A Mathematical Toolkit to improve the Similarity Analysis for Fingerprint Images

1st Elioenai M. F. Diniz
PPGEEC
UTFPR
Pato Branco, Brazil
elioenai@alunos.utfpr.edu.br

2rd João V. C. Mazzochin
PPGEPS
UTFPR
Pato Branco, Brazil
joaomazzochin@alunos.utfpr.edu.br

3th Luiz A. Zanlorensi
Research Dept
Infant.ID
Curitiba, Brazil
luiz@natosafe.com

4th Wesley A. C. de Bona
PPGEEC
UTFPR
Pato Branco, Brazil
wesleybona@alunos.utfpr.edu.br

5nd Luiz F. P. Southier
COENS
UTFPR
Dois Vizinhos, Brazil
luizsouthier@utfpr.edu.br

6th Fábio Favarim
DAINF
UTFPR
Pato Branco, Brazil
favarim@utfpr.edu.br

7th Dalcimar Casanova
PPGEEC
UTFPR
Pato Branco, Brazil
dalcimar@utfpr.edu.br

***Abstract*—While fingerprint biometrics are a mature technology, preprocessing steps such as image alignment and minutiae filtering still significantly impact their performance. To address these challenges, this paper introduces a comprehensive package of analytical methods designed to enhance fingerprint image similarity analysis. Initially, we present an alignment method based on partitioning and statistical image analysis, which does not rely on specific information, such as the location of singular points. Next, a minutiae filtering method is developed, focusing on detecting and removing edge minutiae, which tend to provide non-discriminative information. To establish an agnostic distance metric approach, we conducted extensive experiments on public fingerprint datasets, incorporating an ablation study to evaluate the impact of alignment and filtering methods. Our results demonstrate that the proposed combination of alignment and filtering significantly outperforms untreated images, providing an efficient and competitive solution that improves and contributes to the biometric analysis field.**

***Index Terms*—fingerprint, similarity, analytical, comparative, outperforming**

I. Introduction

Biometric identification, particularly fingerprint verification, plays a fundamental role in various applications, such as security, forensic science, and access control. When analyzing fingerprints, the main factor is the minutiae, which are points where the ridges terminate or bifurcate. Some processing steps may be necessary to improve the performance of these models [1]. Among these steps are standardized alignment of the image and the removal of spurious minutiae. An example of these processes is illustrated in Figure 1.

Several approaches have been proposed to address the alignment and filtering of minutiae in fingerprint images. These solutions involve image processing techniques, such as orientation fields, and machine learning methods, such as Convolutional Neural Networks (CNN). Although these solutions have shown promising results, they still present limitations regarding computational complexity, dependence on the choice of free parameters that influence the final results, and dependence on specific information, such as the location of singular points.

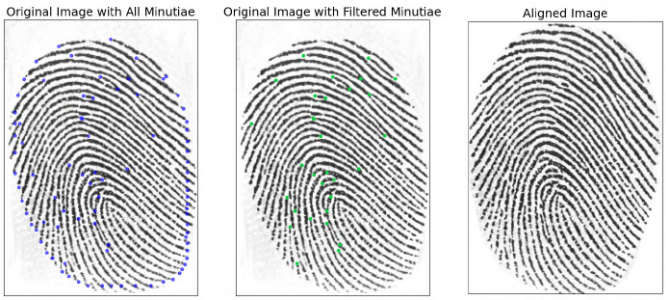

Fig. 1. Original image with minutiae, image with filtered minutiae and aligned image

The main gap identified in this context is the need to develop efficient minutiae alignment and filtering methods for metrics that evaluate the similarity, are robust concerning parameter instabilities, and are less dependent on specific image information. Many existing solutions still face challenges when applied in environments without a labeled dataset or under critical time constraints.

This work proposes a package of improvements that utilizes only pure mathematical calculations to enhance the quality of the final minutiae map and increase its discriminative power. Furthermore, it is sought to make it robust in terms of computational cost, enabling it to be used in scenarios such as embedded systems, and efficient in terms of measured results employed in the analysis step concerning the original image without any processing, thus becoming to become a viable and competitive solution.

