



Low-Complexity Pupil and Iris Detection for Enhanced IoT-Driven Biometric Systems

Nada I. Najim^{1*}, Dawlat Mustafa Sulaiman², Suha M. Noori³

¹ Computer Engineering Department, College of Engineering, University of Mosul, Mosul 00964, Iraq

² Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Duhok 00964, Iraq

³ Department of Soil & Water Management, Agriculture, Salahaddin University, Erbil 00964, Iraq

Corresponding Author Email: nada.ismail@uomosul.edu.iq

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.590801>

ABSTRACT

Received: 11 May 2026

Revised: 3 July 2026

Accepted: 11 August 2026

Available online: 31 August 2026

Keywords:

biometric systems, edge devices, Internet of Things systems, pupil detection, iris segmentation

Internet of Things (IoT) environments benefit from heightened urgency to deploy lightweight, accurate real-time iris detection algorithms because of their expanded biometric system integration. This research introduces a minimalist approach for pupil and iris detection, evaluated via MATLAB simulations to assess its suitability for biometric systems under IoT management. The method addresses multiple objectives related to performance, memory usage, and computation speed, which fit the requirements of edge devices with limited resources. The system combines adaptive pixel-intensity classification with heuristic, position-based boundary tracing algorithms to provide accurate detection results at reduced memory consumption rather than using conventional machine learning (ML) models. Pixel intensity, along with position-based segmentation analysis, is used to improve pupil and iris localization by examining dynamic RGB channel pixel features across the image. Experimental tests prove the system performs optimally in various lighting environments alongside different image types, and it requires low computational complexity. The proposed detection framework can serve as an efficient front-end module for iris recognition and biometric authentication systems in future IoT applications.

1. INTRODUCTION

The incorporation of modern security and authentication systems with biometric technology became essential because these systems deliver individual-specific identifiers that cannot be replicated [1]. The security realm seeks two specific biometric modalities, namely pupil detection and iris recognition, since they combine exceptional accuracy with protection against forgery, as demonstrated by Ghosh et al. [2]. The deployment of strong biometric methods has become essential because personal and organizational data remain at growing risk in modern times [3, 4]. The widespread growth of Internet of Things (IoT) devices has expanded smart technologies throughout healthcare settings and educational institutions as well as surveillance networks and workplaces [5, 6]. Multiple IoT devices function with restricted capabilities because they possess constrained processing power as well as limited memory resources and battery capacity [7, 8]. These limitations challenge the implementation of high-complexity biometric algorithms on edge devices. The market requires efficient lightweight biometric methods that deliver high-accuracy performance alongside operational capability in IoT networks [1, 9]. The implementation of IoT systems faces challenges when combining advanced biometric algorithms since these forms of verification consume significant processing power and require substantial energy input. High-precision biometrics

require proper balancing with hardware efficiency because real-time implementation on embedded systems containing low-power elements presents a major technical challenge [10, 11]. Many current solutions struggle to dynamically respond to IoT networks composed of multiple system types, which causes difficulties during widespread IoT implementation [12, 13]. This research develops an innovative, low-complexity iris and pupil detection method designed for future integration as an efficient front-end module in IoT-based biometric systems. The main goal is to develop an algorithm that detects reliably, yet places reduced computational load on edge devices' resources. The system achieves its goal by creating an effective detection mechanism that performs quick operations at lower system cost without affecting fundamental accuracy levels.

This research offers multiple important outcomes through its investigation.

- The research produced a fast pupil and iris detection algorithm that suits embedded IoT platforms.

- This research discusses a potential integration method that delivers biometric processing operations in IoT systems that operate with limited resources [14, 15].

- A potential application scenario is discussed, including smart workplaces alongside healthcare monitoring systems [16, 17].

- Analysis concerning the scalability and customization aspects of edge-intelligent devices for biometric

authentication systems [18, 19].

The proposed framework uses novel innovations to create a more secure IoT biometric system by dealing with the simultaneously occurring challenges of accuracy and device-level feasibility.

2. LITERATURE REVIEW

The IoT advances different domains of society, including healthcare, education, workplace risks, and urban development systems. Two research papers [8, 20] demonstrate how IoT-based monitoring systems boost patient life quality and how integrating machine learning (ML) in these systems leads to more intelligent diagnostic capabilities. Nazir et al. [21] examined mobile computing applications for bettering IoT healthcare, alongside Tirandazi et al. [22], who introduced secure fall alert systems for senior citizens. Ghosh et al. [2] expanded the importance of biometric pattern recognition by explaining how it supports the personalized function of IoT healthcare systems.

The paper written by Awad et al. [1] shows how Artificial Intelligence (AI) becomes integral to IoT systems by delivering AI-powered biometric methods for IoT security. Jagatheesaperumal et al. [12] expanded the work of Explainable AI (XAI) models across IoT platforms to provide understandable and interpretable features. The Naïve Bayes algorithms used by Gopi et al. [16] in their work on IoT-based surgical training generate customized feedback for personalizing healthcare training. Dou et al. [10] explored Artificial General Intelligence (AGI) in IoT by studying its potential applications but also discuss necessary challenges for successful integration. Security functions as a primary application alongside surveillance. An ML-driven facial recognition platform linked with IoT serves both security functions and attendance recordkeeping at the university level, according to Kumar et al. [23]. Gupta [18] developed future authentication solutions that boost identity authentication in IoT environments. The application of IoT brought major changes to educational institutions. Spaho et al. [24] conducted a thorough review of IoT-enabled personalized learning systems, and KardanMoghaddam et al. [17] demonstrated the changing influence of IoT on contemporary educational systems. The integration between cyber-physical systems and IoT allows Tariq et al. [25] to describe support for remote labs and smart learning environments. Zeeshan and Neittaanmäki [26] explained how smart school systems receive deployment treatment. The integration of IoT brings advantages to workplace monitoring systems. The research by Ahmed et al. [5] features an RFID-secured IoT system that uses environment monitoring to improve application integration capabilities across the workplace. Anjomshoa et al. [9] explained how "social behaviometrics" enables device customization through observable user behaviors for generating adaptive smart spaces.

