APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 6 No. 2 (2025) 698-712
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

An Enhanced Feature Extraction using Chebyshev Wavelet Filter for Iris Recognition System

Muhammad Hayatudeen[1], Sapiee Jamel[1]*

[1] *Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA*

*Corresponding Author: *sapiee@uthm.edu.my*
DOI: https://doi.org/10.30880/aitcs.2025.06.02.038

Article Info

Received: 20 July 2025
Accepted: 19 November 2025
Available online: 30 November 2025

Keywords

Biometrics, Iris Recognition, Feature Extraction, Chebyshev Wavelet Filter

Abstract

This research explores an enhanced feature extraction method for iris recognition, addressing the limitations of existing techniques like Gabor filters and wavelet transforms, which struggle with noise and image variations in non-ideal conditions. The This research designed and evaluated a feature extraction method using Chebyshev wavelet filter, testing its performance in terms of False Acceptance Rate and False Rejection Rate, and assessing its effectiveness in handling variations. The study utilized two publicly available datasets (CASIA, MMU), the methodology involves preprocessing iris images from the datasets and applying the Chebyshev wavelet filter for feature extraction and employing support vector machine for classification. This system hepls improve recognition accuracy and robustness in non-ideal conditions. The outcome of this project achieved higher accuracy rates and lower FAR and FRR compared to traditional methods. This research contributes to the development of more efficient iris recognition systems.

1. Introduction

In recent years biometric systems have become one of the most important forms of security systems as they have continuously been proving to be fast and reliable when it comes to system security in the world of fast evolving technology [1]. As technology advances the field of biometrics identification has also advanced along with it as it is constantly evolving, and even part of the evolving biometrics iris recognition stands out as it is one of the most accurate and reliable methods for personal identification. Iris recognition takes advantage of the unique patterns in the human iris as it does not change overtime which makes it a reliable biometric trait [2]. Developing an effective iris recognition system in today's world is very important, especially for areas that have already integrated biometrics into their systems, areas like banking and border control but in order to archive that iris recognition has to be very accurate and efficient [3]. However, despite advancements in technology, existing iris recognition systems face challenges, particularly in non-ideal conditions where factors such as noise, occlusions, and variations can significantly affect the systems performance [4].

The accuracy of the Iris recognition system depends on effective feature extraction techniques. Although techniques like the Gabor filters and wavelet transforms have been widely used, they often struggle to handle the complexity of iris images that are in non-ideal conditions [5]. Even though iris recognition being a highly secure biometric technique, efficiently and accurately extracting iris features in real time scenarios where there is variations in image quality and presence of occlusions is challenging and because of that, many iris recognition methods rely on feature extraction algorithms that are computationally intensive and While they demonstrate high accuracy they require great processing power and memory which limits their efficiency and hinders their deployment on devices with limited resources [6]. Therefore, a more efficient feature extraction method for iris recognition that is both reliable and practical while maintaining balance between computational efficiency and recognition accuracy at the same time is required. This research aims to enhance the efficiency and accuracy of iris recognition systems by utilizing Chebyshev wavelet filter for feature extraction. The study aims to address the limitations of feature extraction method by taking advantage of the unique Chebyshev wavelets properties.

This research proposes an enhanced feature extraction method using Chebyshev wavelet filter to address iris recognition challenges. Chebyshev wavelets are a new family of wavelet transforms derived from Chebyshev polynomials. They offer support, sharp frequency selectivity, and excellent denoising properties. In our approach, we designed a Chebyshev based filter bank to capture iris texture at multiple scales. The workflow segments and normalizes the iris image, applies the Chebyshev wavelet transform for feature extraction, and then classifies using a support vector machine (SVM) classifier. We compare this to a baseline Gabor-based extractor as in previous work. Experiments on standard iris databases (CASIA and MMU) show that the Chebyshev-wavelet approach gives higher recognition accuracy and lower false accept/reject rates under noisy and non-ideal conditions. The research also focusses on how the proposed method overcomes the challenges faced by traditional techniques and whether it achieved superior performance metrics, such as lower False Rejection Rate and False Acceptance rate or not [6].

The objective of this study is to design an enhanced feature extraction method using Chebyshev wavelet filter and then evaluate its effectiveness using established datasets and evaluate its performance metrics and compare the result against existing feature extraction methods. The expected outcome of this research is a more reliable iris recognition system capable of functioning in diverse conditions.

2. Related Work

Iris recognition systems have become a focus area of research within biometric systems because of its high accuracy and reliability. The iris systems utilize the uniqueness and stability of iris patterns in the human iris for identification which makes them resistant to change over time compared to other biometric systems. The process of iris recognition includes multiple stages which are image acquisition: this stage is the initial step where iris images are acquired using specialized cameras that capture the intricate details of the iris texture. Preprocessing: this stage is when the acquired images undergo enhancement techniques to improve their quality and prepare them for subsequent processing stages. This includes addressing issues like noise reduction, contrast enhancement and illumination correction to create an accurate feature [7], [8]. Segmentation: this stage involves separating the iris region from the rest of the eye image. Techniques like the circular Hough transform (CHT) and deep learning-based methods are used to accurately outline boundaries of the iris [1] once it is segmented, it is normalized to a standard size and shape typically a rectangular format. Feature extraction: this is the stage to extract distinctive features that represent the unique texture and pattern of the iris. After that the last stage is matching this stage involves comparing the extracted features of an input iris image with those stored in a database to determine a match as seen in Fig. 1 on the iris system steps. In Iris recognition systems feature extraction is important as it impacts the systems performance in terms of accuracy and speed [8].

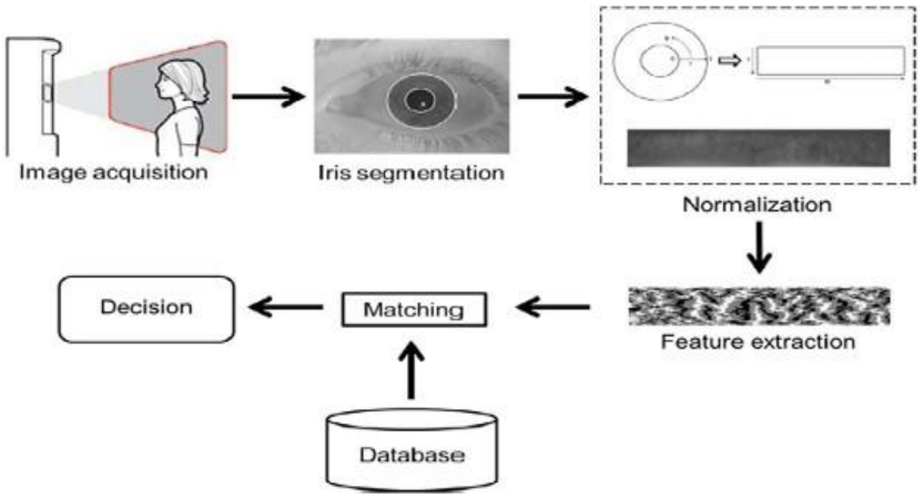


Fig. 1 *Iris recognition System [8]*

2.1 Existing Traditional and Advanced Feature Extraction Techniques

Multiple feature extraction techniques have been developed over the years to enhance feature extraction, including methods like the Gabor filters and wavelet transforms which laid the groundwork for more advanced approaches [10].