This study aims to develop and evaluate a minutiae alignment and filtering method that is efficient in terms of improved

CAPES (Finance Code 001), CNPq and FINEP

similarity analysis and more stable concerning parameters. The methodological approach involves proposing a partition-based alignment method, a statistical image analysis, and a minutiae filtering method focused on detecting and removing border minutiae that can be considered essentially spurious. These methods are then evaluated using public fingerprint datasets, with performance comparison based on relation to different distance calculations to measure similarity.

II. Related Works

Several approaches have been proposed for the task of position estimation in fingerprints. The majority of approaches make use of the orientation field. In [4], a method using dictionaries for patching is proposed. The algorithm uses the Hough transform to align the fingerprint in a universal coordinate system. This method enables robust orientation field estimation, even for low-quality or incomplete latent fingerprints. In [3], a fingerprint alignment method is proposed based on three stages: extraction of the longitudinal axis from the orientation field, mathematical removal of the influence of singular points, and applying a global circular model to define the horizontal axis. The method requires localizing all singular points and has limitations with non-linear distortions. However, in [2], an unsupervised deep learning-based alignment method is proposed, directly applied to images. Their method uses CNNs to estimate fingerprint rotation. The approach is trained unsupervised, using pairs of randomly rotated images of the same fingerprint. The CNN learns to estimate the rotational difference between the two images, thus allowing alignment without the need for manually labeled data.

Concerning the filtering of minutiae, in [5], a method is proposed to eliminate false minutiae matches during fingerprint verification using Support Vector Machines (SVM). Their system employs an SVM classifier trained with pairs of corresponding minutiae from known fingerprints, both from the same user and from different users. This work demonstrates the effectiveness of machine learning methods in identifying and eliminating false or spurious minutiae, contributing to improved accuracy in fingerprint verification systems. A hybrid algorithm for removing false minutiae, aimed at improving the accuracy of low-quality fingerprint matching, is presented in [6]. The method combines two approaches: edge minutiae elimination and false minutiae removal based on statistical characteristics. The authors report that their method outperforms previous techniques based on spatial filtering, showing greater efficiency and precision in eliminating false minutiae. In [7], a set of techniques is introduced to enhance the quality of fingerprint images and more accurately extract minutiae. They used the Crossing Number method [8] to identify terminations and bifurcations for minutiae extraction. The authors also propose rules based on the distance between minutiae to remove false detection, such as removing minutiae that are too close to each other.

III. Proposed Method

This work presents a set of improvements for biometric recognition, including a method for fingerprint alignment and a method for filtering spurious edge minutiae. The alignment method directly uses the image as input, as described in [2], and exploits the geometric properties of images, similar to [3]. The proposed approach employs partitioning and statistical analysis, while other methods range from image processing techniques to deep learning. Although many methods focus on specific features, such as singularities or contours, this one analyzes the pixel distribution. This method does not rely on detecting specific points, which potentially makes it faster than deep learning-based methods. However, its effectiveness may vary depending on the shape and symmetry of the region of interest, which is determined by an external method that generates the corresponding mask.

The minutiae filtering method focuses on detecting edge minutiae, similar to [6]. Unlike other methods that employ more sophisticated and computationally expensive approaches, the approach uses a simple geometric technique. While other methodologies consider additional information, this method focuses exclusively on minutiae positioning. This approach is highly simple, easy to implement, and has a very low computational cost. However, it may be less accurate in identifying false minutiae that are not on the edges, and it may also remove genuine minutiae that are close to the edges.

A. Alignment of Fingerprint

Within the context of the proposed approach for alignment, the first step involves detecting the Region of Interest (ROI) [10]. This step is illustrated in Figure 2, where a proprietary technology was employed to detect the fingerprint [11]. Additionally, the images are reduced to 20% of their size, or until they approach a size of 200 pixels in one dimension. This is done to accelerate the next steps.

Next, we create one ROI image version for each angle within the interval $[+45, -45]$. The following steps are performed to identify which of the ROI image versions is aligned, i.e., was rotated to the best angle. For each ROI image version, we find the four ROI boundaries, shown in blue in Figure 2. Then, the region within the four boundaries is sliced into four quadrants (magenta lines in the Figure 2) analogous to the Cartesian representation in two-dimensional space. The four quadrants are identified by numbers 1 to 4 in the Figure 2. The slicing result is shown in Figure 2 (right).

