IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 13, 2020, accepted January 3, 2021, date of publication January 19, 2021, date of current version January 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3052851

FRCNN-GNB: Cascade Faster R-CNN With Gabor Filters and Naïve Bayes for Enhanced Eye Detection

AHMED KHUDHUR NSAIF[1], SAWAL HAMID MD. ALI[1], (Member, IEEE), KHIDER NASSIF JASSIM[2], ASAMA KUDER NSEAF[3,4], RIZA SULAIMAN[4], AMMAR AL-QARAGHULI[5], OMAR WAHDAN[6], AND NAZRUL ANUAR NAYAN[1]

[1]Department of Electrical, Electronic, and Systems Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia
[2]Department of Statistics, Faculty of Management and Economics, University of Wasit, Al-Kut 52001, Iraq
[3]Centre for Advanced Manufacturing and Design Technologies (CAMDT), Sheridan College, Oakville, ON L6H 2L1, Canada
[4]Institute of IR4.0, Universiti Kebangsaan Malaysia (UKM), Bangi 43600, Malaysia
[5]Faculty of Applied Science and Technology, Sheridan College, Oakville, ON L6H 2L1, Canada
[6]Digital Business Department, Humber College, Toronto, ON M9W 5L7, Canada

Corresponding authors: Sawal Hamid Md. Ali (sawal@ukm.edu.my) and Ahmed Khudhur Nsaif (ahmed.eng.communication@gmail.com)

This work was supported by Universiti Kebangsaan Malaysia, under the Grant Code GPK-C19-2020-013.

ABSTRACT Research into biometric identification technologies has evolved in recent years, as most secure facilities and applications are now based on digital technology. Among the available biometric identification technologies is eye detection. The relevance and impact of the use of eye detection in a variety of biometric authentication systems are very high. The main problems associated with the accuracy of eye detection methods are occlusion or reflections from glass. In view of this, we propose a hybridized and enhanced eye detection method that uses a faster region-based convolutional neural network with Gabor filters and naive Bayes (FRCNN-GNB) model to address the problems associated with eye detection schemes. The proposed method consists of four components: convolution layers, a region proposal network, a detection network, and a decision model. The enhancement method is based on a cascade Faster R-CNN with Gabor filters and the naïve Bayes model, in which the initial bounding boxes of the eye region are detected using Faster R-CNN and the decision step is carried out using Gabor filters and the naïve Bayes model to determine which of the bounding boxes belong to the eye region. Experiments on the proposed FRCNN-GNB eye detection scheme are performed on the CASIA-IrisV4 database, and show that the accuracy in terms of eye detection is 100%. The results of the study demonstrate the efficiency of the proposed solution.

INDEX TERMS Eye detection, faster R-CNN, Naïve Bayes model, object detection.

I. INTRODUCTION

Object detection systems are designed to detect the occurrence of objects of certain classes within digital images (and videos). The object identification system discovers which objects appear in a specific image, and where [1]–[3]. Object detection systems play a vital role in face recognition, eye detection, biometric authentication, car tracking, security surveillance, activity detection, object recognition, and many other applications. With the advent of large visual recognition datasets, object categorization [4] and object detection [5] have been under continuous development, and advancements have been made in convolutional neural networks (CNNs) [6]. Human eyes are essential voluptuous organs, and it features wealthy of a face. Eye detection has started to become a significant technological aspect of computer vision with the advent of biometric recognition systems. Eye detection is a technology that determines the positions of the eyes based on abstract features of the images [7].

Localization of the eye or extraction of the iris is an important strategy in the fields of iris recognition, interpretation of expressions, eye tracking, behavior recognition, and face recognition [8], [9], among others. Particularly in the detection of driver fatigue, the motion characteristics and positions of the eyes are used as a basis for the extraction of features related to tiredness, such as blinking frequency, eye gaze

The associate editor coordinating the review of this manuscript and approving it for publication was Mingbo Zhao.

detection, percentage eyelid closure time, and so on. [10]. In an iris recognition system [11], the first step in the entire process is to determine the location of the eyes, and the pupil and iris can then be segmented. Meanwhile, eye detection can also get the scale of iris. Tiredness detection systems similarly seek to determine the locations of the eyes, and then to apply a model of the eyes to compute the number of blinks. The accuracy of eye detection directly influences the precision of iris recognition and biometric applications. Eye detection is facultative in face detection systems, which can increase the precision of detection of a face when detection of the eye is reliance. In view of the demand for the applications discussed above, several eye detection algorithms have been proposed based on the attributes of the eyes. These algorithms and procedures consist of two steps: a model is first constructed based on manual selection of the eye attributes, and this model is then utilized to classify and locate the eyes. Following the emergence of neural networks, object detection can be done automatically using a convolutional neural network (CNN) to determine the attributes, and in 1992 [12], an eye detection method was suggested based on the geometry of the eyes.

