Iris Recognition System

Nawaf Senan[1], Ahmed Shalaby[1], and Ahmed Ashraf [1]

[1]Affiliation One, Cairo-Egypt

May 20, 2024

Abstract

Iris recognition is a critical technology in the field of biometric authentication, providing high accuracy and security. This paper presents a comprehensive study on the development and implementation of an iris recognition system. The system employs advanced image processing techniques to capture and preprocess iris images, segmentation, A dimensionally consistent representation of the iris region is created in the normalization step, followed by feature extraction using Haar wavelet and the Hamming distance for matching. The proposed system has been tested on a standard iris image dataset, demonstrating a high recognition rate and robustness against various challenges such as noise, occlusions, and variations in illumination. The results indicate that the system is effective for real-world applications in secure access control and identity verification. Future work includes enhancing the system's speed and exploring deep learning methods for improved performance.

Keywords: Preprocessing, Feature Extraction, Matching (Hamming Distance, Cosine Similarity).

1 DB Description

In this study, we utilized the CASIA-Iris-Syn database available from Kaggle, specifically designed for research in iris recognition. This database includes synthetic iris images generated to mimic the variations and complexities found in real-world scenarios. The CASIA-Iris-Syn dataset is widely recognized for its robustness and diversity, making it an ideal choice for developing and testing biometric authentication systems.

Dataset Details

Source: CASIA-Iris-Syn Database on Kaggle [1]

Total Subjects: 1000 (in the full dataset)

Selected Subjects: 48

Images per Subject: Each subject in the dataset has 10 images to account for variations in capture conditions, including changes in illumination, angle, and occlusions.

Image Format: The images are provided in a “jpg” format suitable for image processing tasks.

2 Preprocessing and Segmentation

Preprocessing is a crucial step in the development of our iris recognition system. It involves preparing the raw iris images from the CASIA-Iris-Syn database to enhance their quality and ensure consistency for accurate feature extraction and matching. The preprocessing steps implemented in our study are outlined below:

2.1 Grayscale Conversion

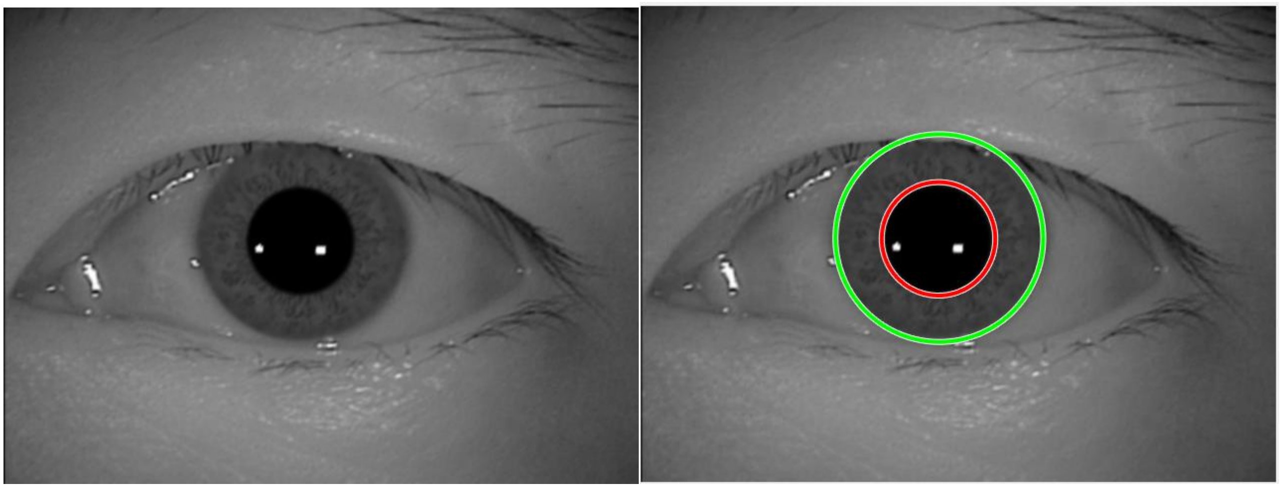
All iris images are converted from color to grayscale if necessary to simplify the processing and reduce computational complexity, as the color information is not essential for iris pattern recognition.