After slicing, the sum of the pixels in each quadrant is calculated. The reason for this is that white pixels are numerically represented by 255, while black pixels are represented by 0; thus, the summation indirectly accounts for the proportion of the area in each quadrant. As a result, four variables are created for the quadrants $S_1(\theta)$, $S_2(\theta)$, $S_3(\theta)$, and $S_4(\theta)$, each representing the sum in each quadrant.

An aligned version of the ROI image would have the four variables with closer values since the proportion of the aligned image in each quadrant tends to be similar.

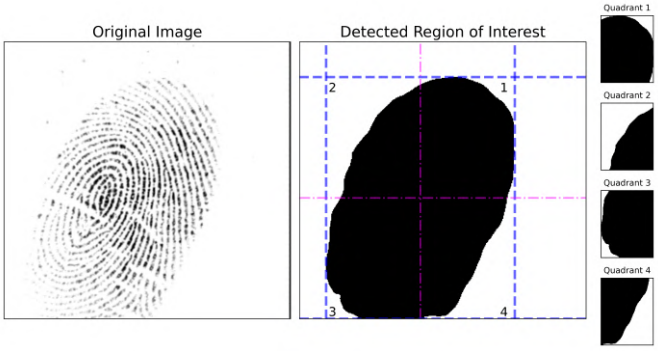

Fig. 2. Example of an original fingerprint image and its detected region of interest

Subsequently, the ratios between the four variables are calculated to incorporate the horizontal and vertical symmetric relation between quadrants, i.e., the vertical ratio between quadrants 2 and 3, and quadrants 1 and 4, and the horizontal ratio between quadrants 1 and 2, and quadrants 4 and 3, as shown in the four equations below.

$$r_1(\theta) = \frac{S_1(\theta)}{S_2(\theta)} \quad (1) \qquad r_2(\theta) = \frac{S_2(\theta)}{S_3(\theta)} \quad (2)$$

$$r_3(\theta) = \frac{S_4(\theta)}{S_3(\theta)} \quad (3) \qquad r_4(\theta) = \frac{S_1(\theta)}{S_4(\theta)} \quad (4)$$

To find the optimal angle θ, the value that minimizes the standard deviation between the ratios is chosen, as shown in Equation 7.

$$\bar{r}(\theta) = \frac{r_1(\theta) + r_2(\theta) + r_3(\theta) + r_4(\theta)}{4} \quad (5)$$

$$\sigma(\theta) = \sqrt{\frac{1}{4}\sum_{i=1}^{4}\left(r_i(\theta) - \bar{r}(\theta)\right)^2} \quad (6)$$

$$\theta^* = \arg\min_{\theta} \sigma(\theta) \quad (7)$$

An example of the result of this method can be seen in Figure 3, where three input images are shown along with their respective aligned versions.

B. Filtering of Minutiae

The proposed method aims to determine which minutiae are border minutiae and which are internal. The reason for this is that minutiae located on the edges of the image are only the endpoint of the contact area between the finger and the sensor surface. Therefore, they do not tend to add discriminative information about a specific finger.

The proposed method uses only the spatial information of the minutiae to perform the classification between each