2.1.1 Gabor Filters

Gabor filters have been widely used for texture analysis in Iris recognition because of their ability to extract frequency information. These filters analyze the frequency and orientation characteristics of iris textures, generating feature vectors for comparison. Gabor filters have shown high accuracy in controlled environments. However, Gabor filters can be computationally intensive, especially when handling large datasets [11]. One of the most influential iris recognition algorithms using Gabor filters was developed by Daugman who a pioneer researcher in this field is. His algorithm involves detecting the boundaries of the pupil and iris using an integrodifferential operator and normalizing the extracted iris region using Daugmans rubber sheet model, then the iris image is then transformed into a series of encodings which is called IrisCode using a Gabor phase-quadrant feature descriptor and finally in the identification stage the Hamming distance between IrisCodes is calculated to determine the recognition result [6], [8].

2.1.2 Wavelet Transforms

Wavelet transforms provide a multi-resolution analysis of iris images, that allows the extraction of features across different scales. This process involves breaking down the original image into sub-images which are labeled low-low (LL), Low-High (LH), High-Low (HL), and High-High (HH), representing approximations and details at different resolution levels. The Discrete Wavelet Transform (DWT) is used for its computational efficiency while the Fast Wavelet Transform (FWT) provides a mathematical technique to convert signals into measurement based on wavelets. Different types of mother wavelets, include Haar, Log-Gabor, Morlet, Daubechies, Symlet, Coiflet, Bi-orthogonal, Meyer, Fejer-Korovkin, and Reverse bi-orthogonal, can be used for feature extraction. The choice of wavelet family impacts on the system's performance, and research indicates that Symlet wavelet features combined with the Spearman distance metric perform well for iris recognition, especially with distant images. There is also research that suggests there is no single best wavelet type for iris feature extraction as the optimal choice may vary depending on the database used and other factors [1], [12].

2.1.3 Hybrid Feature Extraction Method

The hybrid method is a feature extraction method that combines and leverage the strengths of multiple feature extraction techniques. This method involves preprocessing of images and application of advanced classifiers. The hybrid model utilizes techniques like Gabor filters and wavelet transforms which are good at capturing unique features of iris images which provide more detailed representation than when using the traditional single method approach [13]. The choice of which technique to combine in hybrid approach depends on the specific goals of the Iris recognition system, the available computational resources, and the characteristics of the dataset.

2.2 Challenges in Iris Recognition Systems

There are multiple challenges in Iris recognition and here are some of the challenges that are common in Iris recognition.

2.2.1 Noise and Image Quality Variations

Unconstrained iris images often suffer from noise and quality variations caused by factors like lighting, distance, and camera limitations. These variations can degrade image quality and affect the accuracy of feature extraction methods [12].

2.2.2 Occlusions, Reflections

Occlusions from eyelids, eyelashes and also reflections from eyeglasses or light sources or even contact lens, and pupillary membrane, present challenges for accurate iris feature extraction. To tackle this challenge an effective preprocessing technique and iris segmentation are essential to mitigate these occlusions and reflections [12]

2.3 Chebyshev Wavelets

Chebyshev wavelets are gotten from Chebyshev polynomials which form a complete orthonormal system. Chebyshev polynomials are polynomials with the largest possible leading coefficient whose absolute value on the interval [-1,1] is bounded by 1. Chebyshev wavelets are useful for numerical methods in integral equations and Partial Differential Equations because of their operational matrices. Chebyshev wavelets are defied on the real interval [0, 1] and they show orthogonality on this interval with respect to a specific weight function. They can be constructed as recursive wavelets for piecewise polynomial spaces Chebyshev can be applied in multiple fields, like it can be applied in signal processing for denoising and compression[2] It is also applied in fields like Data mining, machine learning, and cryptography.

Chebyshev polynomials are a family of orthogonal polynomials with superior approximation properties, and they come in two kinds first-kind and second Kind and satisfy the recurrence ($T0(x)=1$, $T1(x)=x$, $Tm(x)=2xTm-1(x)-Tm-2(x)$) [15] these polynomials have minimax behavior and are excellent for function approximation. Chebyshev polynomial-based filters are known for having very sharp frequency cutoffs and equiripple passbands in signal processing.

Chebyshev wavelets and similar polynomial wavelets have found applications in other signal processing tasks. Chebyshev wavelet transformed into image compression and reported dramatically compared to Haar/Daubechies filters. This indicates strong approximation performance. We hypothesize that analogous benefits will hold for iris textures, yielding high information “codes” and lower matching. The present work is the first to investigate this in the context of biometric iris data [4], [16], [17].

The Chebyshev wavelet filter design used in this project is based on the method proposed by Cintra *et al*. [30], which introduced the construction of orthogonal filter banks using Chebyshev polynomials for signal decomposition. This method has also been extended in recent work by Abdulrahman and Tahir and Muhi-Aldeen *et al*. [31], [32] demonstrating the effectiveness in biometric image processing tasks such as face recognition and skin lesion classification, highlighting their robustness in feature extraction and denoising. These studies confirm the practicality and performance advantages of Chebyshev-based wavelet designs in classification systems, especially under noisy or limited-data conditions.

2.3.1 Application of Chebyshev Wavelets

Chebyshev wavelet transforms have been effectively applied in image compression, enhancement, and denoising. Studies [18], [19] show that Chebyshev wavelets particularly Discrete Chebyshev Wavelet Transforms (DCWT and SCWT) achieve better quality in signal processing compared to traditional wavelets like Haar, Symlet, Coiflet, and Daubechies. These wavelets demonstrate strong performance in multi-level image decomposition, making them suitable for tasks like face recognition and denoising.

In image analysis, Chebyshev filters decompose an image into sub-bands (LL, LH, HL, HH), like standard wavelets, while offering sharper frequency selectivity and denoising benefits [14]. The SCWT variant has shown efficiency in compressing and enhancing images for facial recognition applications, suggesting similar benefits may extend to iris recognition [1], [10], [14], [19], [20], [21], [22], [23].

Chebyshev wavelets can be applied in iris recognition by normalizing the iris image, applying Chebyshev-based decomposition, and extracting texture features from the resulting sub-bands. Their high selectivity allows for capturing fine iris structures like ridges and furrows, while their smoothing properties can suppress noise from reflections, occlusions, or sensor artifacts. The extracted coefficients may be further quantized or reduced (e.g., using PCA) to form compact iris codes suitable for classification [24], [25].

Though direct applications into iris recognition are limited in the literatures, the strong performance in other biometric areas like face recognition indicates Chebyshev wavelets have potential to improve feature clarity and robustness in iris-based systems [10], [17], [21], [24], [25], [26].

Advantages:

- High frequency selectivity and compact support, leading to efficient, localized feature extraction.
- Simple implementation, especially for Type-II filters, which are computationally light (e.g., running averages).
- Natural multi-scale analysis, allowing flexible filter design for coarse-to-fine texture encoding.
- Effective noise suppression, improving robustness in challenging imaging scenarios.

Limitations:

- Non-orthogonality, which may result in redundancy and correlation among features.
- Limited directional selectivity, requiring additional mechanisms to handle orientation-specific texture.
- Fewer established tools and libraries, as Chebyshev wavelets are relatively new in biometric applications.
- Potential sensitivity to artifacts, due to non-smooth filter shapes in certain variants (e.g., Type-I).