2.2 Noise Reduction

To mitigate the impact of noise and enhance image quality, we apply gaussian filtering. This technique helps in removing any salt-and-pepper noise, specular reflections in the eye and other artifacts that might interfere with the recognition process.

2.3 Iris Localization

Iris localization involves detecting the boundaries of the iris and the pupil. We use the Circular Hough Transform for circular edge detection after getting image edges using Canny Edge Detector to accurately locate the inner (pupil) and outer (iris) boundaries. This step ensures that only the iris region is considered for further processing, eliminating irrelevant parts of the image. You can see the result after segmentation in Fig. 1.



(a) Original Image (b) Processed Image

Fig. 1

2.4 Normalization

The localized iris regions are normalized to a fixed size using Daugman's rubber sheet model [2]. This method converts each point in the segmented iris region from Cartesian coordinates (x, y) to a point in a rectangular region represented by polar coordinates (r, θ), effectively making the 360 degree of the iris into a rectangle shape[3]. The transformation is described as follows:

$$I(x(r,\theta), y(r,\theta)) \rightarrow I(r,\theta) \tag{1}$$

and

$$\begin{aligned} x(r,\theta) &= (1-r)x_p(\theta) + r x_I(\theta) \\ y(r,\theta) &= (1-r)y_p(\theta) + r y_I(\theta) \end{aligned} \tag{2}$$

where $I(x,y)$ represents the iris region, (x,y) are the Cartesian coordinates, (r,θ) are the corresponding normalized polar coordinates, and x_p, y_p and x_I, y_I are the coordinates of the pupil and iris boundaries in the θ direction, respectively. The result of this normalization process is the unwrapped iris image, as illustrated in Fig. 2.

Fig. 2 Normalized Iris

3 Feature Extraction

Feature extraction is a critical step in iris recognition, where distinctive patterns in the iris are identified and encoded for comparison and matching. In our approach, we use Haar wavelet decomposition to extract meaningful features from the normalized iris image. The detailed steps are as follows:

3.1 Wavelet Decomposition

We apply Haar wavelet decomposition to the normalized iris image at the third level Fig. 3. Wavelet decomposition helps in capturing both frequency and location information, which is essential for detailed feature extraction.

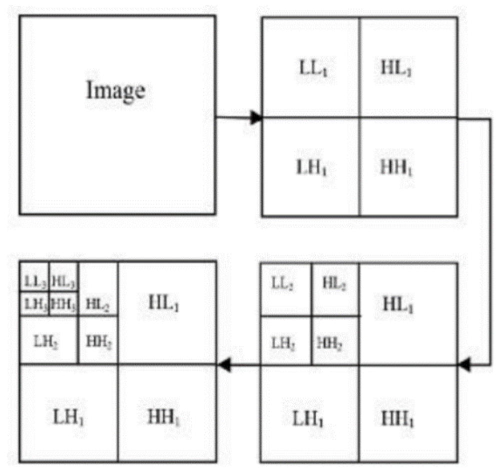


Fig. 3: Haar wavelet 3rd level decomposition

3.2 Analysis of Detail Sub-Bands

In wavelet decomposition, the image is broken down into approximation (CA) and detail sub-bands (CH, CV, CD). For feature extraction, we focus on the detail sub-bands which capture the high-frequency information, such as edges and textures in the iris pattern.

- HL (Horizontal Detail Coefficients): Captured in CHn
- LH (Vertical Detail Coefficients): Captured in CVn
- HH (Diagonal Detail Coefficients): Captured in CDn

3.3 Combination of Detail Sub-Bands

The detail sub-bands (HL + LH + HH) are combined to form a comprehensive feature set. This combination ensures that the extracted features represent various directional details in the iris image[2].