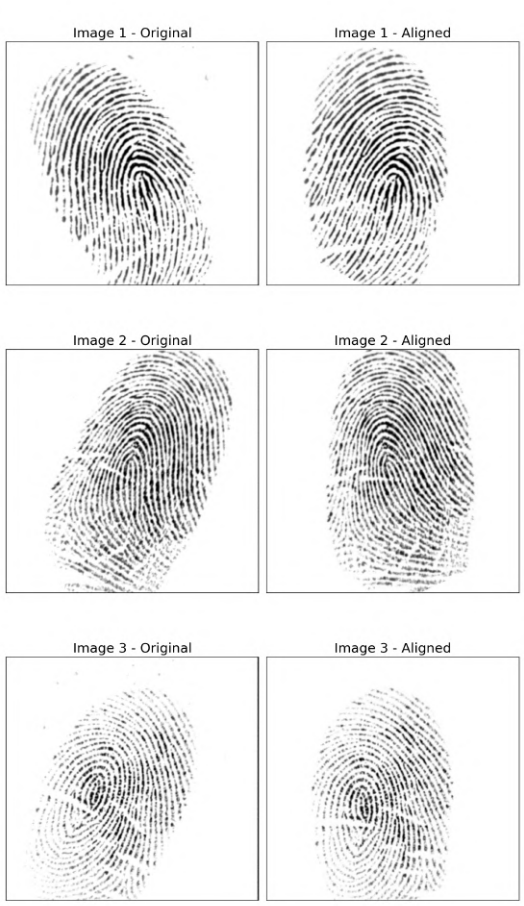

Fig. 3. Three examples of alignment of fingerprints method

minutia. The first step of the method occurs precisely in the detection of minutiae, after the skeletonization of the segmented image using the crossing number algorithm [8].

With the detected points, the x and y coordinates and an angle θ are used for each minutiae point [12], where the proposed method utilizes only the coordinate values. Analogous to a compass rose, the minutiae corresponding to the cardinal and intercardinal directions are identified. These eight specific points are then connected to create a polygon, an octagon in this case. To ensure that the minutiae on the boundary of the polygon are detected, the Euclidean norm is used to calculate the distance between each minutiae and the polygon's edges. If this distance is smaller than a very small margin value ϵ, which can be trivially defined as $1e-10$, the minutiae are considered external. Subsequently, a Boolean logic operation reclassifies border minutiae as external, ensuring that only those strictly within the polygon are considered, thereby eliminating false positives in the analysis. An example for this task can be seen in Figure 4.

An example of the method's application can be seen in Figure 5, where three input images are shown, with their respective minutiae classified by color: green indicates internal

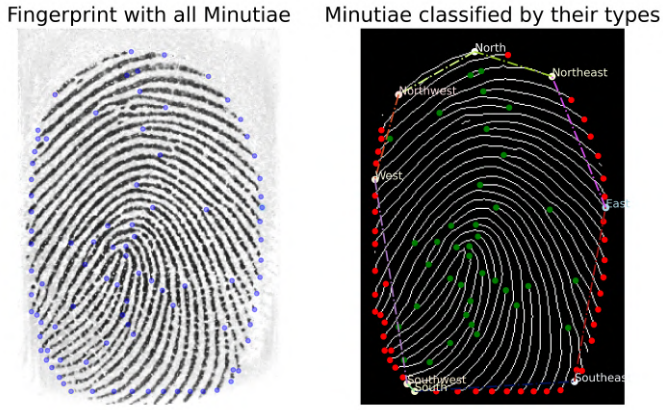

Fig. 4. Representation of the application of the Method

minutiae, and red indicates border minutiae.

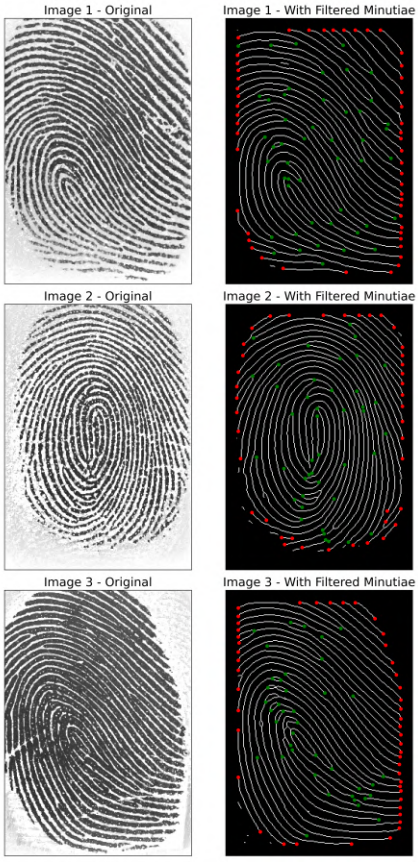

Fig. 5. Example for Minutiae Filtering

C. Complexity Analysis

To evaluate the computational efficiency of the proposed method, an asymptotic analysis of the components is conducted.

