder Codes for fingerprint matching (arxiv.org)
15. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition. Handbook of Fingerprint Recognition | SpringerLink