3.4 Iris Code Generation

To generate the binary iris code, we apply thresholding to the combined wavelet features. This step converts the continuous wavelet coefficients into a binary format, which simplifies the matching process and enhances robustness. You can see the iris code in Fig. 4.

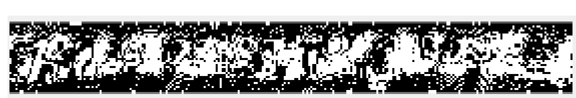


Fig. 4: Iris code

4 Matching

In iris recognition systems, matching is the process of comparing an extracted iris code with stored iris codes to determine if there is a match. This process involves calculating a similarity or dissimilarity measure between the iris codes. Two common methods for this are Hamming distance and Cosine similarity. Below is a detailed conceptual overview of these methods.

4.1 Hamming Distance

The Hamming distance is a measure of the difference between two binary strings of equal length. It counts the number of positions at which the corresponding bits are different. In the context of iris recognition, the Hamming distance provides a straightforward way to measure how many bits differ between two iris codes.

Application in Iris Recognition:

- Binary Representation: Iris codes are represented as binary strings, where each bit corresponds to a specific feature in the iris pattern.
- XOR Operation: The Hamming distance is calculated using an XOR operation on the corresponding bits of two iris codes. The XOR operation outputs 1 for differing bits and 0 for matching bits.
- Summation: The resulting binary string from the XOR operation is summed to get the total number of differing bits.

4.2 Cosine Distance

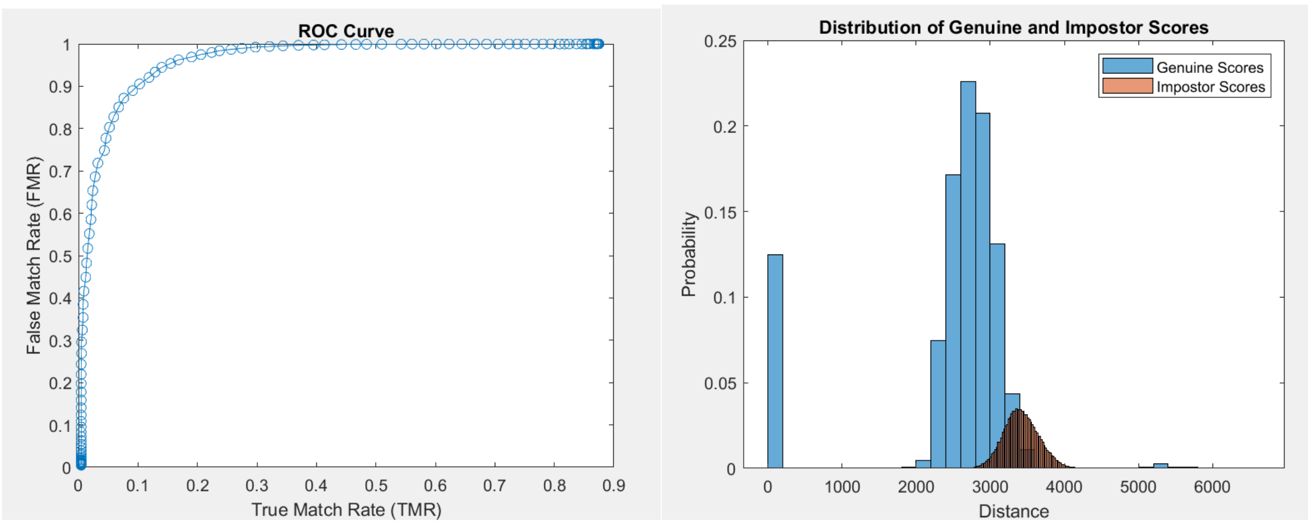
Cosine Distance is a measure of distance between two vectors in an inner product space. It is calculated as the cosine of the angle between the two vectors. In the context of iris recognition, it treats the iris codes as vectors and measures their directional distance.