The complexity of the fingerprint alignment process is primarily linked to the number of angles tested and the size of the image. The Big-O analysis for the fingerprint alignment algorithm is as follows:

- Determination of the region of interest (ROI) mask using a CNN: O(n), where n represents the number of pixels in the image.
- Reduction of the image to 20% or up to 200x200 pixels: O(n),
- Rotation and analysis process, performed for 90 rotation angles of the candidates;
  - Identification of the boundaries in the reduced image: O(m), where m represents the image with dimensions reduced to 20%, that is, $0.04 \cdot n$
  - Rotation of the image: O(m),
  - Division into quadrants: O(m),
  - Calculation of the variables of sum, ratio, and deviation for the current rotation: O(m),
- Total complexity: O(n)

The minutiae filtering process, in turn, exhibits a total dependence on the number of minutiae. This is primarily due to the need to calculate the distance of each minutia relative to the polygon formed by the extreme points. The Big-O analysis for the filtering algorithm is as follows:

- Identification of extreme minutiae: O(q), where q is the number of minutiae,
- Creation of the polygon: O(1) (always 8 points),
- Calculation of the distance for each minutia: O(q),
- Total complexity: $O(q)$

It is important to note that, although the asymptotic complexity is as described above, the use of optimized libraries such as OpenCV, NumPy, and TensorFlow can result in superior practical performance. For example, using a TPUv2 [18] dedicated for this tests, the computed times for the minutiae filtering approach were $1.41ms \pm 215\mu s$. As for the alignment approach, there are two main times to consider: the first to find the ROI, $129ms \pm 4.83ms$, and the one related to the alignment process itself, $46.8ms \pm 1.28ms$.

In [2], a CNN with complexity O(n) is used. However, this approach requires a training step to adjust the network's weights. In [4], an initial estimation of the orientation field is necessary, and the method depends on training. The complexity is variable and depends on the number of patches used. For [3], it requires the precise location of all singular points, and this process may have a complexity of O(n), as the operations involve each pixel of the image.

In [5], it is necessary to train a classifier for the task, and its complexity is approximately O(P^2) to O(P^3), where P represents the size of the training set [19]. In [6], the use of two nested loops results in a complexity of O(q^2). Additionally, the algorithm's performance depends on some free parameters. In [7], the need to compare one detail to all others introduces a computation that tends to be quadratic, resulting in a complexity of O(q^2), furthermore, it presents complete sensitivity to the quality of the ROI found.

IV. Experiments

To evaluate the quality of the proposed methods, the FVC2000-Db2, FVC2002-Db2, and FVC2004-Db3 datasets from the Fingerprint Verification Competition (FVC) [13] were used, with a focus on set B, which contains 80 images distributed across 8 samples for 10 classes. The choice of these datasets was motivated by the variability in capture technologies: capacitive, optical, and thermal scanning, as well as the two-year time gap between them, allowing for a robust analysis of the performance of the proposed models.

For the task of segmenting the fingerprint images into ridges and valleys, a Gabor filter-based approach is used [9]. Subsequently, the image is skeletonized [14], and then the minutiae are extracted using the method of Crossing Number [8]. An example of this pipeline can be found in Figure 6.

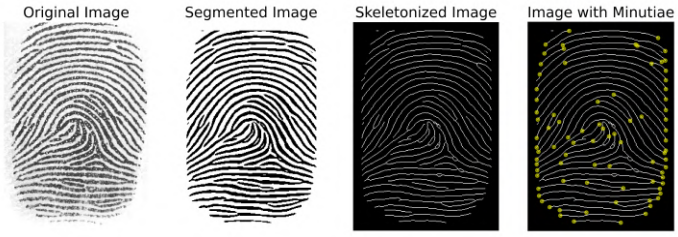

Fig. 6. Representation of the basic process occurring on an image

A. Approach

To perform the experimental task, four variations of the images were used: the original image(raw), the aligned image, and both approaches with filtered and unfiltered minutiae. In the images with filtered minutiae, the minutiae located on the edges were removed from the set of points applied to the distance measures that were employed.