2.4 Comparison of Existing Feature Extraction Techniques

This part is the comparison of the performance and findings between the already existing feature extraction techniques. Table 1 shows that Gabor filters achieve up to 95% accuracy but struggle with noise and occlusions and Wavelet transforms are effective in multi resolution analysis, but they are computationally demanding with accuracy around 87.4% while the hybrid wavelet combines multiple wavelet features like with convolutional neural networks (CNNs) which achieves up to 94.1% accuracy [27][28].

Table 1 *Comparison of Existing Feature Extraction Techniques*

Reference	Authors	Technique	Dataset	Key findings	Performance Metrics
[3]	O. Aiyeniko, Y. Adekunle, M. Eze, O. Alao *et al.* (2020)	Gabor Filter	CASIA, MMU, UBIRIS	It performs well in controlled environments, but it is sensitive to noise and occlusions.	Accuracy: 90.3% FRR:9.7% FAR: 8.5%
[4]	A. Mukherjee *et al.* (2021)	Wavelet Transform	UBIRIS.V2, CASIA	It is effective in handling noise, but it is computationally expensive.	Accuracy: 87.4% FRR:11.5% FAR: 12.1%
[5]	A. Ullah, A. Salam, H. El Raoui, D. Sebai, M. Rafie (2022)	Hybrid Wavelet	MMU, UBIRIS	It combines wavelet features with CNNs which improve robustness but increase system complexity.	Accuracy: 94.1% FRR:5.9% FAR: 6.0%

3. Methodology

This section outlines the methodology used in this study to enhance feature extraction for iris recognition using Chebyshev wavelet filter. The research framework for the proposed project is designed to guide the implementation of the project. This framework incorporates Image Acquisition, preprocessing, feature extraction using the Chebyshev Wavelet filter, classification and performance evaluation and Fig. 1 shows an example of an iris recognition system framework.

3.1 Image Acquisition

The methodology begins with the image acquisition from three datasets which are CASIA, MMU, and UBIRIS. Each dataset serves a distinct purpose in testing the robustness of the proposed method. These datasets provide a diverse range of iris images under various conditions allowing for a comprehensive evaluation of the proposed method.

Table 2 *Dataset Parameters*

Dataset	Subjects	Images
CASIA_V4	3 subjects with 10 left and 10 right images = 60 images	81 images
Interval	3 subjects with 7 right images only = 21 images	
MMU	10 subjects with 5 left and 5 right images and 1 subject with 5 left images only	105 images

3.2 Preprocessing

After the image acquisition, this phase is responsible for performing processing tasks which include segmentation, normalization, and noise reduction. So, segmentation isolates the iris region from the rest of the image then normalization standardizes the iris to a fixed scale and orientation. The next is the noise reduction technique which eliminates occlusions caused by the eyelashes, eyelids, and reflections, to ensure the quality of input data for feature extraction. It is done by using Edge detection and Hough Transform to locate the pupil and iris boundaries. This segmentation discards irrelevant regions (such as eyelids and sclera) and outputs a circular iris region. Next, the localized iris is unwrapped and normalized using Daugman's "rubber sheet" model. This transforms the annular iris region into a fixed-size rectangular block, compensating for pupil dilation and eye rotation. The result is a normalized iris image with uniform dimensions, ready for feature extraction.

3.3 Feature Extraction

This is the core methodology, this phase applies the Chebyshev Wavelet Filter to extract key features from the pre-processed iris images, and this involves breaking up the iris image into high and low-frequency components also known as feature vectors and capturing texture details that are essential for discrimination. The phase is designed for efficiency and accuracy by leveraging the unique properties of Chebyshev polynomials to enhance the system.

A bank of 2D Chebyshev wavelet filters to analyze the normalized iris texture was designed for it. Specifically, Chebyshev polynomials of the first kind (orders 1 through 4) in both horizontal and vertical directions were used. By evaluating these polynomials in two dimensions, we construct a set of 16 distinct filters which are 4 horizontal orders × 4 vertical orders. Each filter is size 32×32 pixels and defines a particular frequency response. The filters are not simple low-pass/high-pass pairs; instead, each filter acts as a localized 2D template. These filters inherit the Chebyshev properties: they have compact spatial support and strong selectivity for certain frequency bands.

Once the filters are built, feature extraction proceeds as follows and each normalized iris image is convolved with all 16 Chebyshev filters. This produces 16 filtered images, each highlighting different texture details. For each filtered output, we compute four statistical features the mean intensity, standard deviation, skewness, and kurtosis of the filter response. These statistics capture the distribution of the texture information at that scale and orientation. Collectively, the 16 filters × 4 statistics yield a 64-dimensional feature vector representing the iris pattern. This one-level decomposition (using all 16 filters once) effectively captures both coarse and fine texture details. As you can see implementation wise from Fig. 2, The implementation uses nested loops (lines 5–6) to iterate over the number of wavelet levels, convolving the normalized iris with each filter using conv2 (line 8). The 'valid' mode ensures only fully overlapping regions are processed, avoiding edge artifacts. Instead of using the full convolution output, the code computes four statistical features mean, standard deviation, skewness, and kurtosis of the filtered response (lines 10–11). This reduces dimensionality and captures the essential distributional characteristics of the texture in each sub-band. The resulting statistics are appended to a growing feature vector (line 12), forming a compact representation of the iris image. This method balances accuracy and efficiency by combining Chebyshev wavelet filtering with statistical summarization, making it well suited for biometric classification tasks.

```
1   % Optimized 2D Chebyshev Feature Extraction
2           features = [];
3
4           % Use pre-computed filters for faster processing
5           for m = 1:waveletLevels
6               for n = 1:waveletLevels
7                   % Apply pre-computed 2D filter
8                   coeff = conv2(normalizedIris, chebyFilters{m,n}, 'valid');
9
10                  % Extract statistical features instead of all pixels
11                  featStats = [mean(coeff(:)), std(coeff(:)), ...
12                              skewness(coeff(:)), kurtosis(coeff(:))];
13                  features = [features; featStats'];
14              end
15          end
```

Fig. 2 *Chebyshev wavelet filtering and feature extraction code*

3.4 Classification

Once the feature vectors are obtained, classification algorithms specifically the Support Vector Machines (SVM) is used to classify them. SVM has a high performance in binary classification task which makes it very suitable for matching iris patterns. Also to optimize the performance of the SVM algorithm Parameter tuning can be conducted.

The extracted 64-dimensional feature vectors are used for classification. We employ a Support Vector Machine (SVM) classifier, which is trained on labelled iris features from the training set. During testing, each new iris image is preprocessed and converted to features as above, then fed to the trained SVM. The SVM outputs the identity class of the closest match. Performance is evaluated by comparing predicted identities to ground truth, computing metrics such as Accuracy, FAR, and FRR.

The same pipeline (preprocessing, feature extraction, classification) is applied for both the proposed Chebyshev method and a baseline system that uses a traditional Gabor filter bank for feature extraction. This allows a direct comparison of performance between methods [6].

3.5 Evaluation

This phase Evaluates the system's performance metrics using accuracy, false Rejection Rate (FRR), and False Acceptance Rate (FAR). It creates a detailed report and visualization to help provide a deeper and more comprehensive understanding of the system's strengths and areas for improvement. The evaluation module also allows for a comparison with already existing systems such as Gabor filter.