In 2005 [13], appearance-related features were used to detect eyes. A human eye localization algorithm based on features textile was proposed in 2011 [14]. In recent years, systems started to locate the eyes based on facial landmarks and have given good results. However, when the image is affected by reflection by glass or is occluded, algorithms based on features textile, geometry, and appearance have shown very poor performance. When an eye is detected using facial landmarks, the image may include partial faces that would be arising failure to detect faces and thus lead to failure to locate the eyes. Under non-ideal conditions, eye detection still poses problems, since the precision of the eye detection method is inevitably affected by reflections from glass, image resolution, individual differences, occlusion, illumination, and other factors. Recently, deep CNNs (convolutional neural networks) have given state-of-the-art results when applied to the many tasks of computer vision; however, the use of CNNs in eye research is limited due to a lack of large datasets.

To address the problems associated with eye detection systems, we propose an improved method of eye detection based on a cascade Faster R-CNN with Gabor filters and a naïve Bayes model (FRCNN-GNB), which locates the positions of the eyes with increased precision even in pictures which contain only partial faces.

This article provides answers to the following research questions:

- Why is it important to study eye detection in the context of the task?

- How can an eye be located within an image?

- How can an improved eye detection method be developed in order to achieve optimum performance?

Our research work makes the following new contributions to eye detection research:

- We develop a system based on a convolutional neural network (CNN) for eye detection.

- After detecting the bounding box in the image via Faster R-CNN, which represents the initial eye region, the features of the initial eye region are extracted using Gabor filters, and the naïve Bayes model is then applied to determine the final eye region.

Compared to other eye detection methods proposed in previous works, our FRCNN-GNB eye detection scheme has the following advantages:

- The proposed method does not rely on the method selected for facial detection and the scale of the eye patch cropped.

- It can be applied even to pictures of faces that are severely occluded or with local pictures view of the facials with eyes when it fails the method of facial detection.

- It does not rely on illumination when detecting eyes. In addition, it is not dependent on the distance or alignment between the centers of the right and left eyes, and can detect both right and left eyes without alignment.

The remainder of the paper is structured as follows. Related works on methods and algorithms for eye detection are discussed in Section 2, and a thorough overview of the proposed FRCNN-GNB eye detection method is given in Section 3. Section 4 presents an experimental study of the proposed scheme, and the final section summarizes and concludes the paper.

II. RELATED WORK

A. CLASSICAL EYE DETECTION METHODS

The detection of objects is an important research topic within the field of computer vision. Conventional object detection frameworks building features and classifiers manually via training, learning through big data and making end-to-end workout training. In the domain of conventional computer vision, object detection algorithms typically have three parts: selection of the detection windows, selection of features and building of the classifier. Detection window selection algorithms have sophisticated through the sliding window dependent on edge box or scale selective search that is more efficacious in generating region proposals due to the use of diverse features. There are several conventional methods for feature selection, such as histogram of oriented gradients (HOG), local binary patterns (LBP), and so forth [15]. In many object detection systems, a decision tree or support vector machine (SVM) is used as a classifier.

Conventional methods for eye detection can be basically classified into three distinct categories: appearance-based, shape-based, and synthesis approaches. Appearance-based approaches usually involve the matching of models, in which a model of an eye is created, and similarity measurements are used to detect (locate) the eyes in the image.

Hallinan [16] constructed a model of an eye by detecting valleys and peaks in intensity and decreasing the energy function to match the images. Rao [17] identified eye and non-eye patterns, which were represented using a Gabor filter, and a shallow neural network was then trained as a classifier. Despite changes in illumination and rotations through tiny angles, their model could be applied. However, it ignores this approach the image's spatial information and thus is difficult to transact the translation contrast. Huang and Wechsler [18] used an optimum wavelet bundle to capture the most important characteristics of the eye area and functions of the radial base that can be identified as eye regions or otherwise from regions of the face.

