



The Human Eye Pupil Detection System Using BAT Optimized Deep Learning Architecture

S. Navaneethan^{1,*}, P. Siva Satya Sreedhar², S. Padmakala³ and C. Senthilkumar⁴

¹Department of Electronics and Communication Engineering, Saveetha Engineering College, Chennai, 602105, India

²Department of Information Technology, Seshadri Rao Gudlavalluru Engineering College, Krishna, 521356, India

³Department of Computer Science Engineering, Saveetha School of Engineering, SIMATS, Chennai, 602105, India

⁴Department of Electronics and Communication Engineering, Saveetha School of Engineering, SIMATS, Chennai, 602105, India

*Corresponding Author: S. Navaneethan. Email: navaneethans@saveetha.ac.in

Received: 20 July 2022; Accepted: 29 September 2022

Abstract: The pupil recognition method is helpful in many real-time systems, including ophthalmology testing devices, wheelchair assistance, and so on. The pupil detection system is a very difficult process in a wide range of datasets due to problems caused by varying pupil size, occlusion of eyelids, and eyelashes. Deep Convolutional Neural Networks (DCNN) are being used in pupil recognition systems and have shown promising results in terms of accuracy. To improve accuracy and cope with larger datasets, this research work proposes BOC (BAT Optimized CNN)-IrisNet, which consists of optimizing input weights and hidden layers of DCNN using the evolutionary BAT algorithm to efficiently find the human eye pupil region. The proposed method is based on very deep architecture and many tricks from recently developed popular CNNs. Experiment results show that the BOC-IrisNet proposal can efficiently model iris microstructures and provides a stable discriminating iris representation that is lightweight, easy to implement, and of cutting-edge accuracy. Finally, the region-based black box method for determining pupil center coordinates was introduced. The proposed architecture was tested using various IRIS databases, including the CASIA (Chinese academy of the scientific research institute of automation) Iris V4 dataset, which has 99.5% sensitivity and 99.75% accuracy, and the IIT (Indian Institute of Technology) Delhi dataset, which has 99.35% specificity and MMU (Multimedia University) 99.45% accuracy, which is higher than the existing architectures.

Keywords: BAT algorithm; IRIS datasets; DCNN; pupil detection; black box method

1 Introduction

Pupil detection algorithms are critical in medical ophthalmology diagnostic equipment such as non-contact tonometers, auto ref-keratometers, optical coherence tomography, and so on in today's digital world. Most eye diseases, such as glaucoma and cataracts, are detected early on using this equipment. Any algorithm used in medical equipment must be precise and fast. Because the eye is not always stable,



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

funds are invested in diagnostic equipment to align the centre region of the eye. The most crucial step in measuring diagnostic parameters is pupil detection. Due to their complexity, traditional approaches rarely provide real-time pupil centre detection in a general-purpose computing system. Using Convolutional Neural Networks-based models, it will be possible to determine the pupil's centre [1] accurately. Artificial neural networks [2] are also improved the accuracy of medical diagnostic applications. Intelligent learning, enhanced network designs, and intelligent training methods are used, and CNN is thought to perform best in all vision-related applications. Several convolutional neural network (CNN) models are compared, to parallelize them using a distributed framework to detect the human eye pupil region [3]. Traditional global optimization methods have several advantages over the basic bat algorithm. This bat algorithm and convolutional neural network (BA-CNN) contribution is used to find the human eye pupil region efficiently. Based on the performance of deep networks in other applications, we propose BOC-IrisNet for pupil detection. This robust deep CNN architecture can manage large-size iris datasets with complex distributions. The following is the work's contribution:

1. The BAT design optimized convolution layers to improve pupil detection accuracy.
2. A new black box method is proposed to determine the pupil center coordinates.
3. Implementing the above architecture for effective evaluation parameter usage and accuracy.