The public sector feels the same transformative power as industrial facilities do under IoT. Rao et al. [6] carried out a mini review of how IoT transforms various industrial sectors, and then Yadav et al. [15] examined smart parking systems in urban areas using IoT with ML technology. Nwankwo et al. [19] recommend building IoT systems based on Bayesian learning to stop commercial vehicle traffic accidents. Mammadovc and Kucukkulahli [11] presented a design of smart library infrastructure that combines fog-cloud

architecture with IoT environmental monitoring and ML-based optimization. Mane et al. [27] assessed how IoT influences the transformation of library services. Gupta [18] developed mobility solutions through multiple ML approaches that lead to intelligent transportation systems. Researchers are currently studying both sensor-based authentication and biometric authentication methods. Kaviyazhiny et al. [28] conducted an assessment of EEG-based biometric systems, which reveals efficiency performance factors. The facial detection system developed by Zeeshan and Neittaanmäki [26] provides users with unrestricted IoT public health monitoring capabilities enabled by the Firebase database platform. The paper by Asi et al. [3] demonstrates Recurrent Neural Network (RNN) model applications for extracting health information from IoT sensor data in optometric studies. A comprehensive analysis of IoT development in multiple fields appears in references [7, 29], who presents an overview of smart device expansion. Raad [14] established essential definitions about IoT and wearable technology design that new innovations need to build upon. The research by Tirandazi et al. [22] examines the Internet of Everything (IoE) concept alongside its neuromarketing potential for human-centered IoT applications. In their work, Mishra et al. [13] assemble academic studies regarding information and Communication Technology (ICT) competitive strategies that align with IoT-based operational advancements. Khang [30] demonstrated how smart technology utilizes artificial intelligence for medical diagnosis purposes to enhance healthcare quality. Recently, Jiang et al. [31] proposed Segment Anything Model (SAM)-Iris, an enhanced iris segmentation framework based on the SAM, introducing an IrisAdapter, a parallel Convolutional Neural Network (CNN) branch, and a cross-branch attention mechanism to improve segmentation accuracy under challenging imaging conditions. Although the proposed approach achieved excellent segmentation performance, it relies on a large pretrained foundation model and additional network modules, resulting in increased computational complexity that may limit its suitability for resource-constrained embedded devices. Milman et al. [32] adapted SAM for visible-light pupil segmentation using a multi-stage fine-tuning strategy. Their framework demonstrated robust segmentation under varying illumination, eye colors, and occlusions; however, it depends on transformer-based deep learning models and iterative refinement stages, making real-time deployment on lightweight edge platforms computationally demanding. Furthermore, Tato and Kareem [33] presented a comprehensive review of recent ML and deep learning techniques for iris recognition, highlighting that current research increasingly focuses on transformer architectures, foundation models, and hybrid deep learning frameworks. Despite their high recognition accuracy, these approaches generally require considerable computational resources, motivating the development of lightweight and computationally efficient iris segmentation algorithms suitable for embedded and IoT-based vision systems.

Research establishes that the convergence of IoT operates with AI, ML and biometric systems in multiple technological domains. The ongoing development in healthcare along with education and security and public infrastructure, encounters obstacles regarding customization and safety aspects and ethical standards and network compatibility, which requires additional investigation. Table 1 summarizes the most relevant previous studies related to iris localization, lightweight image

processing, or embedded deployment. The comparison highlights their methodologies, contributions, and existing

limitations.

Table 1. Comparison of previous works and current research contribution

Reference	Method	Hardware/Platform	Speed	Memory Demand	Limitations
Othman et al. [34]	Traditional: Hough Transform + Active Contour (V2), Viterbi algorithm (V4), or non-geometric contour parameterization (V4.1).	PC/Open-source software (Central Processing Unit (CPU_based))	15 Frames Per Second (FPS) (Obtained from this study)	Moderate (~85 MB) (Obtained from this study)	Designed as a research baseline rather than a lightweight embedded solution; relies on classical image processing, making segmentation less robust under noisy, unconstrained, or IoT scenarios. No optimization for real-time edge devices is reported.
Wang et al. [35]	Lightweight stacked Hourglass CNN with multi-label segmentation (iris mask + inner/outer boundaries).	Mobile devices	-	Low (lightweight architecture) 0.69 million parameters (2.83 MB storage)	Uses simple least-squares circle fitting for post-processing boundaries, which may limit localization accuracy in highly complex cases.
Sun et al. [36]	Multitask lightweight segmentation network (MSC Block + Two-stage refinement encoder + Grouped Spatial Attention).	Mobile biometric devices (need Graphics Processing Unit GPU)	-	2.2 M parameters	Although lightweight, it still depends on CNN inference and GPU/AI acceleration for optimal performance.
Jiang et al. [31]	Segment Anything Model (SAM) + IrisAdapter + CNN branch + Cross-Branch Attention.	High-performance GPU/Deep learning platform	-	Very High (> 16 GB VRAM for ViT-B)	High computational cost, unsuitable for resource-constrained IoT edge devices, requires interactive prompts (points/boxes). Fails under poor illumination and dark irises (low contrast); reflections cause false positives; iterative stages increase computational cost; unsuitable for resource-constrained IoT edge devices.
Milman et al. [32]	Multi-stage SAM fine-tuning (SAM-BaseIris → SAM-RefinedIris → SAM-RefinedPupil).	GPU-based deep learning platform	-	Very High (> 40 GB during training/inference)	Limited depth of field (~15 cm), relies on histogram analysis for localization (less accurate for irregular boundaries compared to Hough).
Ruiz-Beltrán et al. [37]	YOLOX detector + Lightweight U-Net segmentation embedded on MPSoC.	AMD Ultrascare MPSoC (DPU)	Detection: > 45 FPS Segmentation: 16 FPS	~3.3 million parameters (FPGA resources nearly exhausted)	
Proposed model	Adaptive intensity thresholding + Heuristic sclera-based Region of Interest (ROI) extraction + dynamic pupil isolation (non-geometric, non-deep learning).	MATLAB simulation (target: Raspberry Pi/ESP32/sIoT Edge)	≈42 FPS (Obtained from this study)	Very Low (~31 MB (Obtained from this study)	Currently validated in MATLAB simulation; embedded hardware implementation is future work.