Wang *et al.* [19] suggested an approach involving non-parametric discriminant analysis for eye detection. Typically, appearance-based approaches require large training datasets of eyes, with different head poses and types of illumination.

One existing model of the iris, eyelid, and pupil uses a shape-based approach to match test samples. Young *et al.* [20] used a Hough transform to effectively extract the iris and pupils in the form of elliptical contours. Similarly, Lam and Yan [21] extended a distorted mold to parameterize the iris and eyelids, which arise from the intersection of two parabolas and a circle. To improve the eye detection process, an additional four intersecting locations of the eyelids and the iris were suggested. Nevertheless, this approach needs highly primary positions index of the eye form template.

Kawato and Tetsutani [22] used a circle frequency filter to reveal candidate points within eyes and the angle of rotation of the images. Eyes on every side of the discovered dot have detected as tiny darkest bits. Synthesis techniques collect, the methods, strategies of shape and appearance to evolve their respective benefits. Xie *et al.* [23] suggested a partial model based on shape to find the positions of many sub-components and implicitly model the appearance. Ishikawa [24] utilized an active appearance model to combine methods based on patterns of appearance and shape. The shape aspect was built on an active shape model, while the texture was modeled independent of the shape, using principal components analysis. Their paradigm was able to detect and monitor eyes.

When these approaches to eye detection are compared, our work can be considered an appearance-based model that depends on data rather than on the initialization of a good template of an eye. In a novel approach to eye detection, we combine a Faster R-CNN with Gabor filters and the naive Bayes model.

Our FRCNN-GNB model takes advantage of the strong features learned by CNNs, which allows it to better locate the initial eye region in an image of a person at a distance, and the final eye region is then determined using Gabor filters and a naïve Bayes model.

B. EYE DETECTION WITH DEEP MODELS

With the ongoing developments in computing ability, many computer vision tasks have been dominated by the use of deep CNNs. Abstract features can be extracted using deep learning via condensed data and frequent training that can to the crosses betterment of essential information. A region-based neural convolution network has good efficiency, making it the primary algorithm used in the domain of object detection. Nevertheless, due to the considerable benefits of deep learning in the field of computer vision, several deep neural networks have been implemented for eye-associated tasks.

Reinders *et al.* [25] were the first to suggest exploiting a neural network to solve the eye detection problem. They proposed an approach based on multilayer perception to determine the positions of the eyes in an anterior face image, with no requirement for manually crafted features. Krafka *et al.* [26] used combined learning CNNs for inputs from the patch of facial pictures and the facial mask, and then merged this information to infer the gaze.

Huang *et al.* [27] suggested a multi-mission learning algorithm for the appreciation of the eye condition and for detection of the landmarks of the eye. They utilized a coarse-to-fine approach in which CNNs were cascaded in two stages to fine-tune the shape of the eye. Our model can also be viewed as a multi-mission learning system, as we use Faster R-CNN to detect the initial eye regions, and then apply Gabor filters and a naïve Bayes model for the final eye detection.

The region-based CNN (R-CNN) was first proposed for object detection in [28], in an algorithm suggested that demonstrated a high level of accuracy in locating and classifying objects compared to traditional algorithms. However, it involved several feature extractors and there were several requirements for the SVM classifiers, meaning that the training period was lengthy.

Two methods were proposed to alleviate these issues: the SPP-Net [29] and the Fast R-CNN [30]. Rather than feeding each deformed proposal picture zone into the CNN, the Fast R-CNN and the SPP-Net work across the CNN for the input picture whole once exactly.

That every proposal's in detection layers can obtain scores of classes and coordinates after the proposals mapping to the maps of feature for the convolutional ultimate layer. SPP-Net, R-CNN and Fast R-CNN depend on public object proposals of the input, that come from the seeking of selective [6]. However, these algorithms upon being dense computed. Ren's team suggested Faster R-CNN [31] to minimize the computational burden of generating the proposal. Several object detection applications have employed Faster R-CNN since it was proposed. For instance, the authors of [32] proposed a Faster R-CNN approach to face detection. They published their findings based on two commonly utilized benchmarks for face detection: the IJB-A, and the newly released FDDB.