2 Related Works

The pupil detection technique is used in various crucial medical research applications. Bergera et al. [4] propose an accurate pupil detection technique for tracking the eye using a convolutional neural network. This network's architecture is similar to that of DeepLabCut. It is made up of the ResNet-50 pre-trained with the ImageNet dataset. Lin et al. [5] created an eye detection scheme for video streams. Because of the video data set, this method is costly in computation. Daugman [6] and Dubey et al. [7] used an integrodifferential operator and hough transform to locate iris boundaries, but the accuracy and resource utilization are too high. Yu et al. [8] proposes the Starburst algorithm and random sample consensus (RANSAC) algorithm, which accurately detects the pupil region with less accuracy. Tann et al. [9] proposed segmentation, contour matching, normalization, and Daugman coding using FCN. A tri-stage software and hardware collaborative design methodology is introduced to achieve an efficient FCN model, which includes a review of FCN architecture, precision quantification, and hardware acceleration. GPU findings show higher energy performance for larger models but much lower efficiency for smaller models. Swadi et al. [10] demonstrate significant performance gains and speed improvements in the Spartan 7 iris identification and validation process. The results of this work are validated against the CASIA V1 and CASIA Interval databases, which are used to generate various rectangular aperture patterns. The only issue with the iris pattern is that it is larger than a quarter of the iris, which means that the time required to identify is less than a quarter of the iris. Chun et al. [11] proposed a new iris authentication framework for root systems based on feature extraction and classification using Iris Match-CNN. The proposed structure provides a faster, more reliable, and more secure iris examination model. The Iris Match-CNN model does not rely on live training data at the registration phase. Kumar et al. [12] described a unique iris positioning tool that can be used as a coprocessor in the development of a low-cost embedded real-time biometric system for the iris. The prototype system has been described as being used to locate the iris in visible wavelength (VW) images, but it can also be used in near-infrared (NIR) images. Navaneethan et al. [13] proposed using average black pixel density to detect the pupil region on an FPGA. Simple threshold and morphology techniques were used in this work, which is unsuitable for larger datasets. Nandhagopal et al. [14] proposed a center of gravity (COG) method for detecting the pupil region in various iris datasets. The disadvantage of the COG method is that it cannot determine the exact center region of the pupil and is not predictable. Navaneethan et al. [15] discussed the FPGA

realization of human eye pupil detection using the e2v camera interface. Here, we examined existing algorithms developed by the researchers, such as CHT, starburst, COG, and CNN-based gaze detection methods, which have less accuracy and more computational complexity in terms of architecture. To overcome this complexity, the BAT optimized deep learning architecture (BOC-IrisNets) was proposed for the human eye pupil detection system.

3 BAT Optimized CNN Architecture

Convolutional Neural Networks (CNN) are not only deep learning types but also fall under the umbrella of Artificial Neural Networks (ANN), which have applications in image recognition and video analytics. The CNN platform is made up of five layers. The entry layer is made up of the normalized sequence matrix, and feature maps are used to link the inputs to the layers above it. The features obtained by the convolutional layers are used as inputs to the pooling layers. Both neurons in a function map share the same kernel and weight parameters [16].

3.1 Convolutional Neural Networks

A CNN has several convolutional layers as well as a pooling layer that determines the average or maximum of a neuron window that follows each convolutional layer. This layer reduces the complexity of the output function's maps, achieves invariance in localization, and reduces over-fitting. The final layers of a CNN are fully connected (FC) or dense layers that interlink all neurons from previous layers. Complex characteristics originating from convolution layers are globally related. Each fully connected layer often contains several kernels. However, these kernels are only connected to the input map once in thick layers. As a result, since each kernel is only used once, dense layers are not machine intensive. A large number of weights to be transferred from external memory is a limit imposed by thick layers. CNN's with these three-layer types are said to be regular.

3.2 BAT Algorithm

The following are the concepts of BAT algorithms: at first, bats use echolocation. They then distinguish between an obstacle and a victim. The standard estimate of the bat was based on microbat echolocation or bio-sonar characteristics. Most bats use fixed frequency signals, but some use tuning frequency signals. Bats' frequency range is typically between 150 and 25 kHz. Therefore, In light of the echo cancellation computations, Bangyal et al. [17] established the following three glorified instructions:

1. Both bats use echolocation to assess distance and magically "remember" the difference between food/prey and ground barriers.
2. Bats migrate randomly at position x_i with a stable frequency f_i and loudness A_0 to hunt for prey.
3. Even though the volume will change in various ways, we agree that the volume ranges from a big A_0 to the lowest constant value of A_{min} .