Application in Iris Recognition

- Vector Representation: Iris codes are treated as vectors in a high-dimensional space.
- Dot Product: The dot product of the two vectors is calculated, which measures the extent to which the vectors point in the same direction.
- Norms: The norms (lengths) of the vectors are calculated to normalize the dot product.
- Cosine Similarity: The cosine of the angle between the vectors is computed using the dot product and the norms. This value ranges from -1 (completely dissimilar) to 1 (completely similar).
- Cosine Distance: The cosine distance is derived as 1−cosine similarity, providing a measure of dissimilarity.

5 Results & Discussion

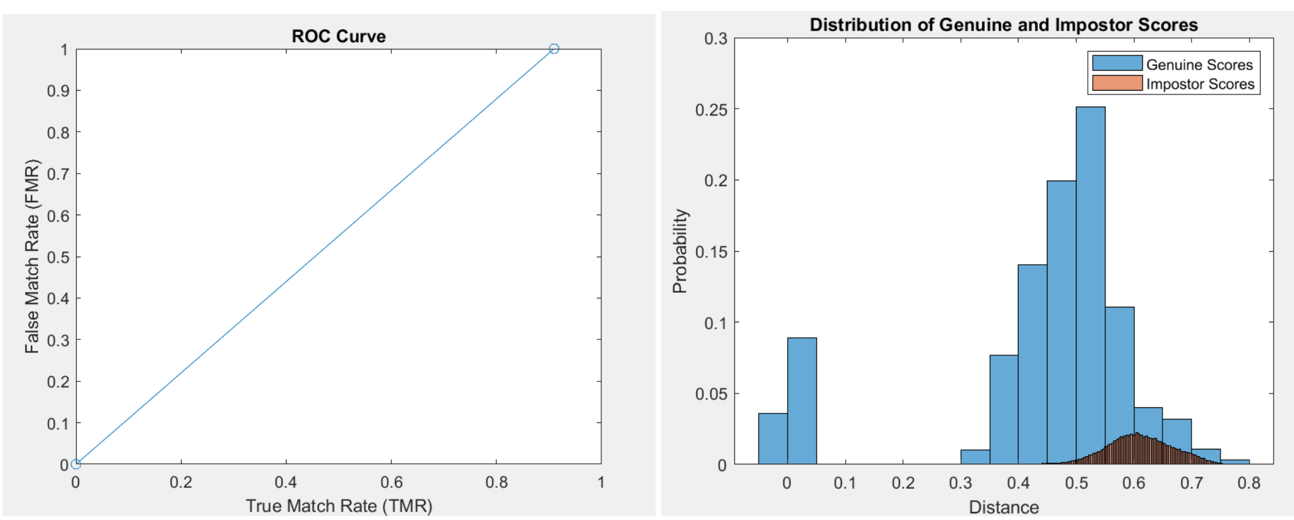
Here we can see the difference in the two figures Fig. 5 and Fig. 6 for Hamming Distance and Cosine Distance, respectively.



(a) ROC Curve

(b) Genuine/Imposter Curve

Fig. 5: Hamming Distance Curves



(a) ROC Curve

(b) Genuine/Imposter Curve

Fig. 6: Cosine Distance Curves

As observed, the ROC curve for Hamming Distance demonstrates superior performance compared to Cosine Distance. The ROC curve for Cosine distance resembles that of random classification, indicating its ineffectiveness in distinguishing between classes accurately.

Evaluation Metrics

Evaluation Metrics	Hamming Distance	Cosine Distance
D-prime	1.3926	0 (NaN)
EER	0.4973 %	0.0000 %
TMR (1-FRR) at FMR (1%)	0.276	0.6139
TMR (1-FRR) at FMR (0.01%)	0.6283	0.875
Identification Rate	98.96 %	90.62 %

Table 1: Evaluation Metrics

From the Evaluation values (table 1):

- **Hamming Distance** Specifically designed for binary (logical) vectors, Hamming distance counts the number of differing bits between two binary codes. This makes it a natural fit for comparing binary iris codes, as it directly measures dissimilarity in a straightforward and effective manner.

  Cosine similarity (and by extension, cosine distance) measures the cosine of the angle between two vectors. While effective for high-dimensional real-valued data, it is less suitable for binary vectors. Binary vectors do not benefit from the angular similarity measure, which is why the performance is less efficient in this context.