Note: Internet of Things (IoT), field programmable gate array (FPGA), Convolutional Neural Networks (CNN).

3. METHOD

3.1 Theoretical background and system requirements

The identification process through verification and authentication methods relies on physiological traits and behavioral patterns that biometric systems use to function. The

recognition technology that uses irises and pupils provides unique superiority due to their stable characteristics. The integration of these systems within IoT environments creates operational restrictions because they possess low processing speed as well as minimal power reserves and restricted storage capacity.

A set of system requirements exists for biometric systems

that operate within IoT environments.

- Low-latency computation suitable for real-time response.
- Such systems require minimal memory storage capacity because they belong to embedded systems.
- The algorithms must be designed for energy efficiency to extend battery uptime.
- Reliable network protocols with low bandwidth consumption.

3.2 Biometric modalities

Overview of iris and pupil characteristics:

The ring-shaped iris structure, which surrounds the pupil, contains stable, unique patterns during the entire lifespan of a person. The pupil, a dynamic central aperture, serves as a key boundary for iris segmentation.

Important traits:

- Images are captured without any physical intervention to achieve them.
- The patterns within the iris remain unaltered after persons turn two years old.
- A person's identity becomes distinguishable through enough separate distinctive features present in the iris structure.

3.3 Internet of Things device constraints

The proposed algorithm is designed to target constraints of IoT platforms such as Raspberry Pi and ESP32. The algorithm's complexity is intentionally kept low (integer arithmetic, no CNN) to match the capabilities of these devices, which typically include these factors:

- The platforms operate at a frequency of 1.5 GHz with Central Processing Unit (CPU) operations unsupported by Graphics Processing Unit (GPU).
- Memory: As low as 512 MB of Random Access Memory (RAM) or less.
- The device requires a battery with power efficiency under 5 watts.

The following criteria make an algorithm viable for use:

- Time-efficient (low runtime and latency).
- Memory-efficient (low model size and RAM usage).
- Power-efficient (minimal floating-point operations and background activity).

3.4 Performance metrics

The proposed system requires evaluation through three primary metrics.

- Complexity metrics
 - a. Runtime (ms)
 - b. Floating Point Operations (FLOPs)
 - c. Memory Usage (MB)
- Accuracy metrics

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

$$\text{F1-score} = \frac{(2 * \text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} \quad (3)$$

where, TP, FP, and FN denote the numbers of true positives, false positives, and false negatives, respectively.

- Efficiency metrics

- a. Frames Per Second (FPS)
- b. Latency (ms/frame)

3.5 Proposed method

3.5.1 System architecture

Pipeline overview:

The flow of operations in the proposed system starts with image acquisition, then continues to lightweight preprocessing and pupil localization until reaching iris boundary detection, and ends with the extraction of the Region of Interest (ROI), as seen in Figure 1. The proposed system ensures efficiency by designing every process with low complexity requirements for the IoT edge platform.

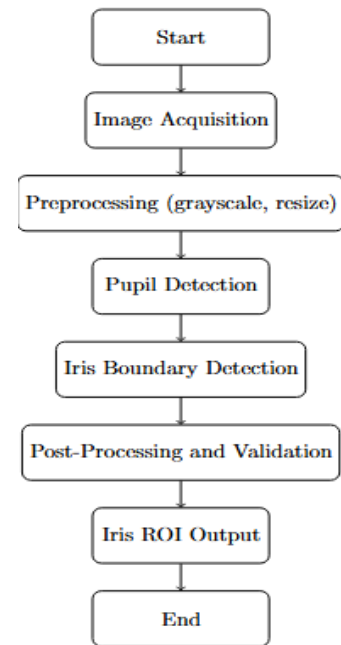


Figure 1. Flowchart of the proposed low-complexity pupil and iris detection system

3.5.2 Pixel intensity/position-based iris segmentation algorithm

Unlike conventional methods that rely on computationally heavy edge detectors, the proposed algorithm adopts a minimalist, single-pass scanning strategy based on dynamic RGB intensity thresholding and heuristic position analysis. The process contains the following sequential stages:

1. Global illumination estimation

The whole 8-bit acquired iris image $I(x, y)$ is to be tested based on pixel value and position. Pixels are classified into three categories according to the deviation between pixel intensities of iris and sclera portions. The categories are classified into high-level intensity pixels, which have pixel intensities ($I(x, y) > 200$), medium-level intensity pixels with ($100 < I(x, y) < 200$), and low-level intensity pixels with ($I(x, y) < 100$).

2. Adaptive brightness normalization

The numbers of the three pixel-level categories are counted individually setting N_{hli} for the number of high-level intensity pixels, N_{mli} for the number of medium-level intensity pixels and N_{lli} for the number of low-level intensity pixels. To

reduce the deviation between original dusky and non-dusky pixels in the whole iris image, the following steps are done:

- If $N_{hli} > \frac{1}{2}$ of the whole size of the iris image,

Then all pixels are multiplied by 0.6 to make them dusky pixels. This multiplication is accomplished to enhance the red (R) component of the whole RGB image. Thus such (R) component is then isolated and stored.

After that isolation of R component, the whole image is then multiplied by 3 for image whitening so that the iris can be distinguished from other parts of the image.

- If $N_{lli} > \frac{2}{3}$ of the whole size of the iris image,

All three components R, G, and B of the low-level intensity pixels are isolated and stored individually. Then all pixels are multiplied by 2.5 for image whitening.

- If $N_{mli} > \frac{1}{2}$ of the whole size of the iris image,

Then all pixels are multiplied by 0.7 to enhance the (R) component of this portion. This (R) component is isolated, stored, and multiplied by 3.1.

•If neither of the above-mentioned conditions appear, then (R) component of such pixels is isolated, stored and multiplied by 4.