The study focuses on measuring the similarity between minutiae maps from different samples of the same finger. These maps often vary in size, as the number of minutiae differs, which introduces asymmetry between the sets. To address this, three distance measures were selected for their ability to handle such asymmetries and capture different aspects of similarity, namely:

- Pompeiu-Hausdorff: measures the largest of the smallest distances between two sets of points and can be defined as the maximum of the minimum distance that a point in one set has to the other set [15].
- Earth Mover's: evaluates the minimum amount of "work" required to transform one distribution of points into another, where work is defined as the amount of "mass" moved times the distance that mass is moved [16].
- Chamfer: given two point sets α and β, the distance is defined as the sum of the distances of each point in α to its nearest neighbor in β, plus the sum of the distances of each point in β to its nearest neighbor in α [17].

Since these distances are used as similarity verification mechanisms, it is important to remember that the smaller the distance (closer to zero), the greater the similarity; the larger the distance, the more different the analyzed elements are. To evaluate the results, the threshold where the Equal Error Rate (EER) occurs the values were calculated, which is the point at which the acceptance and rejection errors become equal. And using this point, several metrics were calculated, such as Accuracy, TAR, TRR, Cohen's Kappa, beyond ROC AUC which is the only independent of this threshold.

B. Comparative Analysis

The results obtained are presented in the tables, I, II, III that follows. The results are based on metrics for each distance method used in the experiments. The metrics are expressed as percentages. The last four values of these, are calculated using the binary predictions produced using as the threshold the value where the errors are equals.

The analysis of the approaches reveals that the combination of alignment and filtering is the most effective in terms of performance based on the metrics used. This approach outperforms the others in almost all distance methods and test scenarios, offering more precise and consistent results. The introduction of alignment and filtering significantly improves performance, especially in more complex methods, such as Chamfer and Earth Mover's.

TABLE I
Pompeiu-Hausdorff metrics

Approach	ROC AUC	Kappa	TAR	TRR	EER
2000/Db2_b (Capacitive Sensor)					
Raw	59.44	4.55	56.11	56.61	43.64
Raw/Filtered	55.10	2.37	53.44	53.49	46.54
Aligned	56.04	2.14	53.05	53.23	46.86
Aligned/Filtered	**63.42**	**6.25**	**58.40**	**58.56**	**41.52**
2002/Db2_b (Optical Sensor)					
Raw	57.60	3.96	55.65	55.69	44.33
Raw/Filtered	58.10	3.67	55.23	55.32	44.73
Aligned	56.65	3.15	54.81	54.39	45.40
Aligned/Filtered	**60.99**	**5.98**	**58.16**	**58.29**	**41.78**
2004/Db3_b (Thermal sweeping Sensor)					
Raw	60.82	4.98	57.14	56.84	43.01
Raw/Filtered	58.47	4.20	56.07	55.90	44.02
Aligned	59.19	4.94	56.79	57.05	43.08
Aligned/Filtered	**64.86**	**7.52**	**60.00**	**60.10**	**39.95**

In Table IV, the accuracy representing whether the pairs are about genuine or non-genuine. Calculated using the same threshold for the tables above, the EER's threshold. For different scenario combinations is compared. In the last two lines, is calculated the difference between the best result in the case for two approach, relative to the case of using the original image and with the entire proposed method applied. It can be determined that, on average, the fingerprint without any processing is 2.59 percentage points worse than the best case, while the proposed approach is 0.56 percentage points worse, representing a difference of nearly five times in absolute terms. Furthermore, the highest average within all approaches, that is showed in the last column, is achieved with the use of the toolkit.