To estimate the distance, bat exploits their echolocation ability; also they can differentiate between insects and background obstacles. To hunt prey, bats always fly randomly with a certain velocity (v_i) at a location (x_i) that has a frequency (f_i) and loudness (A_0). Bat's pulse loudness may differ in different ways. However, Yang assumed that it ranged from a large positive number (A_0) to a minimum constant number (A_{min}). The loudness and pulse emission continuum is critical for a better approach. While the loudness usually decreases when a bat discovers its prey as the rate of pulse emission increases, the loudness may be selected between A_{min} and A_{max} as any accommodation calculation, accepting $A_{min}=0$ means that a bat has actually located the prey and temporarily stopped the movement of any safe.

3.3 BAT Optimized CNN Layer

Fig. 1 depicts the overall structure of BOC-Iris Network's proposed architecture. The proposed models are trained using various iris image databases. The proposed model employs the BAT-optimized CNN architecture to classify the pupil area. The CNN architecture combined a bat algorithm to quickly identify the pupil region. The proposed models are trained using the various categories of iris images. The bat algorithm randomly initializes the hyper-parameters based on the upper and lower boundary established during the parameter initialization step. The upper and lower bounds are initialized arbitrarily with integer values between 4 and 100 for the number of filters in the first and second convolution layers and the number of neurons in the hidden layer. In this situation, the number of filter layers is optimized to five to obtain the whole output. Bat parameters such as bat population size, the maximum number of iterations, optimum value, lower bound, dimension, noise level, pulse rate, and maximum and minimum frequency range are all initialized, as shown in Table 1. The Bat algorithm is used to tune four CNN parameters: the number of first and second convolution layer filters, the number of hidden neurons, and the filter height. The bat algorithm randomly initializes these hyper-parameters based on the upper and lower borders defined during the parameter initialization phase.

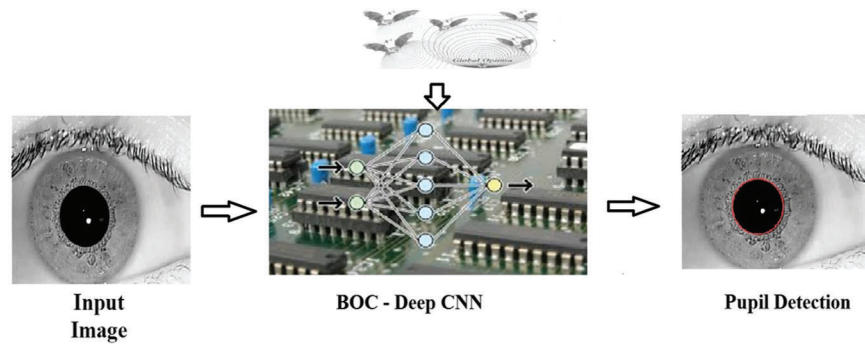


Figure 1: Overall framework for the proposed architecture used for pupil detection

Table 1: BAT parameter used for filter layer initialization of the proposed network

S. no	Parameters	Specifications
01	No BAT population	05
02	Initial velocity	23
03	Initial loudness	0.5
04	Initial pulse rate	0.5
05	Minimum frequency	0 KHZ
06	Initial loss function	0.001
07	Maximum frequency	0.2 KHZ

To generate a new solution in the simulation, the frequency, velocity, and bat position have been tuned from Eqs. (1) to (3) and the obtained solution is regarded better than the previous one depending on the fitness of the solutions adjusted by the frequency and the velocity, Which are in turn depends on how close it is to the optimal solution.

$$f_i = f_{min} + (f_{max} - f_{min}) * R \quad (1)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x^*)f_i \quad (2)$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

x_i^{t-1} -last bat iteration position

x^* -the best position among all bat positions

v_i^{t-1} -bat iteration velocities

R-a random number between 1 and 0

f_i -refers to the bats' frequency, we can change the bats' frequency parameters using maximum (f_{max}) and minimum (f_{min}) frequency values.

The entire BAT population's fitness is estimated using the mathematical expression in Eq. (4).

$$\text{FitnessFunction} = (1 - \alpha) + (1 - \beta) + (1 - \mu) \tag{4}$$

The fitness value of a bat is equal to the amount of time it takes to complete all the jobs. To calculate this value, we must first schedule all the tasks. At the end of the scheduling, the exact time of the start and end of each task is determined. Where α is the accuracy of detection, β is the sensitivity and μ is the specificity. The weights and filter layers are assigned in such a manner so that accuracy, sensitivity, and specificity are maintained as peak values. Here, iterations are based on the number of populations taken to find out the fitness function. Figs. 2 and 3 show the input image and pupil segmented image after the optimized convolutional layers.