- **The Hamming distance** effectively differentiates between genuine and impostor pairs because it is sensitive to the bit-level differences in the binary iris codes. **The cosine distance**, however, fails to leverage the binary nature of the iris codes, resulting in significant overlap in the score distributions of genuine and impostor pairs.

While a higher TMR values at specific FMR thresholds, highlighting its superior performance in maintaining low false match rates, there are few considerations we need to point out.

- Hamming Distance: An EER of 0.4973% is very low and indicates good performance. The accompanying ROC curve and the separation of genuine and imposter score distributions further support the effectiveness of the Hamming Distance method.
- Cosine Distance: An EER of 0.0000% seems ideal, but it could indicate overfitting or other issues. The ROC curve for Cosine Distance shows a straight diagonal line, which suggests that the method is performing at the level of random guessing. This discrepancy suggests that while the EER might be low, other performance metrics do not support the Cosine Distance method's reliability.

with the Cosine Distance method has higher TMR at specific FMR values, the overall performance of the Hamming Distance method is superior due to:

- Better and more consistent discriminative power across various thresholds.
- Reliable ROC curve indicating good performance at multiple operating points.
- Clear separation in score distributions, indicating reliable identification.
- Higher identification rate, suggesting better practical application.

In biometric systems, consistent and reliable performance across multiple metrics is crucial. Despite the higher TMR at specific FMR values, the overall poor performance metrics of the Cosine Distance method make it less suitable for reliable iris recognition. Therefore, *Hamming Distance* is the preferred method due to its robustness and consistency in performance.

Conclusion

In conclusion, our study presents a comprehensive investigation into the development and implementation of an iris recognition system, leveraging advanced image processing techniques to ensure high accuracy and security in biometric authentication.

Starting with the preprocessing and segmentation stage, we meticulously prepared the iris images from the CASIA-Iris-Syn database, employing techniques such as grayscale conversion, noise reduction using Gaussian filtering, and precise iris localization through Circular Hough Transform. This step ensured the extraction of relevant iris regions while mitigating the impact of noise and artifacts.

Normalization followed, where the localized iris regions were transformed into a dimensionally consistent representation using Daugman's rubber sheet model. This crucial step facilitated feature extraction by providing a standardized format for further processing.

Moving to feature extraction, we employed Haar wavelet decomposition to capture distinctive patterns in the iris. Analyzing detail sub-bands enabled us to extract meaningful features capturing high-frequency information crucial for accurate recognition. The combination of detail sub-bands ensured a comprehensive feature set, contributing to the robustness of our system.

Feature extraction culminated in iris code generation, where we applied thresholding to the combined wavelet features, converting them into a binary format. This step simplified the matching process and enhanced the system's robustness against variations in capture conditions.

In the matching stage, we evaluated two common methods: Hamming Distance and Cosine Distance. While both methods were assessed rigorously, the Hamming Distance method demonstrated superior performance across various evaluation metrics. Its sensitivity to bit-level differences in binary iris codes, coupled with consistent discriminative power, made it the preferred choice for reliable iris recognition.

In summary, our study underscores the importance of robust preprocessing, feature extraction, and matching algorithms in ensuring the effectiveness of iris recognition systems. By meticulously addressing each stage of system development and evaluation, we have established a framework for secure access control and identity verification in real-world applications.

References

[1] Reyhanii, M. (n.d.). CASIA-Iris-Syn. Kaggle. Retrieved from https://www.kaggle.com/datasets/monareyhanii/casia-iris-syn

[2] Houston, C., 2010. Iris Segmentation and Recognition Using Circular Hough Transform and Wavelet Features. *Rochester Institute of Technology*.

[3] Zhou, Z., & Yang, J. (2015). Iris recognition based on the low-order norms of gradient components. *ResearchGate*. Retrieved from https://www.researchgate.net/publication/281825031_Iris_Recognition_Based_On_the_Low_Order_Norms_of_Gradient_Components