TABLE II
EARTH MOVER'S METRIC

Approach	ROC AUC	Kappa	TAR	TRR	EER
2000/Db2_b (Capacitive Sensor)					
Raw	**77.84**	**17.65**	**69.85**	**69.82**	**30.17**
Raw/Filtered	72.55	13.28	66.03	65.95	34.01
Aligned	73.03	14.12	66.79	66.74	33.24
Aligned/Filtered	76.54	16.80	69.08	69.14	30.89
2002/Db2_b (Optical Sensor)					
Raw	64,70	8.36	61.09	60.97	38.97
Raw/Filtered	**67,23**	9.97	62.76	62.79	37.23
Aligned	65,60	9.45	62.34	62.14	37.76
Aligned/Filtered	66,75	**10.71**	**63.60**	**63.53**	**36.44**
2004/Db3_b (Thermal sweeping Sensor)					
Raw	67,95	8.38	61.07	61.01	38.96
Raw/Filtered	66,97	9.37	62.14	61.12	38.37
Aligned	**75.31**	**16.55**	**68.93**	**69.06**	**31.01**
Aligned/Filtered	73.09	14.96	67.86	67.50	32.32

TABLE III
CHAMFER METRICS

Approach	ROC AUC	Kappa	TAR	TRR	EER
2000/Db2_b (Capacitive Sensor)					
Raw	78.02	18.45	70.61	70.42	29.49
Raw/Filtered	76.50	16.01	68.32	68.51	31.59
Aligned	**80.49**	**22.79**	**73.66**	**73.80**	**26.27**
Aligned/Filtered	79.03	21.02	72.52	72.45	27.52
2002/Db2_b (Optical Sensor)					
Raw	65.15	7.36	**59.83**	59.91	40.13
Raw/Filtered	**65.54**	**7.52**	**59.83**	**60.24**	**39.97**
Aligned	60.03	4.61	56.49	56.54	43.49
Aligned/Filtered	63.80	6.30	58.58	58.65	41.39
2004/Db3_b (Thermal sweeping Sensor)					
Raw	82.20	23.53	75.00	74.13	25.44
Raw/Filtered	81.78	25.12	75.36	75.52	24.56
Aligned	82.35	24.22	74.64	74.97	25.20
Aligned/Filtered	**82.21**	**25.77**	**76.07**	**75.83**	**24.05**

TABLE IV
ACCURACY

Approach	2000/Db2	2002/Db2	2004/Db3	Average
Pompieu-Hausdorff Distance				
Raw	56.56	55.68	57.06	56.43
Raw/Filtered	53.49	55.31	56.23	55.01
Aligned	53.21	54.42	56.96	54.86
Aligned/Filtered	**58.54**	**58.27**	**60.13**	**58.98**
Gap for Original	1.98	2.59	3.07	2.55
Gap for Toolkit	0.00	0.00	0.00	0.00
Earth Mover's Distance				
Raw	**69.82**	60.98	61.01	63.95
Raw/Filtered	65.96	62.79	61.90	63.55
Aligned	66.75	62.16	**68.86**	65.92
Aligned/Filtered	69.14	**63.53**	67.37	**66.68**
Gap for Original	0.00	2.55	7.82	2.73
Gap for Toolkit	0.68	0.00	1.49	0.00
Chamfer Distance				
Raw	70.44	59.90	74.21	68.18
Raw/Filtered	68.49	**60.20**	75.51	68.07
Aligned	**73.79**	56.53	74.94	68.42
Aligned/Filtered	72.45	58.64	**75.85**	**68.98**
Gap for Original	3.35	0.30	1.64	0.80
Gap for Toolkit	1.34	1.56	0.00	0.00

V. CONCLUSION

The results obtained demonstrate the effectiveness of the proposed method for improving the analysis of similarity between fingerprints. The results show that the use of this approach leads to significant improvements in the performance of fingerprint comparison, surpassing the use of original images in different distance metrics evaluated.

However, it is important to interpret the results with caution. It is essential to recognize that no biometric system is perfect [1]. Both methods critically depend on previous processes, such as the quality of the input images, and processes of digital segmentation, mask creation, and feature extraction. Future work should focus on improving the methods, making them less dependent on the procedures mentioned above.

This work contributes to the analysis of biometric similarity by proposing an innovative approach. Future research should focus on refining this methodology, exploring new mathematical methods to minimize possible errors in performance while making it more independent.

VI. ACKNOWLEDGMENT

We sincerely thank InfantID, CAPES (Finance Code 001), CNPq, Araucária Foundation, and FINEP, for their financial support.