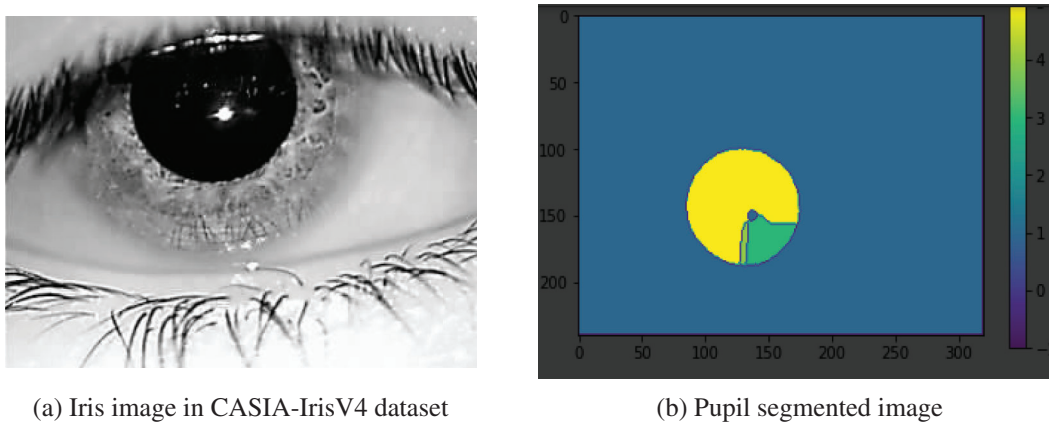


Figure 2: Iris image in CASIA-IrisV4

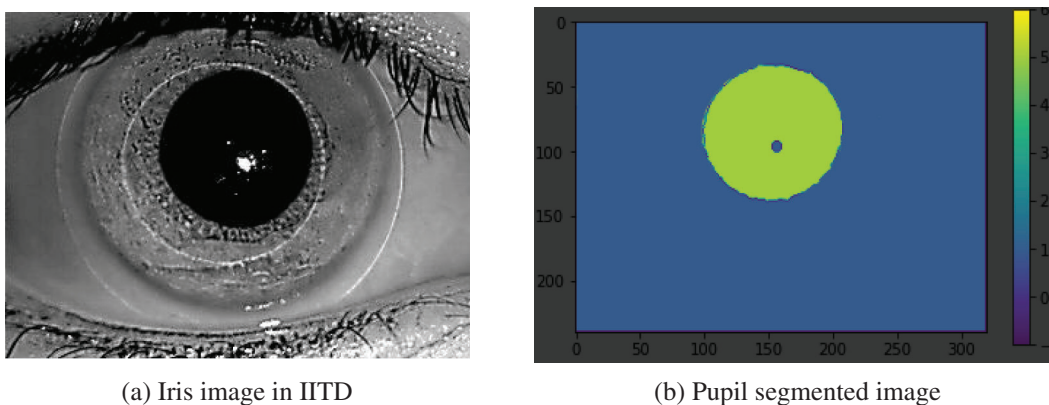


Figure 3: IIT-Delhi database image

3.4 The Black Box Method

Many studies [18–20] developed to determine the pupil region. The procedure for determining the pupil centre coordinate using the black box form is shown in Fig. 4. The BAT optimized CNN layer produced an image with pupil segmentation. The suggested black-box approach to avoiding the computation time of finding the pupil area coordinate in each pixel, here IIT Delhi image as an example and its resolution 320×240 can be split as 20×15 boxes with each box size 16×16 . These boxes surround the entire image. Count the number of boxes (N) in the image that is filled with black pixels. These boxes encircle the entire image. Count the number of boxes (N) filled by black pixels in the image. Calculate the average sum of x(xsum), and y(ysum) coordinates for matching boxes in the same way, then divide the xsum, ysum values by the total number of filled black boxes to get the pupil area core values and represents in Eq. (5). The coordinates of a 320×240 image is calculated by multiplying the center of the pupil area (xcoord and ycoord values) by 16.