3. Sclera masking and bounding box extraction

a. RGB thresholding: A mask is made containing all pixels with R, G, and B components whose level is >100 . That is to identify the whole white portion in the image. The total pixels in the whole image is now classified into two categories (black and white) only.

b. Heuristic scanning: Starting from the left-lower edge of the image a white pixel at the same horizontal level of one of the iris rows is reached. The position of the iris edges in the whole image can be determined according to the position of this white point by investigating all pixels on the left or on the right at the same row. A black line inside the iris is reached. The center of this line is calculated. Then the levels of all pixels at the top and bottom of this center pixel are tested to identify the up and down black points of the iris.

c. A special histogram modification is accomplished by converting most of the image pixels to white color to make it easy to determine the outer boundary of the iris.

d. A square containing the iris is formulated using the positions of the iris border resulting from step b.

e. ROI Extraction: The square area of the previous step is cropped, resulting in selecting the iris portion of the whole image.

4. Noise reduction and boundary tracing

a. Median filtering: The noise in the resulting frame of the previous step is removed using a median filter. Also, the pixels having gray-level more than 200 are converted into black pixels.

b. Fills traced edge points to create solid ROI mask: The iris border points can be chosen as the edge points of the black area. Another black area is chosen and each edge white pixel on the iris boarder of the previous steps is painted there with its original position. The portion inside those edges is filled with a white color forming a mask which will be then multiplied with the square area resulting from step 3(d). So, an area is segmented containing both iris and pupil.

5. Pupil isolation

Finally, a mask is formed to eliminate the pupil area. Pixel levels are chosen according to different pixel-intensity categories that are assigned in step 1. Finally, this mask is multiplied with the resulting image from step 4 to obtain an image with a cropped iris only.

Algorithm. Intensity and Position-Based Iris Segmentation

Input: RGB Eye Image I

Output: Final Segmented Iris Image I_seg

// Stage 1: Illumination Estimation & Normalization

Count pixels in I to get N_{high} (>200), N_{med} (100-200), and N_{low} (<100).

Determine scaling factor α based on dominant N category.

$I_{adj} \leftarrow I \times \alpha$

Extract and enhance Red channel: $R_{enh} \leftarrow Red(I_{adj}) \times \beta$

// Stage 2: Sclera Detection & ROI Extraction

$Mask_sclera \leftarrow (Red > 100) \text{ AND } (Green > 100) \text{ AND } (Blue > 100)$

$Mask_sclera \leftarrow MorphClean(Mask_sclera)$ // Remove noise, close gaps

Bounding Box $BB \leftarrow HeuristicScan(Mask_sclera, R_{enh})$ // Bottom-up scanning for eye corners

$I_{roi} \leftarrow Crop(I, BB)$

// Stage 3: Noise Reduction

$I_{roi} \leftarrow MedianFilter(I_{roi}, 5 \times 5)$

// Stage 4: Iris Boundary Tracing (Directional Probing)

Initialize Boundary Matrix $H \leftarrow Zeros$

Calculate dynamic thresholds T, T2 based on ROI dimensions

For each pixel (x,y) in I_{roi} :

If pixel is non-white AND has a white neighbor:

Cast a directional probe of length T opposite to the white neighbor

If probe remains strictly non-white:

Cast a secondary diagonal probe of length T2

If secondary probe remains non-white:

$H(x,y) \leftarrow 1$

// Stage 5: Boundary Gap Filling

$H_{horiz} \leftarrow$ Fill horizontal gaps in H via left/right sweeps

$H_{vert} \leftarrow$ Fill vertical gaps in H via top/bottom sweeps

$H_{merged} \leftarrow H \text{ OR } H_{horiz} \text{ OR } H_{vert}$

// Stage 6: Iris Mask Generation

$Mask_iris \leftarrow MorphClean(H_{merged})$

$Mask_iris \leftarrow FillHoles(Mask_iris)$

// Stage 7: Dynamic Pupil Isolation

Determine dark threshold γ based on initial $N_{high}/N_{low}/N_{med}$ counts

$Mask_pupil \leftarrow (Red < \gamma) \text{ AND } (Green < \gamma) \text{ AND } (Blue < \gamma)$ within $Mask_iris$

$Mask_pupil \leftarrow MorphClean(Mask_pupil)$

// Stage 8: Final Extraction

$I_{seg} \leftarrow I_{roi} \odot Mask_iris \odot NOT(Mask_pupil)$

Return I_{seg}

3.5.3 Computational optimizations

•**8-Bit Logic Compatibility:** The core of the proposed algorithm relies on basic thresholding and fixed-precision multiplications (e.g., 0.6, 2.5) on 8-bit pixel values (0-255). This underlying logic maps directly to native 8-bit unsigned integer arithmetic in C/C++, avoiding the overhead of floating-point computations on microcontrollers.

•**Lightweight filters:** The algorithm using simplified Median filter instead of computationally heavy convolution for Gaussian blur.

•**No CNNs:** Avoid deep learning overhead for faster runtime and ensures minimal memory footprint and deterministic.

•The proposed algorithm underlying logic relies on basic convolution and morphological operations, making it readily

portable to C/C++ for deployment using OpenCV Lite or TensorFlow Lite on edge devices in future work.

•Hardware-agnostic design: The algorithm requires no specialized hardware accelerators (such as Neural Processing Units (NPUs) or GPUs). Its deterministic execution time and low RAM requirements make it ideal for integration into IoT firmware.

3.5.4 Experimental setup

- (1) Datasets used
- CASIA-IrisV3: Indoor and outdoor iris images.
 - UBIRIS.v2: Noisy images with motion blur and poor lighting.