REFERENCES

[1] D. Maltoni, "A tutorial on fingerprint recognition," in *Advanced Studies in Biometrics, Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2005, pp. 43-68

[2] P. Schuch, J. M. May, and C. Busch, "Unsupervised learning of fingerprint rotations," in *Proc. 2018 Int. Conf. Biometrics Special Interest Group (BIOSIG)*, 2018, pp. 1-6, doi: 10.23919/BIOSIG.2018.8553096.

[3] T. Hotz, "Intrinsic coordinates for fingerprints based on their longitudinal axis," in *Proc. 6th Int. Symp. Image and Signal Processing and Analysis*, 2009, pp. 500-504, doi: 10.1109/ISPA.2009.5297679.

[4] X. Yang, J. Feng, and J. Zhou, "Localized dictionaries based orientation field estimation for latent fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 955-969, May 2014, doi: 10.1109/TPAMI.2013.184.

[5] P. Mansukhani *et al.*, "Using support vector machines to eliminate false minutiae matches during fingerprint verification," in *Proc. Salil Prabhakar and Arun A. Ross (Eds.),* 2007, p. 65390B. DOI: 10.1117/12.720769.

[6] S. Jabeen and S. A. Khan, "A hybrid false minutiae removal algorithm with boundary elimination," in *Proc. 2008 IEEE Int. Conf. System of Systems Engineering*, 2008, pp. 1-6, doi: 10.1109/SYSOSE.2008.4724177.

[7] F. Abed and A. Maroof, "Fingerprint image pre- and post-processing methods for minutiae extraction," *AL-Rafidain J. Comput. Sci. Math.*, vol. 6, no. 1, pp. 97-110, Mar. 2009, doi: 10.33899/csmj.2009.163768.

[8] C. Arcelli and G. S. Di Baja, "A width-independent fast thinning algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-7, no. 4, pp. 463-474, Jul. 1985, doi: 10.1109/TPAMI.1985.4767685.

[9] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777-789, Aug. 1998, doi: 10.1109/34.709565.

[10] S. Chavan, P. Mundada, and D. Pal, "Fingerprint authentication using Gabor filter based matching algorithm," in *Proc. 2015 Int. Conf. Technologies for Sustainable Development (ICTSD)*, 2015, pp. 1-6, doi: 10.1109/ICTSD.2015.7095910.

[11] Natosafe, "Infant.ID - Born to be unique," *Natosafe*, [Online]. Available: https://natosafe.com.br/. [Accessed: Jul. 6, 2024].

[12] J. H. Wegstein, "An automated fingerprint identification system," *U.S. Government Publication*, 1982, doi: 10.6028/nbs.sp.500-89. [Online]. Available: https://dx.doi.org/10.6028/nbs.sp.500-89.

[13] FVC-onGoing: Fingerprint Verification Competition*, Biometric System Laboratory - University of Bologna, [Online]. Available: https://biolab.csr.unibo.it/FvcOnGoing/UI/Form/Home.aspx. [Accessed: Aug. 20, 2024].

[14] L. Lam, S.-W. Lee, and C. Y. Suen, "Thinning methodologies—a comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 9, pp. 869-885, Sep. 1992, doi: 10.1109/34.161346.

[15] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 9, pp. 850-863, Sep. 1993, doi: 10.1109/34.232073.

[16] O. Pele and M. Werman, "Fast and robust earth mover's distances," in *Proc. 2009 IEEE 12th Int. Conf. Computer Vision*, Sep. 2009, pp. 460-467.

[17] A. Bakshi, P. Indyk, R. Jayaram, S. Silwal, and E. Waingarten, "A near-linear time algorithm for the Chamfer distance," *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2307.03043.

[18] N. P. Jouppi, D. H. Yoon, G. Kurian, S. Li, N. Patil, J. Laudon, C. Young, and D. Patterson, "A domain-specific supercomputer for training deep neural networks," Commun. ACM, vol. 63, no. 7, pp. 67–78, Jul. 2020, doi: 10.1145/3360307.

[19] L. Bottou and C.-J. Lin, "Support Vector Machine Solvers," Large-Scale Kernel Machines, pp. 1-28, Aug. 2007. [Online]. Available: https://dx.doi.org/10.7551/mitpress/7496.003.0003. doi: 10.7551/mitpress/7496.003.0003.