In view of these existing problems in the research area of computer vision, our proposed FRCNN-GNB method uses Faster R-CNN techniques together with Gabor filters and a naïve Bayes model for eye detection. In brief, we use Faster R-CNN to find the initial region of the eye, and then detect the final eye region using Gabor filters and a naïve Bayes model. Importantly, it concentrates our FRCNN-GNB model on eye patches, besides it could been worked together with local show of face picture without methods of alignment of the face.

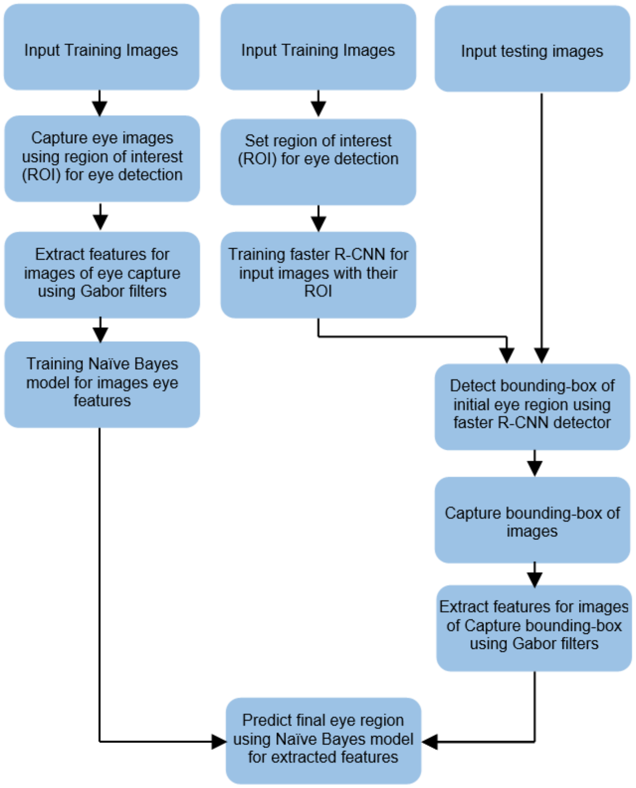

Input Training Images
Capture eye images using region of interest (ROI) for eye detection
Extract features for images of eye capture using Gabor filters
Training Naïve Bayes model for images eye features
Input Training Images
Set region of interest (ROI) for eye detection
Training faster R-CNN for input images with their ROI
Input testing images
Detect bounding-box of initial eye region using faster R-CNN detector
Capture bounding-box of images
Extract features for images of Capture bounding-box using Gabor filters
Predict final eye region using Naïve Bayes model for extracted features

FIGURE 1. Framework of the proposed FRCNN-GNB enhanced eye detection algorithm.

III. PROPOSED FRCNN-GNB APPROACH

A. OUTLINE OF THE PROPOSED METHOD

In this section, we describe the structure of the proposed FRCNN-GNB enhanced eye detection algorithm. As shown in Figure 1, our algorithm is basically divided into two sub-modules: (i) Faster R-CNN, which identifies the bounding box of the initial eye coordinates from pictures, and (ii) Gabor filters and a naïve Bayes model, which predict and determine the final bounding box coordinates of the eye.

B. DETECTION OF INITIAL EYE REGION WITH FASTER R-CNN

1) STRUCTURE OF FASTER R-CNN

Images from the CASIA Iris-Distance database V4 were utilized as the input to the Faster R-CNN [31], and eye regions were detected from the images. Figure 2 illustrates the structure of the Faster R-CNN used here. It can be divided into three sections: a classifier, a region proposal network (RPN), and a feature extractor. It can be seen from the Faster R-CNN process that the map of features is generated after the final convolution layer in the feature extractor, and is used as input for the RPN. The initial region of object detection is then created. The produced proposals pass in the completely linked layer as input, and they are categorized into classes and then given a score.

The Faster R-CNN is configured by utilizing ResNet-50 [33] as the mainstay that has 50 commonly convolutional covers. The images are taken, and the matched proposal regions of the eye as inputs to Faster R-CNN [31] as well as the initial region of the eye of the person are identified. The structure of the Faster R-CNN used here is presented in Figure 2. It is mainly separated into three sections: the classifier, the RPN, and the feature extractor.