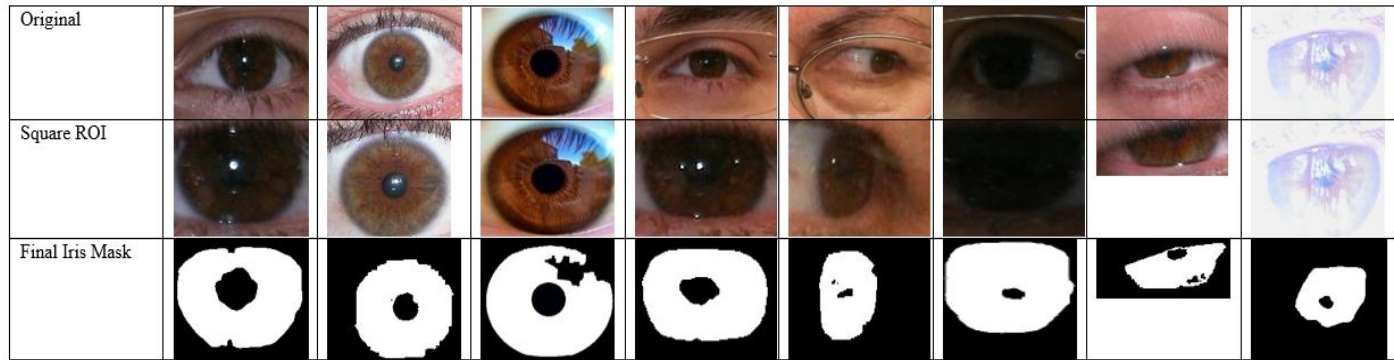


Figure 2. Examples of the results achieved by our segmentation method

4. RESULTS AND DISCUSSION

4.1 Accuracy evaluation

Standard measures confirmed the accuracy of the proposed detection algorithm through testing across CASIA-IrisV3 and UBIRIS.v2 and IITD benchmark sets (refer to Table 2 for results). The evaluation of the solution utilizes Precision, Recall, F1-Score, and Intersection over Union (IoU) as the main assessment metrics.

Table 2. Accuracy evaluation across datasets

Dataset	Precision	Recall	F1-Score	IoU (Iris ROI)
CASIA-IrisV3	0.981	0.976	0.978	0.925
UBIRIS.v2	0.942	0.935	0.938	0.876
IITD	0.987	0.982	0.984	0.932

The proposed method delivered uniform results when processing different iris colors while working with various angles and camera distances. The proposed method demonstrated high-precision performance under difficult or less-than-ideal lighting conditions (seen in Section 4.5).

4.2 Complexity and efficiency

The system operates efficiently on hardware systems using minimal operating power because of its compact design features suitable for embedded devices with limited processing capabilities. The analysis data for complexity and efficiency appears in Table 3.

The system acquired values when it processed streaming of grayscale video at 640 × 480 pixels resolution. There was no

•The IITD Iris Database contains NIR-enabled high-resolution iris scans.

These datasets were selected to represent diverse imaging conditions. The experiments considered variations under low light, occlusion, eyeglass reflection, motion blur, and brightness to evaluate the robustness of the proposed method, as shown in Figure 2.

- (2) Baselines for comparison
- OSIRIS: Open-source iris recognition framework [34, 38].
 - Lightweight CNN: Lightweight iris detection method with CNN backbone.
 - Fast Hough Transform (FHT) for segmentation [39].

implementation of deep learning models, which resulted in lower system complexity.

Figure 3 shows how the proposed system achieves much higher real-time frame rates than the baseline methods during direct comparison. The proposed solution demonstrates superior performance to lightweight solutions by keeping high frame rates on limited hardware systems.

Table 3. Computational complexity and efficiency metrics

Metric	Value
Average Runtime	22.4 ms (Pure processing per frame)
Memory Usage	~31 MB
Frames Per Second	~42 Frames Per Second (FPS) (Includes I/O and post-processing overhead)

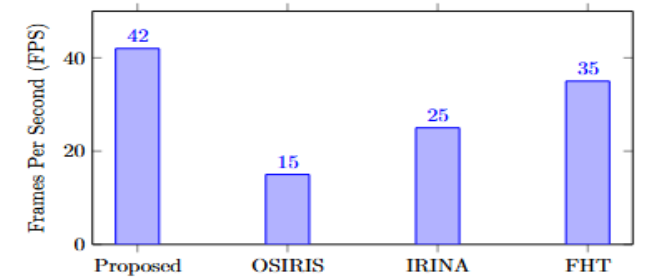


Figure 3. Frames Per Second (FPS) comparison of proposed and baseline methods

4.3 Comparison with existing methods

The proposed system evaluated performance against three fast iris/pupil detection methods and its results are described in Table 4.

- OSIRIS: Open-source iris recognition framework [34, 38].

•Lightweight CNN: Lightweight iris detection method with CNN backbone [implemented in this work].

•FHT for segmentation [39].

Our method was tested against recognized lightweight iris detection solutions while assessing their operational speed and computational requirements and performance quality. Table 4 presents a comprehensive comparison. All baseline methods were evaluated using the CASIA-IrisV3 dataset under identical experimental settings described in Section 3.5.4 and the parameter values in the proposed method are described in 3.5.2, while the baseline methods were executed using the default parameter settings recommended in their original publications. No hardware-specific acceleration (e.g., GPU acceleration) was applied during the comparative evaluation.

Table 4. Comparison with baseline methods on the CASIA-IrisV3 dataset

Method	Precision	F1-Score	FPS	Memory (MB)	FLOPs (M)
Proposed	0.981	0.978	42	31	3.2
OSIRIS	0.964	0.960	15	85	10.4
Lightweight CNN	0.973	0.970	25	50	28.7
FHT Baseline	0.948	0.940	35	40	6.1

Note: Frames Per Second (FPS), Convolutional Neural Networks (CNN), Floating Point Operations (FLOPs), Fast Hough Transform (FHT).

The main benefit our method offers consists of faster FPS and reduced computational requirements, which enable real-time processing needed by IoT applications.

4.4 Ablation study

The detection pipeline components were analyzed separately through an ablation study. This study reveals the impact of inactivating critical modules within the system based on data presented in Table 5.

Table 5. Ablation study on algorithm components using the CASIA-Irisv3 dataset

Configuration	F1-Score	FPS
Full Pipeline (Proposed)	0.978	42
Without Gaussian Preprocessing	0.963	44
Without Histogram Equalization	0.948	43
Without Morphological Filters	0.955	44
Using Floating-Point Arithmetic	0.978	27

Note: Frames Per Second (FPS).

The implementation of histogram equalization together with morphology filters creates substantial improvements in accuracy levels. Integer arithmetic significantly boosts speed.