4. Experimental Setup

This section describes the experimental setup we used to implement and evaluate the proposed Feature extraction method. It describes techniques, tools, and procedures that we implemented to conduct the experiment.

The Experiment was conducted using MATLAB a high-level programming and interactive platform that was developed for numerical computations, data analysis, visualization, and algorithm development. It was chosen because of its capabilities in handling mathematical computations, image processing and machine learning. It provides an integrated development environment (IDE) and large libraries that streamline the implementation and evaluation of the study. MATLAB allows users to scale analysis to larger datasets or even run computations on clusters or cloud environments with minimal changes to the code. That is why MATLAB was used in this research project. Table 2 shows the hardware and software that was used for the experiment.

We evaluate the proposed method on two publicly available iris datasets: CASIA-Iris V4 and MMU. The CASIA database contains tens of thousands of near-infrared iris images from many subjects, offering a wide variety of real-world variations but we used a selected few. The MMU database includes high-quality iris images captured under controlled conditions. Using both datasets ensures that our method was tested across different qualities and populations.

For each dataset, we randomly split the images into training and testing sets (e.g., 70% for training, 30% for testing). During preprocessing, each image is segmented and normalized as described. Then we extract features using the Chebyshev filters. The same procedure is applied with Gabor filters for the baseline comparison.

The SVM classifier uses a radial basis function (RBF) kernel, and its hyperparameters are tuned via cross-validation on the training data. We use accuracy (the percentage of correct identifications) as the primary performance metric. Additionally, we compute the False Acceptance Rate (FAR) and False Rejection Rate (FRR) from the confusion matrix. Specifically, FAR measures the proportion of impostor attempts incorrectly accepted, while FRR measures the proportion of genuine attempts incorrectly rejected. Lower FAR/FRR values indicate better system security and reliability. All experiments are implemented in MATLAB on a standard PC platform. The workflow is logged to ensure reproducibility any visual outputs such as example filter responses can be saved for analysis.

Table 2 *Experimental Setup*

Software	Hardware
MATLAB Dataset: CasiaV4-Interval, MMU	Laptop CPU: AMD RYZEN 7 5700U with Radeon Graphics RAM: 16 GB

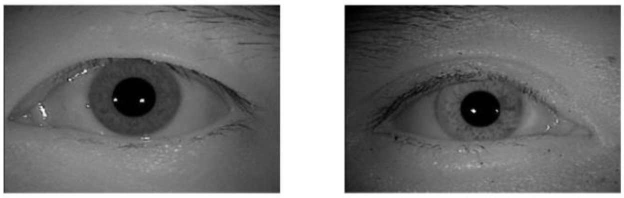
4.1 Simulation Setup

It has already been mentioned that the primary platform that was used for the experiment is MATLAB. In order to complete the experiment some essential toolboxes must be incorporated like the image processing toolbox for preprocessing and the wavelet toolbox for feature extraction, and the statical and machine learning toolbox for classification. So, when all the toolboxes were incorporated and ready the datasets that we used for the experiment were preloaded and organized and then the experiment began.

4.2 Dataset Preparation

The datasets that we used for the experiments are CASIA and MMU. They were prepared for experimentation by dividing them into training and testing sets. The training set we used to train the classification model that was utilized after extracting features while the testing set was reserved for evaluation. Each dataset undergoes the preprocessing steps outlined in the previous section.

The CASIA dataset contains 54,607 iris images from over 1,800 actual individuals and 1,00 virtual subjects. The dataset comprises of all the captured images iris database released to the international biometrics community and updates from CASIA- Iris V1 in 2002 up to the latest CASIA- Iris V4 datasets. All Iris images are 8- bit grey level Bitmap Image files (BMP). Following the date development of the CASIA database, the Chinese Academy of Science has created the following database versions. The CASIA- Iris V4 dataset comprises iris images obtained with human subject constraints and includes contributions dedicated to reducing conditions, Fig. 2(a) Shows sample iris image from CASIA-IRIS V4. The images in the MMU database were collected from 100 volunteers of varying ages and ethnicities with everyone contributing five photographs from each eye. Also, the iris images in the MMU database are uniform and were collected at a close distance using a NIR light source with human subject participation which results in eyelash obstruction and circular rotation in the images Fig. 2(b) shows sample figure of iris image from MMU Database.



(a)

(b)

Fig. 3 *Example of Iris image from multiple Datasets (a) Iris from CASIA V4; (b) From MMU*

4.2.1 Module Implementation

The system is modularly implemented using MATLAB, with each processing stage handled by a dedicated script. Below is a summary of the key modules: Preprocessing: Raw iris images (RGB or grayscale) are first converted to grayscale. Denoising is performed using a median filter, and local contrast is enhanced via adaptive histogram equalization. These steps clarify iris textures and standardize input quality for feature extraction. The process is part of a full preprocessing pipeline for the project, as seen in Fig. 4. It begins with a workspace cleanup using clc; clear; close all; (line 2). Lines 5–9 set up the project paths, specifying the root directory and defining paths for raw data (dataPath) and pre-processed output (preprocessedPath). The script supports two datasets: 'CASIA' and 'MMU', with acceptable image formats defined in line 9. In lines 11–15, the code checks for the existence of subdirectories for each dataset inside the pre-processed folder and creates them if they do not exist. This ensures the output structure is correctly initialized before processing begins.

```
1  % ==== Iris Recognition Project: Full Restart ====
2  clc; clear; close all;
3
4  % === Project Path Setup ===
5  projectRoot = 'C:\MATLAB\iris_project';
6  dataPath = fullfile(projectRoot, 'data');
7  preprocessedPath = fullfile(projectRoot, 'preprocessed');
8  datasets = {'CASIA', 'MMU'};
9  formats = {'*.jpg', '*.bmp'};
10
11
12 for d = 1:length(datasets)
13     outDir = fullfile(preprocessedPath, datasets{d});
14     if ~exist(outDir, 'dir'); mkdir(outDir); end
15 end
16
17 % === Processing Loop ===
18 for d = 1:length(datasets)
19     folder = fullfile(dataPath, datasets{d});
20     images = dir(fullfile(folder, formats{d}));
21
22     for i = 1:length(images)
23         % === Load Image ===
24         imgPath = fullfile(folder, images(i).name);
25         img = imread(imgPath);
26
27         % === Grayscale Conversion ===
28         if size(img, 3) == 3
29             grayImg = rgb2gray(img);
30         else
31             grayImg = img;
32         end
33
34         % === Preprocessing ===
35         denoised = medfilt2(grayImg);
36         enhanced = adapthisteq(denoised);
37
38         outName = fullfile(preprocessedPath, datasets{d}, images(i).name);
39         imwrite(enhanced, outName);
40
41         fprintf('Processed: %s\n', images(i).name);
42     end
43 end
44
45 disp(' Preprocessing complete.');
```

Fig. 4 *Preprocessing code snippet*

The main processing loop begins on line 18. For each dataset, it loads all images matching the specified formats (line 20). Within the inner loop (line 22), each image is read (lines 24–25). Then, grayscale conversion is handled in lines 27–31. If the image has three channels (i.e., it is RGB), it is converted to grayscale using rgb2gray; otherwise, it's already grayscale and passed through unchanged.

Lines 33–35 apply preprocessing: a median filter (medfilt2) to remove salt-and-pepper noise and adaptive histogram equalization (adapthisteq) to improve local contrast. In line 37, the enhanced image is saved to the corresponding output directory, and line 39 prints the processed filename to the console. The script completes processing each dataset and displays "Preprocessing complete." at the end (line 42). This process prepares the data for segmentation.