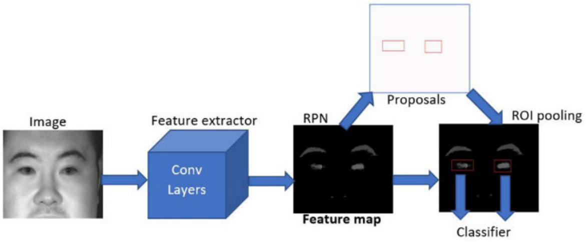


FIGURE 2. Structure of Faster R-CNN.

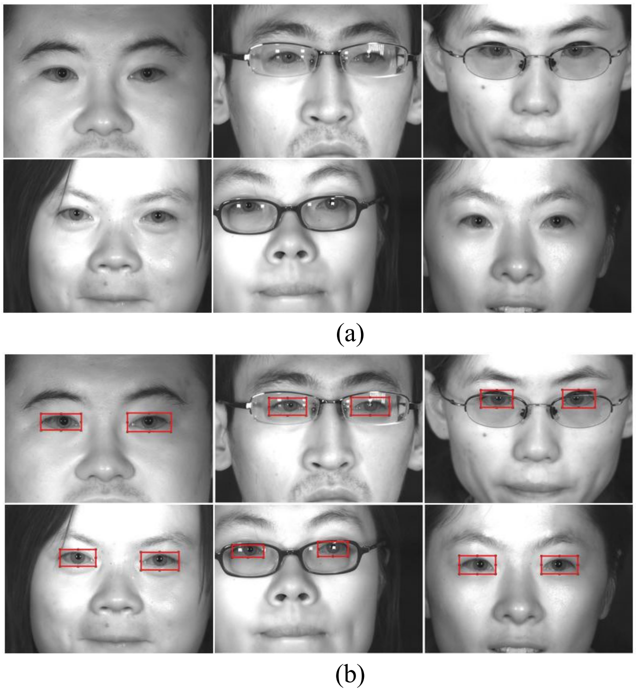
We used a pre-trained ResNet-50 network to extract the features. This network included 50 ReLU layers, four max-pooling layers, and 50 convolution layers. Initially, the categorized image is accepted as input and is passed to the ReLU, convolution and max-pooling layers to obtain the feature map for the final output image. In an input image of dimensions 2352 × 1728 (width × height) pixels, the ROI is set as the area around the eyes within the whole face, which allows the eyes to be identified more efficiently. On the basis of the training data employed in the experiment, the ROI is set taking into account different and shifting locations as the image is of a person at a distance. Some real images from the training dataset are displayed in Figure 3(a), and the matching labels are shown in Figure 3(b). The red bounding boxes are annotations showing the ground truth or ROI.

As shown in Figure 3(b), the ROI is determined on the basis of the initial input image and is cropped to reduce the area from which points can be selected as initial candidates for eyes, thus raising the detection rate.

The size of the final feature maps is 1024 × 51 × 28 pixels (width × height × channel) is based on the ROI of the image. These feature maps are then used in the RPN with the ROI images, and are then passed to the classifier and RPN as input.

The RPN is a network of fully convolutional layers that is used to effectively identify proposed regions with a wide range of aspect ratios and scales that will be passed to the classifier. These regions are represented as rectangular areas that may or may not contain a candidate object. A classifier is used to optimize the proposals, and is the next component of the Faster R-CNN. The RPN and the first component (feature extractor) of the Faster R-CNN detector share the same convolution layers, allowing for concurrent training. The Faster R-CNN is applied only once to the whole input picture via the CNN, and then purify proposals of objects. Due to the use of convolution layers, ResNet-50 can generate high-quality object proposals.

In the RPN during training, while utilizing RPN to foretell proposals of the eye from the maps of features attained from former phase extraction of feature, RPN takes the features maps as input and outputs a collection of candidate


(a)

(b)

FIGURE 3. Examples of ROI in images input to Faster R-CNN.

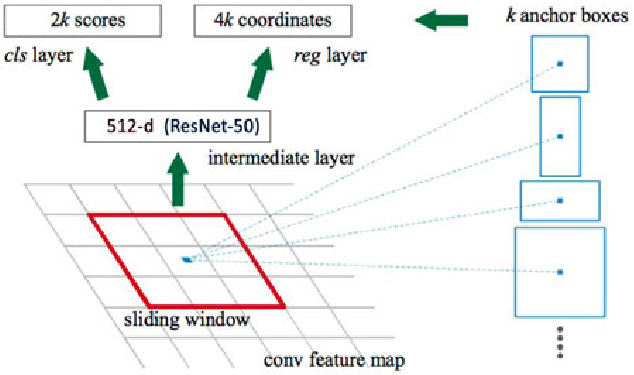


FIGURE 4. Region proposal network (RPN) of the Faster R-CNN [31].

eye regions in the form of rectangular areas (i.e., bounding boxes), every with a score of objectless. In this article, we use a ResNet-50 [33] with 50 shareable convolution layers.