The system component contributions can be observed in Figure 4, which illustrates the ablation results. The F1-Score received the most significant improvement from Histogram equalization and morphology filters, yet the conversion to integer arithmetic maintained performance while delivering major FPS increases.

4.5 Robustness

The robustness of the proposed iris segmentation method was evaluated on the CASIA-IrisV3 dataset under a variety of

challenging imaging conditions.

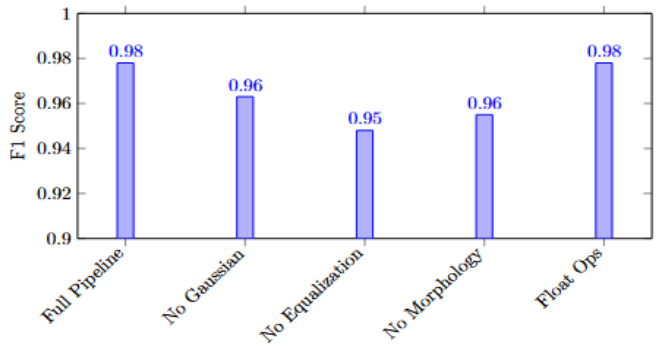


Figure 4. Impact of each component on accuracy and efficiency (ablation study)

Tests were conducted on the algorithm under these specific conditions:

•The detection system operates optimally with three different lighting conditions, including direct sunlight illumination as well as indoor lighting that is either dim or illuminates backlit images.

•Occlusion: Eyelashes, eyelids, glasses.

•Noise: Gaussian, motion blur, JPEG compression.

We evaluated our proposed detection method by testing its robustness under various challenging image conditions consisting of changing light conditions together with partial obstruction problems and digital noise. The analysis of F1-Score along with IoU conducted under different conditions appears in Table 6.

Table 6. Robustness evaluation under challenging conditions on the CASIA-IrisV3 dataset

Condition	F1-Score	IoU
Normal Indoor Light	0.978	0.925
Low Light	0.945	0.887
Occlusion (partial)	0.952	0.894
Motion Blur (mild)	0.938	0.872

A small system performance reduction exists; however, the operation stays reliable. Most failures occurred with highly degraded images under challenging imaging conditions.

4.6 Use case demonstration/application layer integration

4.6.1 Illustrative Internet of Things deployment

A demonstration example deals with using a Smart Door Lock System integrated with IoT technologies.

•Step 1: Instant eye image recording occurs through the camera after a user appears in its field of view.

•Step 2: The system localizes the pupil and iris using the developed algorithm in step two.

•Step 3: A comparison of the extracted iris code against the stored database reference takes place in Step 3.

•Step 4: The door automatically unlocks using the system when the matching result meets the predefined threshold.

Other potential applications:

•Contactless attendance systems for schools or offices.

•The technology integrates healthcare ID functionality into Augmented Reality (AR) eyeglasses for wearable purposes.

•Access control in secure environments.

4.6.2 Illustrative end-to-end workflow overview

Data flow:

1. Capture: The camera device takes a picture of the eye for iris capture.
2. Processing: The embedded system handles local image processing tasks at the processing stage.
3. Detection: ROI extraction takes place by performing pupil and iris localization on the detected data.
4. Matching: The iris code is derived from a database matching either local or remote storage.
5. Decision: The system gives authorization or denies access through the matching score evaluation.

In a future deployment, the cloud system could be connected through Message Queuing Telemetry Transport (MQTT) or Hypertext Transfer Protocol Secure (HTTPS) protocols for both monitoring and logging purposes.

The integration process for the suggested end-to-end workflow is illustrated in Figure 5. It is assumed that all operations from image capture to feature processing and matching would be executed locally on the edge device, although the system could support logging purposes or scalability extension.

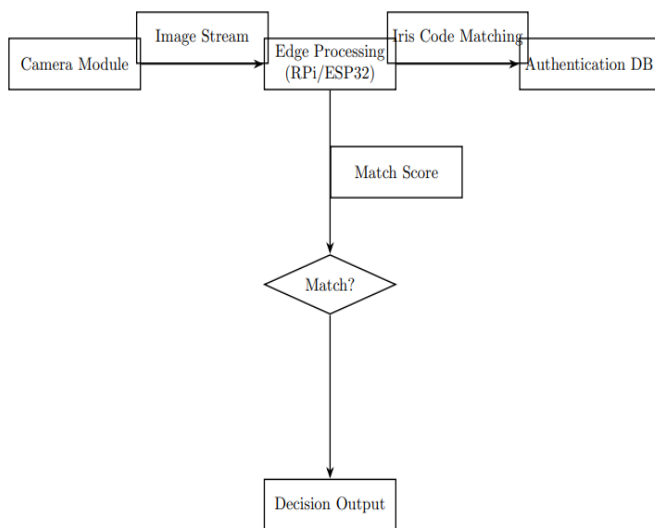


Figure 5. Illustrative end-to-end deployment pipeline of the biometric Internet of Things (IoT) system

5. CONCLUSION

The presented algorithm delivers an efficient pupil and iris detection technique specifically designed to be a suitable candidate for work within IoT-driven biometric systems. Tests on benchmark datasets show that the proposed algorithm achieves more than 0.97 F1-Score precision, while MATLAB-based experiments indicate a processing speed of approximately exceeding 42 FPS. The proposed approach delivers equivalent accuracy to classical lightweight solutions but decreases their computational requirements and memory usage remarkably, which makes it a suitable candidate for edge computing operations with low resource availability. Performance limitations emerge in this method even though it shows satisfactory robustness and efficiency characteristics. System performance diminishes during conditions where complete shadows cover detected features and rapid camera movement occurs together with harsh lighting variations beyond accepted quality levels. Several promising prospects

exist to enhance this work's capabilities in the future. The integration of the proposed detection framework with edge-cloud systems could support the future development of biometric authentication systems by providing an efficient front-end for iris recognition pipelines. The use of mobile transformers alongside compact CNNs or other lightweight deep learning components may further improve accuracy and adaptability while maintaining relatively low computational requirements. Hardware-specific optimizations that include special digital signal processors and field programmable gate arrays (FPGAs) and GPUs could be investigated to boost performance levels of this system. The proposed detection framework could be merged with face and voice biometric systems to enhance security protocols in healthcare and smart home and surveillance systems.