Segmentation**:** The iris and pupil boundaries are located using the imfindcircles function, which applies to a circular Hough Transform to detect the pupil (dark) and iris (brighter outer circle). Detected circle parameters are saved for reuse and visually verified using viscircles. And if you will check Fig 5 there is a sample of what a detected iris and pupil look like.

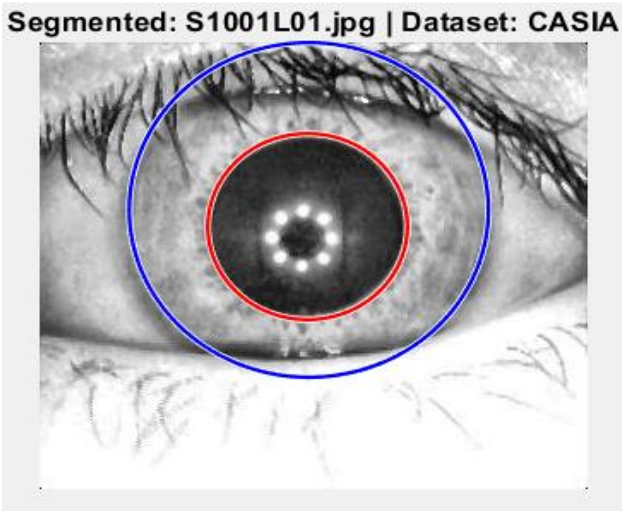


Fig. 5 *Segmentation sample*

Chebyshev Feature Extraction: The segmented iris is normalized using Daugman's rubber-sheet model, converting the circular region to a fixed-size rectangular grid as shown in Fig. 6. A 2D Chebyshev filter bank is applied at four pyramid levels to extract multi-scale texture features. Statistical descriptors such as energy and variance are computed from filtered images. Dimensionality is reduced using Principal Component Analysis (PCA), retaining the top 50 components.

Fig 6 *Normalization sample output*

Gabor Feature Extraction: A parallel feature set is computed using a bank of 12 Gabor filters across various orientations and scales. The magnitude responses are concatenated and reduced using PCA to extract the most informative components for classification.

Classification: The final classification stage uses a Support Vector Machine (SVM) trained on the reduced feature vectors. Multiple kernels (linear, RBF, polynomial) are evaluated, with hyperparameters optimized using MATLAB's fitcsvm. Performance is measured using accuracy, FAR, FRR, and Equal Error Rate (EER). Confusion matrices and ROC curves are generated to visualize classifier performance. An ensemble SVM strategy is optionally implemented to enhance robustness.

4.3 Experimental Design

Before experimenting with the three datasets which are CASIA, MMU, and UBIRIS datasets, it is necessary to construct the experiment stages to ensure that the experiment succeeds. By following the workflow, if the experiment should provide a failed result, we can be able to easily Identify which component of the process needs to be modified or re-tested from the start. Fig. 7 illustrates a flowchart of the experiment phases that the process needs to go through to complete the experiment.

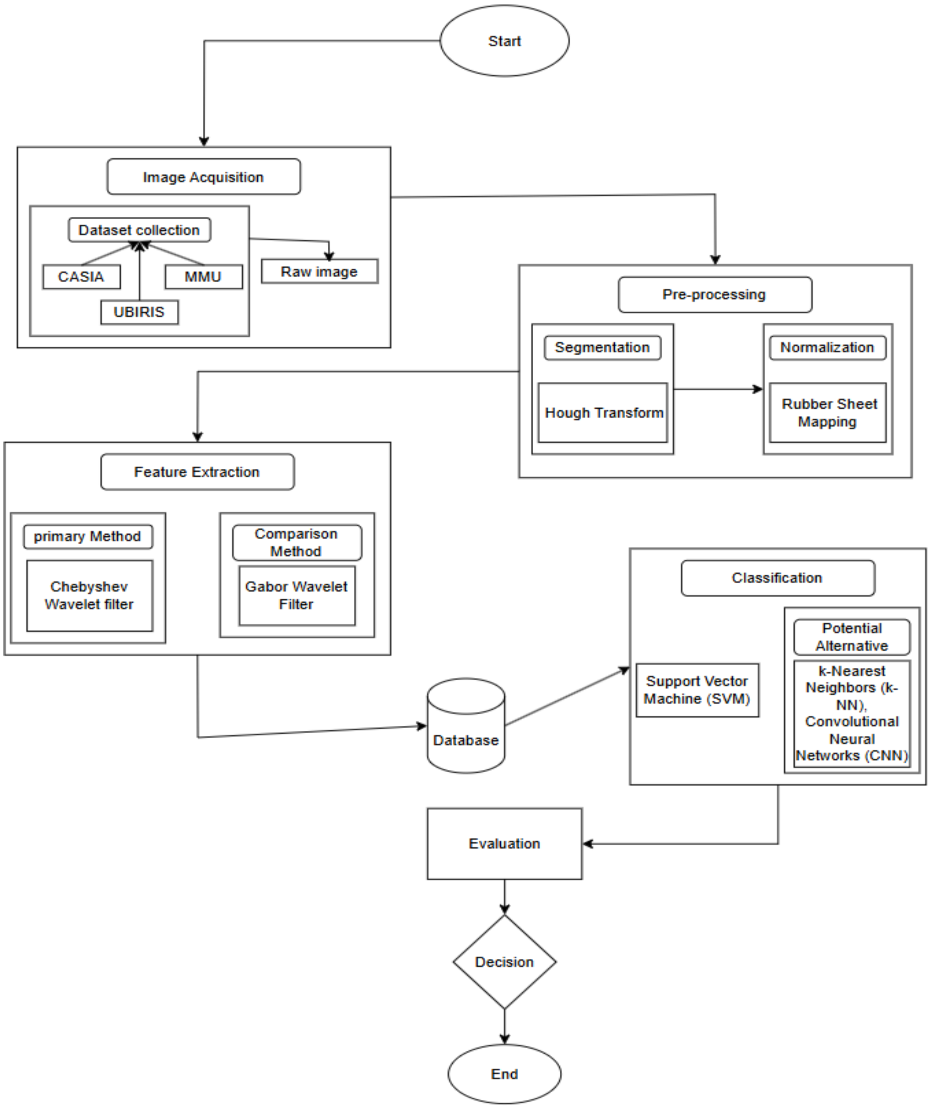


Fig 7 *Flowchart showing Phases of the research framework.*

4.4 Parameter and Testing Methods

Parameters are models used to predict the method. Testing method refers to the evaluation used to support the result produced from each technique or process primarily through evaluation metrics derived from confusion metrics. The method used uses confusion metrics that can show the accuracy, the false acceptance rate and the false rejection rate of the experiment. Confusion matrix provides a systematic approach to analyze the effectiveness of feature extraction. Different metrics are calculated to measure the validity of the model. As shown in Table 3 the system will be evaluated with True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN). from the metrics mentioned which are accuracy, False Rejection Rate (FRR), False Acceptance Rate (FAR).

Accuracy reflects the overall proportion of correct classifications. FAR measures the rate at which unauthorized individuals are incorrectly accepted, while FRR evaluates how often legitimate users are mistakenly rejected. These metrics provide a quantitative basis for comparing the Chebyshev wavelet-based method against traditional techniques like Gabor filtering, offering insight into both precision and system reliability.