$$\left. \begin{aligned} \text{Black Pupil density} &= \sum z \\ \text{xsum} &= \sum \text{xcoord} \\ \text{ysum} &= \sum \text{ycoord} \\ \text{xcenter} &= \sum \text{xcoord} / \sum z \\ \text{ycenter} &= \sum \text{ycoord} / \sum z \end{aligned} \right\} \quad (5)$$

where z is the black pixel count value

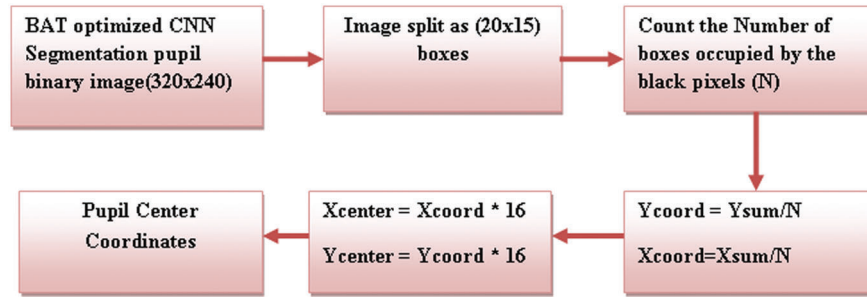


Figure 4: Data flow diagram of pupil center coordinates detection method

Eqs. (6) & (7) predict the pupil region radius.

$$A = \pi r^2 \quad (6)$$

Here A is the area of the circle.

A = Total number of black pixels.

To find r,

$$r = \text{sqrt}(A/\pi) \quad (7)$$

3.5 Evaluation Metrics

The CASIA Iris V4, IIT Delhi database, and MMU database are considered for input images. For each dataset, 80% has been taken as the training data and 20% as the testing data. From the confusion matrix of the BOC CNN network, we obtained the True positive (TP), True negative (TN), False positive (FP), and False negative (FN) values for the given dataset.

The valuation parameter includes sensitivity (SE), specificity (SP), accuracy (AC), and Recall (R). The distinct presentation factors are shown in Eqs. (8) to (11).

$$\text{Accuracy} = (\text{TP} + \text{TN}) / \text{TN} \times 100 \quad (8)$$

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{TN}) \times 100 \quad (9)$$

$$\text{Specificity} = \text{TN} / (\text{TP} + \text{TN}) \times 100 \quad (10)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \times 100 \quad (11)$$

4 Results and Discussion

In this unit, we discuss model outcomes of various parameters using both proposed and current schemes. The proposed method is investigated using MATLAB with a 3.0 GHz frequency Intel i5 CPU (Central Processing Unit), a 1 TB hard disc drive storage slot, and 8 GB RAM (Random Access Memory). Also, the proposed system's simulation outcome performance is compared to some recent conventional systems on the open datasets defined in Section 3.5.

Tables 2–4 demonstrate the comparative study of sensitivity, specificity, accuracy, and recall between the various frameworks using the massive CASIA-IrisV4, IIT Delhi, and MMU database, in which the proposed model outperformed the other learning models. IIT Delhi database use 1120 images, the CASIA database use 22,035 images, and the MMU dataset use 460 images of which 80% were taken for training and 20% for the test image. The original algorithm exhibits a marginal drop in efficiency as the number of datasets increases, while the optimized proposed algorithm maintains high accuracy of 99.75 percent even for larger datasets. And as the datasets were expanded, the Bat configured CNN layer maintained a consistent accuracy of 99.75 percent. The proposed architecture obtained 99.5% sensitivity and recall, 99.75% accuracy, and 99.23 specificities using the CASIA Iris V4 dataset which is higher than other datasets. In the same way, using the IIT Delhi dataset, the proposed architecture got 99.45% sensitivity, 99.60% recall, 99.68% accuracy, and 99.35% specificity. Finally, the MMU dataset used for comparison has 99.40% sensitivity, 99.32 recall, 99.45% accuracy, and 99.20 specificities in the proposed architecture. This result concludes that the proposed BAT-optimized CNN architecture is more accurate than the existing frameworks in different datasets. The sensitivity, specificity, accuracy, and recall parameters measure the performance of the proposed architecture which results from better improvement in this research work.

The trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) for various parameters is provided by the Receiver Operating Characteristic curve (ROC). The percentage of observations for which a positive outcome was accurately anticipated is known as the true positive rate. The percentage of observations that are falsely expected to be positive is known as the false positive rate. Different TPRs and FPRs will be obtained for various datasets. Fig. 5 depicts a typical ROC curve for IITD, CASIA iris V4, and MMU databases.

Fig. 6 explains the sensitivity, specificity, accuracy, and recall parameters of the IIT Delhi, CASIA irisV4, MMU database in the proposed BOC-IrisNets architecture.

Based on the confusion matrix obtained in all three datasets, the roc curve indicates true positive and false positive rates.

Table 2: Sensitivity, specificity, accuracy, and recall analysis between the different frameworks using CASIA-Iris V4 datasets

S. no	Framework mechanism	Sensitivity (%)	Specificity (%)	Accuracy (%)	Recall (%)
01	GST [21]	80.5	80.0	81.45	80.5
02	OSIRIS [22]	93.5	91.4	95.5	94.0
03	WAHET [23]	94.5	92.5	93.45	95.5
04	CAHT [24]	93.0	92.2	96.5	96.0
05	MASEK [25]	94.0	94.0	96.0	97.4
06	IRIS-SEG [26]	95.0	94.2	95.0	93.5
07	IDN [27]	96	94.8	95.0	96.7
08	DEEP FCN [28]	97.20	97.0	98.0	98.2
09	BOC-IRIS NETS	99.50	99.23	99.75	99.50

Table 3: Sensitivity, specificity, accuracy, and recall analysis between the different frameworks using IITD dataset

S. no	Framework mechanism	Sensitivity (%)	Specificity (%)	Accuracy (%)	Recall (%)
01	GST	83.5	81.4	80.35	82.0
02	OSIRIS	95.5	94.3	94.5	94.2
03	WAHET	96.4	94.5	91.45	93.5
04	CAHT	96.0	96.0	95.5	96.0
05	MASEK	96.0	96.3	97.1	96.4
06	IRIS-SEG	97.0	95.0	94.6	95.5
07	IDN	96.78	96.8	97.0	96.7
08	DEEP FCN	99.0	98.7	98.4	98.5
09	BOC-IRIS NETS	99.45	99.35	99.68	99.60

Table 4: Sensitivity, specificity, accuracy, and recall analysis between the different frameworks using MMU datasets

S. no	Framework mechanism	Sensitivity (%)	Specificity (%)	Accuracy (%)	Recall (%)
01	GST	82.4	81.0	82.55	81.75
02	OSIRIS	95.0	94.5	95.0	95.2
03	WAHET	96.0	94.2	96.45	95.5
04	CAHT	96.2	96.2	96.5	96.4
05	MASEK	97.2	96.4	97.75	97.4
06	IRIS-SEG	97.0	95.5	97.6	97.2
07	IDN	96.5	96.7	98.0	98.7
08	DEEP FCN	99.0	98.5	98.6	98.7
09	BOC-IRIS NETS	99.40	99.20	99.45	99.32