REFERENCES

- [1] Awad, A.I., Babu, A., Barka, E., Shuaib, K. (2024). AI-powered biometrics for Internet of Things security: A review and future vision. *Journal of Information Security and Applications*, 82: 103748. <https://doi.org/10.1016/j.jisa.2024.103748>
- [2] Ghosh, U.B., Sharma, R., Kesharwani, A. (2022). Symptoms-based biometric pattern detection and recognition. In *Augmented Intelligence in Healthcare: A Pragmatic and Integrated Analysis*, pp. 371-399. https://doi.org/10.1007/978-981-19-1076-0_19
- [3] Asi, B.A., Mahmood, F.E., Najim, N.I. (2025). Breaking barriers in vivo THz communication analysis for nano networks in human tissues. *Periodica Polytechnica Electrical Engineering and Computer Science*, 69(1): 26-32. <https://doi.org/10.3311/PPee.37844>
- [4] Abdulmalek, S., Nasir, A., Jabbar, W.A., et al. (2022). IoT-based healthcare-monitoring system towards improving quality of life: A review. *Healthcare*, 10(10): 1993. <https://doi.org/10.3390/healthcare10101993>
- [5] Ahmed, I., Rakhaine, N., Mahbub, A., Halder, B., Islam, M.R., Shafi-ul-Mulk. (2024). IoT-driven smart workplace ecosystem with RFID security and environmental monitoring featuring App integration. In *2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON)*, Rajshahi, Bangladesh, pp. 664-669. <https://doi.org/10.1109/PEEIACON63629.2024.10800453>
- [6] Rao, K.S., Chong, P.L., Kong, F.Y., et al. (2024). Transformative applications of IoT in diverse industries: A mini review. *Malaysian Journal of Science and Advanced Technology*, 4(2): 130-140. <https://doi.org/10.56532/mjsat.v4i2.292>
- [7] Sridhar, V., Rani, S., Pareek, P.K., Bhambri, P., Elngar, A.A. (2024). Evolution of IoT in various application domains. In *Blockchain for IoT Systems*, pp. 13-22. Chapman and Hall/CRC. <https://doi.org/10.1201/9781003460367>
- [8] Sworna, N.S., Islam, A.K.M.M., Shatabda, S., Islam, S. (2021). Towards development of IoT-ML driven healthcare systems: A survey. *Journal of Network and Computer Applications*, 196: 103244. <https://doi.org/10.1016/j.jnca.2021.103244>
- [9] Anjomshoa, F., Aloqaily, M., Kantarci, B., Erol-Kantarci, M., Schuckers, S. (2017). Social