Table 3 *Evaluation Metrics*

Metric	Definition	Formula
True positive (TP)	The model correctly predicted a positive outcome	*
True Negative (TN)	The model correctly predicted a negative outcome	*
False Positive (FP)	The model incorrectly predicted a positive outcome	*
False Negative (FN)	The model incorrectly predicted a negative outcome	*
Accuracy (%)	Accuracy is the fraction of the total samples that are correctly classified by the classifier. It gives the overall accuracy of the model	$\frac{TP+TN}{TP+FP+TN+FN}$
False Rejection Rate (FRR)	The ratio of the number of false rejections divided by the total number of transactions	$\frac{FN}{TP+FN}$
False Acceptance Rate (FAR)	The measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user	$\frac{FP}{FP+TN}$

4.5 Implementation steps And Algorithm Used

After collecting iris images from the datasets, the raw images go to preprocessing phase and it goes through segmentation where Hough Transform is used to isolate the iris region from the sclera and pupil by detecting circular boundaries for the iris. The output will then be a segmented Iris image that contains only the relevant region. And after that it gets Normalized using Daugman's rubber sheet model which standardizes iris size and shape to handle variations in the pupil dilation. The output will be Normalized images with uniform dimensions. The Normalized images go to Feature extraction phase where Chebyshev wavelet Filter is used to decompose iris images into different resolution levels, capturing both high frequency and low frequency components due to the mathematical properties of Chebyshev polynomials. To validate the effectiveness of the Chebyshev Wavelet Filter other feature extraction methods like the Gabor wavelet filter will be implemented for comparison. Numerical Vectors representing the unique texture and pattern of iris that were extracted are stored in the database for classification. In the classification phase the extracted features are analyzed and categorized to determine an identity. In this step we Train and validate models using Support Vector Machine (SVM). The workflow for this stage is divided into three phases which are Training phase, Testing Phase, and Evaluation.

4.5.1 Chebyshev wavelet Design

i. Filter Design: Build a bank of 16 two-dimensional filters by selecting Chebyshev polynomials of orders 1–4 in both horizontal and vertical directions. Each filter is a 32×32 kernel derived from evaluating Tn(x)=cos(narccos x) over [−1,1].
ii. Single-Level Decomposition**:** Convolve the normalized iris image with all 16 Chebyshev filters no spectral complements or multi-stage tree are used, for simplicity and speed.
iii. Feature Computation: From each filtered image, compute four statistics (mean, standard deviation, skewness, kurtosis), yielding a 64-dimensional feature vector.
iv. Dimensionality Reduction: Apply PCA to the 64-D vector to retain the most informative components.
v. Rationale: Captures both coarse and fine iris textures across multiple spatial frequencies and orientations, leveraging Chebyshev filters' strong frequency selectivity and denoising capabilities demonstrated in prior image-processing studies.

4.5.2 Comparative Method

i. Filter Bank Construction: Generate a set of 12 two-dimensional Gabor filters covering three scales (wavelengths λ = 4, 8, 16) and four orientations (θ = 0°, 45°, 90°, 135°).
ii. Convolution: Apply each Gabor filter to the normalized iris image and compute the magnitude of the complex response.
iii. Feature Vector Formation: Flatten and concatenate all magnitude responses into a single high-dimensional vector (one slice per filter).
iv. Dimensionality Reduction (Optional): Pool or apply PCA to reduce redundancy and retain the most discriminative components.
v. Rationale: This well-established approach captures orientation- and frequency-specific texture patterns in the iris and serves as a direct benchmark for evaluating the proposed Chebyshev-wavelet features.

Applied Information Technology and Computer Science Vol. 6 No. 2 (2025) p. 698-712

4.6 Results and Discussion

The proposed Chebyshev-wavelet feature extractor was compared against the Gabor-based baseline. Table 4 and Table 5 summarize the performance results. The Chebyshev method consistently outperforms the traditional approach on both datasets. For example, on the CASIA dataset, the proposed method achieved higher overall accuracy and lower error rates. Similar trends were observed on the MMU dataset and for the implementation that led to that result the project went through a pipeline involving preprocessing, segmentation, normalization, feature extraction, and classification just as described in the methodology.

How it was implemented was, both the Chebyshev wavelet and Gabor wavelet feature extraction methods shared the same preprocessing and normalization steps. The images were converted to grayscale, denoised using a median filter, enhanced with adaptive histogram equalization, and segmented using circular Hough transforms for pupil and iris detection. The normalized iris region was unwrapped using a rubber-sheet model with occlusion masking.

For Chebyshev wavelet features, a bank of 16 2D Chebyshev polynomial filters (4×4) were applied to each normalized image using conv2, and each filter response was summarized using four statistical features: mean, standard deviation, skewness, and kurtosis. This resulted in a 64-dimensional feature vector, which was then reduced using PCA to the top 50 components for classification.

For the Gabor-based method, 12 filters were generated (3 scales × 4 orientations), and the magnitude of the responses was concatenated into high-dimensional vectors, also reduced via PCA. Pairwise matching was performed for both methods using a matching strategy that generated genuine and impostor pairs. Balanced datasets were created using SMOTE to mitigate class imbalance. Final classification used a support vector machine (SVM), with hyperparameter tuning performed using grid search over kernel type (linear/RBF), box constraint, and kernel scale. The best-performing configuration for each method-dataset pair was selected based on 5-fold cross-validation.

Thresholds for decision-making were fixed per method-dataset combination. Using these thresholds, True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) values were computed, and from these:

- Accuracy = (TP + TN) / Total
- FAR = FP / (FP + TN)
- FRR = FN / (FN + TP)

Table 4 *Accuracy comparison*

Method	CASIA Accuracy (%)	MMU Accuracy (%)
Chebyshev wavelet	98.62	97.71
Gabor Wavelet	92.75	97.10

Table 5 *FAR/FRR comparison*

Method	CASIA FAR (%)	CASIA FRR (%)	MMU FAR (%)	MMU FRR (%)
Chebyshev Wavelet	3.08	3.85	2.61	1.96
Gabor Wavelet	7.63	6.87	2.58	3.23

As indicated in the tables showing the results of my experiment, the Chebyshev-based method gave a more higher recognition accuracy than the Gabor baseline in each case. The Chebyshev filters' sharper frequency selectivity appears to better capture the distinctive iris texture, making the features more discriminative. Moreover, the error rates (FAR and FRR) are lower for the proposed method, showing improved robustness. Like where the gabor had occasional false accepts due to noise or occlusion, the Chebyshev method reduced these errors by focusing on cleaner signal components.

Table 4 and 5 summarizes the results for both the Chebyshev-wavelet and Gabor-based feature extractors on the CASIA and MMU iris datasets as the results visuals is shown in Fig. 8(a). On CASIA, the Chebyshev method achieved 96.54 % accuracy, with a false-acceptance rate (FAR) of 3.08 %, false-rejection rate (FRR) of 3.85 %, and equal-error rate (EER) of 3.46 % (decision threshold = 0.551). The Gabor baseline reached only 91.98 % accuracy with much higher error rates (FAR = 8.40 %, FRR = 7.63 %, EER = 8.02 %, threshold = 0.333) as shown in fig. 8(c). On the more variable MMU set, both methods approach high accuracy, but Chebyshev still leads slightly 97.71 % in Fig. 8(b)vs. Fig.8(d) 97.10 % and gave lower EER (2.29 % vs. 2.90 %) and FRR (1.96 % vs. 3.23 %).