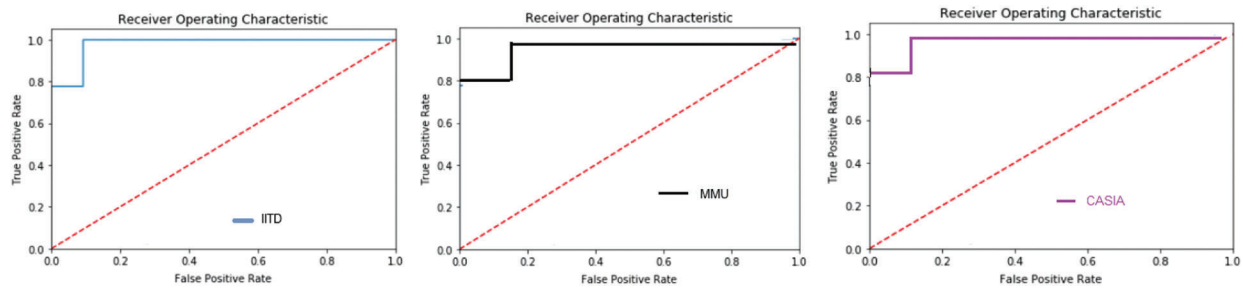


Figure 5: ROC curve for IITD, MMU, CASIA dataset predictions

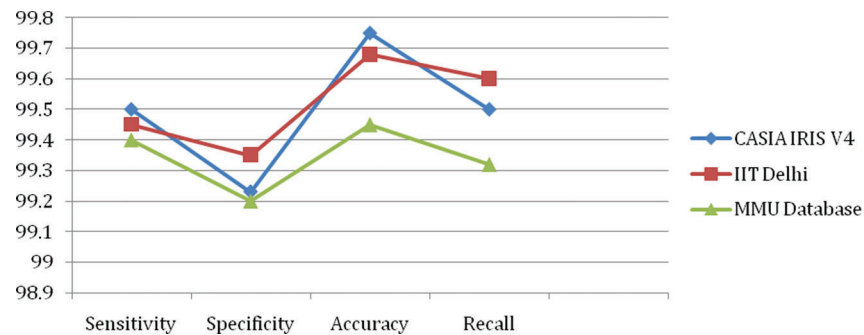


Figure 6: Proposed BOC-iris net performance evaluation for different iris datasets

5 Conclusion

Pupil identification is a critical step in many biomedical diagnostic systems, and it must be completed quickly. In this research, we used optimized BAT convolution layers to reduce the number of layers without sacrificing performance. The method CNN, along with the BAT optimization algorithm, guarantees 99.75 percent accuracy in the CASIA Iris V4 dataset, 99.68 percent accuracy in the IIT Delhi dataset, and 99.45 percent accuracy in the MMU dataset. The advantage of the system is used to predict the human eye pupil region with the highest accuracy of 99%. The proposed research demonstrates that CNN with the BAT optimization methodology was well suited for the prediction and identification of pupil detection using CASIA Iris V4 and IITD and MMU databases. A detailed experiment was carried out in which the true recognition rate of each model was evaluated, and it was discovered that the models exhibit enhanced accuracy, specificity, sensitivity, recall, and higher recognition rates. Due to the complexity of CNN architecture, utilization of the resources becomes high which is the only drawback of the proposed work. The proposed model will be more adapted in the future for resource-efficient low-power applications such as SOC implementation.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] K. I. Lee, J. H. Jeon and B. C. Song, "Deep learning-based pupil center detection for fast and accurate eye tracking system," in *European Conf. on Computer Vision, Berlin/Heidelberg, Germany*, Springer, vol. 12, pp. 36–52, 2020.
- [2] S. Mustafa, A. Jaffar, M. W. Iqbal, A. Abubakar, A. S. Alshahrani *et al.*, "Hybrid color texture features classification through ann for melanoma," *Intelligent Automation & Soft Computing*, vol. 35, no. 2, pp. 2205–2218, 2023.