- behaviometrics for personalized devices in the Internet of Things era. *IEEE Access*, 5: 12199-12213. <https://doi.org/10.1109/ACCESS.2017.2719706>
- [10] Dou, F., Ye, J., Yuan, G., et al. (2023). Towards artificial general intelligence (AGI) in the Internet of Things (IoT): Opportunities and Challenges. *arXiv preprint arXiv:2309.07438*. <https://doi.org/10.48550/arXiv.2309.07438>
- [11] Mammadov, S., Kucukkulahli, E. (2025). A user-centric smart library system: IoT-driven environmental monitoring and ML-based optimization with future fog-cloud architecture. *Applied Sciences*, 15(7): 3792. <https://doi.org/10.3390/app15073792>
- [12] Jagatheesaperumal, S.K., Pham, Q.V., Ruby, R., Yang, Z., Xu, C., Zhang, Z. (2022). Explainable AI over the Internet of Things (IoT): Overview, state-of-the-art and future directions. *IEEE Open Journal of the Communications Society*, 3: 2106-2136. <https://doi.org/10.1109/OJCOMS.2022.3215676>
- [13] Mishra, D.K., Dey, N., Deora, B.S., Joshi, A. (2020). ICT for competitive strategies. In *Proceedings of 4th International Conference on Information and Communication Technology for Competitive Strategies (ICTCS 2019)*, Udaipur, India, pp. 976. <https://doi.org/10.1201/9781003052098>
- [14] Raad, H. (2021). *Fundamentals of IoT and Wearable Technology Design*. John Wiley & Sons, pp. 1-182.
- [15] Yadav, N., Kang, S.S., Verma, R., Garg, V., Singla, I. (2024). Enhancing urban mobility: A smart parking system using IoT and machine learning. In *2024 Second International Conference on Advanced Computing & Communication Technologies (ICACCTech)*, Sonipat, India, pp. 661-666. <https://doi.org/10.1109/ICACCTech65084.2024.00111>
- [16] Gopi, B., Hasan, M.D.A., Kumar, G., Sivakumar, S., Raju, M.J., Malathi, N. (2024). Patient-centric skill evaluation in IoT-based surgical training with naive bayes algorithm for personalized feedback. In *2024 2nd International Conference on Computer, Communication and Control (IC4)*, Indore, India, pp. 1-5. <https://doi.org/10.1109/IC457434.2024.10486657>
- [17] KardanMoghaddam, H., Akbarimajd, A., Jamali, S., Nooshyar, M., Najafabadi, S. (2025). The role of the Internet of Things (IoT) in the education department. *Facta Universitatis, Series: Electronics and Energetics*, 38(1): 1-18. <https://doi.org/10.2298/FUEE2501001K>
- [18] Gupta, S. (2020). Next-generation user authentication schemes for IOT applications. Ph.D. dissertation, Department of Information Engineering and Computer Science (DISI), ICT International Doctoral School, Università di Trento, Trento, Italy. <https://tesidottorato.depositolegale.it/handle/20.500.14242/179140>
- [19] Nwankwo, W., Adetunji, C.O., Olayinka, A.S. (2022). IoT-driven bayesian learning: A case study of reducing road accidents of commercial vehicles on highways. *Artificial Intelligence-based Internet of Things Systems*, 391-418. https://doi.org/10.1007/978-3-030-87059-1_15
- [20] Kumari, M., Gaikwad, M., Chavan, S.A. (2024). Review on open-source IoT and edge-compatible devices for health monitoring applications. *Instrumentation Measure Métrologie*, 23(3): 233-241. <https://doi.org/10.18280/i2m.230307>
- [21] Nazir, S., Ali, Y., Ullah, N., García-Magariño, I. (2019). Internet of things for healthcare using effects of mobile computing: A systematic literature review. *Wireless Communications and Mobile Computing*, 2019: 5931315. <https://doi.org/10.1155/2019/5931315>
- [22] Tirandazi, P., Bamakan, S.M.H., Togholjerdi, A. (2023). A review of studies on internet of everything as an enabler of neuromarketing methods and techniques. *The Journal of Supercomputing*, 79(7): 7835-7876. <https://doi.org/10.1007/s11227-022-04988-1>
- [23] Kumar, M.V., Ramesh, G.P., Pareek, P.K., Deepak, H.A., Babu, J.A. (2023). Robotic attendance scheme in the classroom using artificial intelligence and Internet of Things. In *2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC)*, Dharwad, India, pp. 1-6. <https://doi.org/10.1109/ICAISC58445.2023.10200650>
- [24] Spaho, E., Çiço, B., Shabani, I. (2025). IoT integration approaches into personalized online learning: Systematic review. *Computers*, 14(2): 63. <https://doi.org/10.3390/computers14020063>
- [25] Tariq, R., Casillas-Muñoz, F., Hassan, S., Ramírez-Montoya, M. (2024). Synergy of Internet of Things and education: Cyber-physical systems contributing towards remote laboratories, improved learning, and school management. *Journal of Social Studies Education Research*, 15(2): 305-352.
- [26] Zeeshan, K., Neittaanmäki, P. (2021). Internet of Things enabling smart school: An overview. In *2021 IEEE 18th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*, Karachi, Pakistan, pp. 152-156. <https://doi.org/10.1109/honet53078.2021.9615391>
- [27] Mane, D., Magar, A., Rhode, O., Koli, S., Bhat, K., Korade, P. (2024). The impact of Internet of Things (IoT) on university library services. *Library of Progress-Library Science, Information Technology & Computer*, 44(1): 1-16. <https://doi.org/10.52710/lpi.44.1.15>
- [28] Kaviyazhiny, C., Bala, P.S., Priyadhrashini, R., Ajeeth, S. (2024). Analysis of factors impacting the efficiency of EEG-based biometric authentication system. In *2024 International Conference on System, Computation, Automation and Networking (ICSCAN)*, Puducherry, India, pp. 1-7. <https://doi.org/10.1109/ICSCAN62807.2024.10894108>
- [29] Sagar, M.S.I. (2023). *Intelligent devices for IoT applications*. Washington State University. <https://doi.org/10.7273/000005013>
- [30] Khang, A. (2024). *Driving Smart Medical Diagnosis Through Ai-Powered Technologies and Applications*. IGI Global, pp. 1-312.
- [31] Jiang, J., Zhang, Q., Wang, C. (2025). SAM-iris: A SAM-based iris segmentation algorithm. *Electronics*, 14(2): 246. <https://doi.org/10.3390/electronics14020246>
- [32] Milman, O., Yellin, D., Aperstein, Y. (2025). Adapting SAM for visible-light pupil segmentation baseline. *Electronics*, 14(9): 1850. <https://doi.org/10.3390/electronics14091850>
- [33] Tato, F.R., Kareem, O.S. (2025). State-of-the-art machine learning and deep learning techniques in iris recognition: A review. *Engineering and Technology Journal*, 10(5). <https://doi.org/10.47191/etj/v10i05.35>
- [34] Othman, N., Dorizzi, B., Garcia-Salicetti, S. (2016). OSIRIS: An open source iris recognition software. *Pattern Recognition Letters*, 82: 124-131.

- <https://doi.org/10.1016/j.patrec.2015.09.002>
- [35] Wang, C.Y., Wang, Y.L., Xu, B.Q., He, Y., Dong, Z.W., Sun, Z.N. (2020). A lightweight multi-label segmentation network for mobile iris biometrics. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, pp. 1006-1010. <https://doi.org/10.1109/ICASSP40776.2020.9054353>
- [36] Sun, Y., Lu, Y.N., Liu, Y.N., Liu, S., Liu, Z. (2022). LiSeNet: Multitask lightweight segmentation network for accurate and complete iris segmentation. *Journal of Electronic Imaging*, 31(5): 053035. <https://doi.org/10.1117/1.JEI.31.5.053035>
- [37] Ruiz-Beltrán, C., Pons, Ó., González-García, M., Bandera, A. (2025). Real-time detection and segmentation of the iris at a distance scenarios embedded in ultrascale MPSoC. *Electronics*, 14(18): 3698. <https://doi.org/10.3390/electronics14183698>
- [38] OSIRIS: Open Source Iris Recognition Software. (2018). GitHub repository. <https://github.com/tohki/iris-osiris>.
- [39] Tian, Q.C., Pan, Q., Cheng, Y.M., Gao, Q.X. (2004). Fast algorithm and application of Hough transform in iris segmentation. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*, Shanghai, China, pp. 3977-3980. <https://doi.org/10.1109/ICMLC.2004.1384533>

NOMENCLATURE

$I(x,y)$	Input RGB eye image pixel intensity
I_{adj}	Brightness-normalized image
I_{roi}	Region of Interest image
I_{seg}	
H	Iris boundary matrix
BB	Bounding box enclosing the iris
R, G, B	Red, Green and Blue color channels
N_{high}	Number of high-intensity pixels
N_{med}	Number of medium-intensity pixels
N_{low}	Number of low-intensity pixels
T	Primary probing length (pixels)
$T2$	Secondary probing length (pixels)

Greek symbols

α	Brightness scaling factor (dimensionless)
β	Red-channel enhancement factor (dimensionless)
γ	Adaptive pupil threshold (pixel intensity)

Subscripts

adj	Adjusted image
$high$	High-intensity pixels
low	Low-intensity pixels
med	Medium-intensity pixels
roi	Region of interest
seg	Segmented iris image