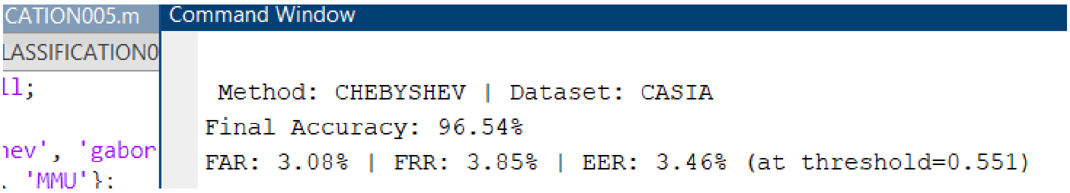


Fig. 8(a) *Result of the CASIA classification on Chebyshev*

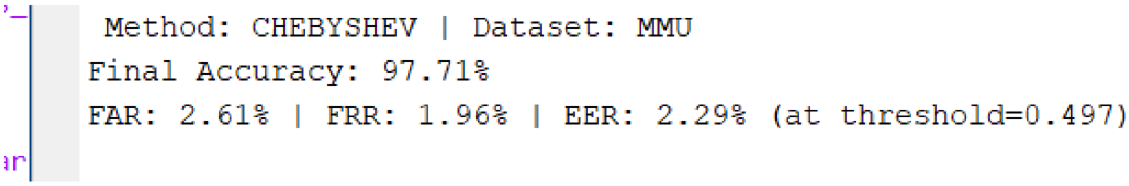


Fig. 8(b) *Result of the MMU classification on Chebyshev*

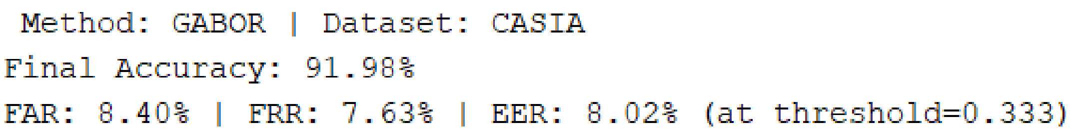


Fig. 8(c) *Result of the CASIA classification on Gabor*

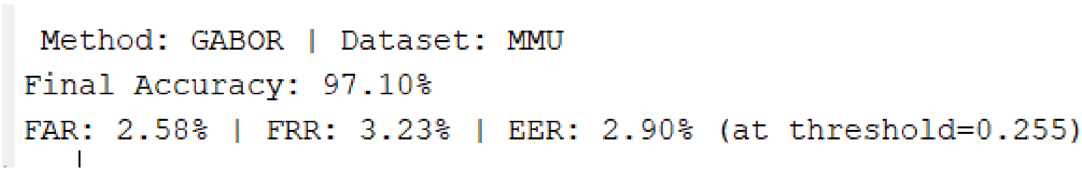


Fig. 8(d) *Result of the MMU classification on Gabor*

In terms of computational efficiency, the Chebyshev filter bank requires only a one-level decomposition (all filters applied once) and simple statistical computations. This is comparable to the cost of applying a multi-scale Gabor bank. Notably, Chebyshev filters can be implemented efficiently by precomputing their polynomial coefficients, as noted in other work. In summary, the results confirm that the Chebyshev-wavelet approach can achieve superior accuracy without incurring extra computational burden.

These findings illustrate the potential of Chebyshev wavelet filters in iris recognition. While other methods can also attain high accuracy, they require extensive data especially for models like deep learning which require high training data and resources. The Chebyshev approach offers a middle ground, a relatively simple classical method that still significantly improves performance under realistic conditions.

5. Conclusion

This research introduces an enhanced feature extraction method using Chebyshev wavelet filter to address limitations in Iris recognition systems. From the findings of the study we know that Existing systems face challenges in non-ideal conditions because of noise, occlusions, and variations which affects their accuracy. While Gabor filters and wavelets transforms are widely used, they can struggle with complex iris images in non- ideal conditions. This proposed method provided improved accuracy and robustness in handling variations and noise in non-ideal conditions. At the end of this research, we evaluated a new feature extraction method that enhance iris recognition system. Also, at the end of the research the proposed method was able to achieve a very low False Acceptance Rate and False Rejection Rate that is superior to traditional feature extraction methods. The research was able to overcome challenges related to computational efficiency and adaptability across datasets by leveraging the unique properties of Chebyshev wavelets.

The research shows the potential of Chebyshev wavelet filter in enhancing feature extraction and the overall practicality of iris recognition systems for real world applications. This study compares the performance of the proposed Chebyshev wavelet filter with existing feature extraction techniques like the Gabor filters. Future research could investigate the impact of different parameters within the Chebyshev Wavelet filters or test the proposed method on a wider range of datasets and also implement the proposed system on devices with limited resources or even combining them with other feature modalities. Overall, the study indicates that Chebyshev wavelet filters are a promising tool for improving biometric iris recognition.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

This journal requires that all authors take public responsibility for the content of the work submitted for review. The contributions of all authors must be described in the following manner:

The authors confirm contribution to the paper as follows: ***study conception and design:*** *M. Hayatudeen, S. Jamel;* ***data collection:*** *M. Hayatudeen, S. Jamel;* ***analysis and interpretation of results:*** *M. Hayatudeen, S. Jamel;* ***draft manuscript preparation:*** *M. Hayatudeen, S. Jamel. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] A. Mukherjee, M. Z. Islam, G. M. Mamun-Al-Imran, and L. E. Ali, “Iris Recognition Using Wavelet Features and Various Distance Based Classification,” in *Proceedings of International Conference on Electronics, Communications and Information Technology, ICECIT 2021*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/ICECIT54077.2021.9641118.

[2] R. Al-Zubi, K. Darabkh, and Y. Jararweh, “A powerful yet efficient iris recognition based on local binary quantization,” *Information Technology and Control*, vol. 43, no. 3, 2014, doi: 10.5755/j01.itc.43.3.5225.

[3] N. H. M. KAMIL and N. A. B. M. Zin, “An Improved Iris Segmentation using Pupil Candidate Bank for Non-Cooperative Iris Recognition,” *Applied Information Technology And Computer Science*, vol. 4, no. 1, pp. 1281–1295, Jul. 2023, Accessed: Nov. 13, 2024. [Online]. Available: https://publisher.uthm.edu.my/periodicals/index.php/aitcs/article/view/7394

[4] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, “A survey on missing data in machine learning,” *J Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00516-9.

[5] J. R. Malgheet, N. B. Manshor, and L. S. Affendey, “Iris Recognition Development Techniques: A Comprehensive Review,” *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/6641247.

[6] G. Liu, W. Zhou, L. Tian, W. Liu, Y. Liu, and H. Xu, “An efficient and accurate iris recognition algorithm based on a novel condensed 2-ch deep convolutional neural network,” Sensors, vol. 21, no. 11, 2021, doi: 10.3390/s21113721.