- [3] D. Datta and S. B. Jamal Mohammed, "Image classification using CNN with multi-core and many-core architecture," in *Applications of Artificial Intelligence for Smart Technology*, Pennsylvania, United States: IGI Global, pp. 233–266, 2021.
- [4] A. L. Bergera, G. Garde, S. Porta, R. Cabeza and A. Villanueva, "Accurate pupil center detection in off the-shelf eye tracking systems using convolutional neural networks," *Sensors*, vol. 21, no. 6847, pp. 1–14, 2021.
- [5] K. Lin, J. Huang, J. Chen and C. Zhou, "Real-time eye detection in video streams," in *Fourth Int. Conf. on Natural Computation (ICNC'08)*, Jinan, China, pp. 193–197, 2008.
- [6] J. Daugman, "Statistical richness of visual phase information: Update on recognizing persons by iris patterns," *International Journal of Computer Vision*, vol. 45, no. 1, pp. 25–38, 2001.
- [7] R. B. Dubey and A. Madan, "Iris localization using daugman's intero-differential operator," *International Journal of Computer Applications*, vol. 93, no. 3, pp. 35–42, 2014.
- [8] P. Yu, W. Duan, Y. Sun, N. Cao, Z. Wang *et al.*, "A pupil-positioning method based on the starburst model," *Computers, Materials & Continua*, vol. 64, no. 2, pp. 1199–1217, 2020.
- [9] H. Tann, H. Zhao and R. Sherief, "A resource-efficient embedded iris recognition system using fully convolutional networks," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 16, no. 1, pp. 1–23, 2019.
- [10] M. J. Swadi, H. Ahmed and R. K. Ibrahim, "Enhancement in iris recognition system using FPGA," *Periodicals of Engineering and Natural Sciences*, vol. 8, no. 4, pp. 2169–2176, 2020.
- [11] Y. L. Chun, C. W. Sham and M. Longyu, "A novel iris verification framework using machine learning algorithm on embedded systems," in *IEEE 9th Global Conf. on Consumer Electronics (GCCE)*, Kobe, Japan, pp. 265–267, 2020.
- [12] V. Kumar, A. Asati and A. Gupta, "Hardware implementation of a novel edge-map generation technique for pupil detection in NIR images," *Engineering Science and Technology, an International Journal*, vol. 20, no. 4, pp. 694–704, 2017.
- [13] S. Navaneethan and N. Nandhagopal, "Re-pupil: Resource efficient pupil detection system using the technique of average black pixel density," *Sadhana*, vol. 46, no. 3, pp. 1–10, 2021.
- [14] N. Nandhagopal, S. Navaneethan, V. Nivedita, A. Parimala and D. Valluru, "Human eye pupil detection system for different iris database images," *Journal of Computational and Theoretical Nanoscience*, vol. 18, no. 4, pp. 1239–1242, 2021.
- [15] S. Navaneethan, N. Nandhagopal and V. Nivedita, "An FPGA-based real-time human eye pupil detection system using e2v smart camera," *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 2, pp. 649–654, 2019.
- [16] M. Shafiqul Azam and H. K. Rana, "Iris recognition using convolutional neural network," *International Journal of Computer Applications*, vol. 175, no. 12, pp. 24–28, 2020.
- [17] W. H. Bangyal and J. Ahmad, "Optimization of neural network using improved bat algorithm for data classification," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 1, pp. 669–680, 2019.
- [18] H. T. Ngo, R. Rakvic and P. Broussard, "Resource-aware architecture design and implementation of hough transform for a real-time iris boundary detection system," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 5, pp. 485–492, 2014.
- [19] V. Kumar, A. Asati and A. Gupta, "Iris localization in iris recognition system: Algorithms and hardware implementation," Ph.D. Dissertations, Birla Institute of Technology and Science, India, 2016.
- [20] A. Joseph, "An FPGA-based hardware accelerator for iris segmentation," Ph.D. Dissertations, Iowa State University, Iowa, 2018.
- [21] J. Bigun, "Iris boundaries segmentation using the generalized structure tensor," in *Proc. of the IEEE Int. Conf. on Biometrics: Theory, Applications, and Systems*, Washington, DC, pp. 426–431, 2012.
- [22] O. Nadia, B. Garcia, S. Sonia and Dorizzi, "OSIRIS: An open source iris recognition software," *Pattern Recognition Letter*, vol. 82, no. 2, pp. 124–131, 2015.

- [23] A. Uhl and P. Wild, "Weighted adaptive hough and ellipso polar transforms for real-time iris segmentation," in *Proc. of the IEEE Int. Conf. on Biometrics*, New Delhi, India, pp. 283–290, 2012.
- [24] C. Rathgeb, A. Uhl and P. Wild, "*Iris Biometrics: From Segmentation to Template Security*," USA: Springer, 2013. [Online]. Available: <https://www.springer.com/gp/book/9781461455707>.
- [25] L. Masek, "Recognition of human iris patterns for biometric identification," M.S. Dissertation, School of Computer Science and Software Engineering, University of West Australia, Perth, Australia, 2003.
- [26] A. Gangwar, A. Joshi, A. Singh and F. Alonso, "Irisseg: A fast and robust iris segmentation framework for non-ideal iris images," in *Proc. of the IEEE Int. Conf. on Biometrics*, New Delhi, India, pp. 1–8, 2018.
- [27] S. Hashemi, N. Anthony, H. Tann, R. Iris Bahar and S. Reda, "Understanding the impact of precision quantization on the accuracy and energy of neural networks," in *Proc. of the IEEE Design, Automation, and Test in Europe Conf. and Exhibition*, Dresden, Germany, pp. 1474–1479, 2016.
- [28] M. Arsalan, R. Naqvi, D. S. Kim, P. H. Nguyen, M. Owais *et al.*, "Irisdensenet: Robust iris segmentation using densely connected fully convolutional networks in the images by visible light and near-infrared light camera sensors," *Sensors*, vol. 18, no. 5, pp. 1501–1521, 2018.