[7] S. Davuluri, S. Kilaru, V. Boppana, M. Vekateswara Rao, K. N. Rao, and S. S. Vellela, “A Novel Approach to Human Iris Recognition and Verification Framework Using Machine Learning Algorithm,” Proceedings of International Conference on Contemporary Computing and Informatics, IC3I 2023, pp. 2447–2453, 2023, doi: 10.1109/IC3I59117.2023.10397886.

[8] H. M. Ahmed and M. A. Taha, “A Brief Survey on Modern Iris Feature Extraction Methods,” Engineering and Technology Journal, vol. 39, no. 1A, 2021, doi: 10.30684/etj.v39i1a.1680.

[9] Y. Chen et al., “Accurate iris segmentation and recognition using an end-to-end unified framework based on MADNet and DSANet,” Neurocomputing, vol. 517, pp. 264–278, Jan. 2023, doi: 10.1016/j.neucom.2022.10.064.

[10] R. D. Sah and S. J. Abbas, “Comparative Analysis on Different Wavelet Families to Improvise Iris Recognition System,” in 2022 3rd International Conference on Issues and Challenges in Intelligent Computing Techniques, ICICT 2022, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICICT55121.2022.10064581.

[11] O. Aiyeniko, Y. Adekunle, M. Eze, O. A.-Glob. J. Artif. Intell, and undefined 2020, “Performance analysis of feature extraction and its fusion techniques for iris recognition system,” researchgate.netO Aiyeniko, YA Adekunle, MO Eze, OD AlaoGlob. J. Artif. Intell, 2020•researchgate.net, Accessed: Dec. 14, 2024. [Online]. Available: https://www.researchgate.net/profile/Oluwaseun-Ebiesuwa/publication/361494924_GJAI-2020-03-1105/links/62b4bb85dc817901fc76250d/GJAI-2020-03-1105.pdf

[12] R. Szewczyk, K. Grabowski, M. Napieralska, W. Sankowski, M. Zubert, and A. Napieralski, “A reliable iris recognition algorithm based on reverse biorthogonal wavelet transform,” Pattern Recognit Lett, vol. 33, no. 8, pp. 1019–1026, Jun. 2012, doi: 10.1016/j.patrec.2011.08.018.

[13] A. Ullah, A. Salam, H. El Raoui, D. Sebai, and M. Rafie, "Towards more accurate iris recognition system by using hybrid approach for feature extraction along with classifier," International Journal of Reconfigurable and Embedded Systems, vol. 11, no. 1, pp. 59–70, Mar. 2022, doi: 10.11591/IJRES.V11.I1.PP59-70.
[14] H. K. Nigam and H. M. Srivastava, "Filtering of Audio Signals Using Discrete Wavelet Transforms," Mathematics, vol. 11, no. 19, Oct. 2023, doi: 10.3390/math11194117.
[15] R. J. Cintra, H. M. De Oliveira, and L. R. Soares, "On Filter Banks and Wavelets Based on Chebyshev Polynomials," 2014.
[16] M. Balubaid, M. A. Sattari, O. Taylan, A. A. Bakhsh, and E. Nazemi, "Applications of discrete wavelet transform for feature extraction to increase the accuracy of monitoring systems of liquid petroleum products," Mathematics, vol. 9, no. 24, Dec. 2021, doi: 10.3390/math9243215.
[17] J. Zhang, J. Du, X. Xi, and Z. Yang, "Chebyshev Chaotic Mapping and DWT-SVD-Based Dual Watermarking Scheme for Copyright and Integrity Authentication of Remote Sensing Images," Symmetry (Basel), vol. 16, no. 8, Aug. 2024, doi: 10.3390/sym16080969.
[18] A. A. Abdulrahman and F. S. Tahir, "Face recognition using enhancement discrete wavelet transform based on MATLAB," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 2, pp. 1128–1136, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp1128-1136.
[19] A. A. Abdulrahman, M. Rasheed, and S. Shihab, "The Analytic of Image Processing Smoothing Spaces Using Wavelet," in Journal of Physics: Conference Series, IOP Publishing Ltd, May 2021. doi: 10.1088/1742-6596/1879/2/022118.
[20] H. K. Nigam, R. N. Mohapatra, and K. Murari, "Wavelet approximation of a function using Chebyshev wavelets," J Inequal Appl, vol. 2020, no. 1, pp. 1–14, Jul. 2020, doi: 10.1186/S13660-020-02453-2/METRICS.
[21] N. Kumar and M. Mehra, "Legendre wavelet collocation method for fractional optimal control problems with fractional Bolza cost," Numer Methods Partial Differ Equ, vol. 37, no. 2, pp. 1693–1724, Mar. 2021, doi: 10.1002/NUM.22604.
[22] "2D-Discrete Wavelet Transformation and its applications in Digital Image Processing using MATLAB | by Koushik Chandrasekaran | Medium." Accessed: Jun. 13, 2025. [Online]. Available: https://medium.com/%40koushikc2000/2d-discrete-wavelet-transformation-and-its-applications-in-digital-image-processing-using-matlab-1f5c68672de3
[23] S. Hosseini et al., "Application of wavelet feature extraction and artificial neural networks for improving the performance of gas–liquid two-phase flow meters used in oil and petrochemical industries," *Polymers (Basel)*, vol. 13, no. 21, Nov. 2021, doi: 10.3390/polym13213647.
[24] E. Guariglia and R. C. Guido, "Chebyshev Wavelet Analysis," Journal of Function Spaces, vol. 2022, 2022, doi: 10.1155/2022/5542054.
[25] S. Das and B. Kar, "Fusion of Wavelet Features and Gabor Features for SVM-based Iris Verification," International Journal of Experimental Research and Review, vol. 43, pp. 134–145, 2024, doi: 10.52756/ijerr.2024.v43spl.010.
[26] A. Kazemi Nasab, A. Klçman, Z. Pashazadeh Atabakan, and S. Abbasbandy, "Chebyshev Wavelet Finite Difference Method: A New Approach for Solving Initial and Boundary Value Problems of Fractional Order," Abstract and Applied Analysis, vol. 2013, no. 1, p. 916456, Jan. 2013, doi: 10.1155/2013/916456.
[27] Y. Yin, S. He, R. Zhang, H. Chang, X. Han, and J. Zhang, "Deep Learning for Iris Recognition: A Review," Mar. 2023, Accessed: Jun. 13, 2025. [Online]. Available: https://arxiv.org/pdf/2303.08514
[28] J. X. Shi and X. F. Gu, "The comparison of iris recognition using principal component analysis, independent component analysis and Gabor wavelets," Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010, vol. 1, pp. 61–64, 2010, doi: 10.1109/ICCSIT.2010.5563947.
[29] H. Ali and M. J. E. Salami, "Iris Recognition System Using Support Vector Machines," Biometric Systems, Design and Applications, Oct. 2011, doi: 10.5772/17730.
[30] R. J. Cintra, H. M. De Oliveira, and L. R. Soares, "On Filter Banks and Wavelets Based on Chebyshev Polynomials," 2014.
[31] H. M. Muhi-Aldeen et al., "Improvement of Color Image Analysis Using a New Hybrid Face Recognition Algorithm based on Discrete Wavelets and Chebyshev Polynomials," ArXiv, p. arXiv:2303.13158, Mar. 2023, doi: 10.48550/ARXIV.2303.13158.
[32] A. A. Abdulrahman and F. S. Tahir, "Face recognition using enhancement discrete wavelet transform based on MATLAB," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 2, pp. 1128–1136, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp1128-1136.