The RPN applies sliding windows to the output of the convolutional feature map via the last shared convolution layer to create rectangular region proposals for each location (see Figure 4). An $n \times n$ filter of the spatial window was convolved with feature map convolutional of the input. Then, each sliding window is overthrown on a feature of lower-dimensional (for ResNet-50 is 512-d), via convolving with two filters of 1×1, respectively, for the box-regression layer (reg) and the box classification layer (cls). In the RPN the key task is to assign the abovementioned 1024-dimensional features to lower-dimensional alternatives (512), and two fully connected layers are then applied: the first stage is responsible for regression, and the second for classification. For each position of the sliding window, k potential proposals (i.e., anchors in [31]) were produced in the classification layer. 4k outputs were created for the regression layer to encode the k coordinates of the bounding boxes.

In the classification layer, output 2k objectness scores were extracted to represent the probability of each proposed region containing an eye or a non-eye object. Since several proposed regions strongly overlap, non-maximum suppression (NMS) was applied to integrate regions with high intersection-over-union (IOU). Based on the object probability score, the remaining proposed regions were ranked using NMS, and the top N propositions were utilized for detection. Anchors were described as candidate regions during the training phase, and the RPN was trained using 256 anchors including 128 negative and 128 positive samples. Furthermore, the eyes likelihood and regression vector of the bounding box [28] are acquired for the anchor boxes to efficiently offer the region of the eye proposals.

Faster R-CNN [31] uses anchor boxes more appropriately for detecting objects, allowing for more precise detection. Equations (1) and (2) are regression vectors of the bounding box ($\mathrm{v_x}$, $\mathrm{v_y}$, $\mathrm{v_w}$, $\mathrm{v_h}$), which are values parameterized of the transformation among the anchor box and the expected box [28]:

$$v_x = \frac{x_p - x_a}{w_a}, v_y = \frac{y_p - y_a}{h_a} \tag{1}$$

$$v_w = \log\left(\frac{w_p}{w_a}\right), v_h = \log\left(\frac{h_p}{h_a}\right) \tag{2}$$

where (x, y, w, h) refer to the midpoint coordinates x, y, and the breadth and height of each box, respectively, and x_p, x_a refer to the midpoint coordinates x of each proposed box and the anchor box, respectively (and so on for$y$, w, and h). Equation (1) displays the scale-invariant interpretation among the midpoint coordinates, while (2) refers to the log-space interpretation between the height and width.

When training RPNs, each proposed region is mapped using a binary class label that indicates whether it is an object (i.e., an eye) or part of the background. Using the regression vector for the bounding box, interpretation of the regions with a more appropriate location and scale could be achieved. The boxes ultimately identified in this way match the connection over the threshold of union (IOU) (i.e., IOU $>$ 0.7), and non-maximum suppression (NMS) was carried out in order to select positive samples proposal boxes directly above the scores of the eye (i.e., object likelihood) standard to gain the candidates of the eye. The proposed region is described as a negative (background) sample if IOU $<$ 0.3 for all ground truth boxes.

In the classifier of the Faster R-CNN [31], the final feature map and the regions identified by the RPN in the previous two phases are used as input. The main role is to regress to the coordinate of eye area and class category. Initially, after collecting the matching position to the boxes of the proposal on the map of the feature, it is passed to the fully connected layer, and candidate regions of various sizes are normalized to the same size (i.e., 7×7) through pooling the ROI. Following this, the feature map of normalized size is passed to the fully connected layer to give the regression vector for the bounding box, and the probability score for containing an eye. In addition, the refined appropriate predict boxes are acquired from the regression vector of the box. NMS was employed to remove the corresponding candidate boxes and consequently give the ultimate detection results. Since the current Faster R-CNN was intended for multi-class detection (i.e. 20 classes), each proposal is categorized as multi-class in the classified section. In the present study, to categorize the outcomes into two types (i.e. background as well as an eye), the nodes of the output were minimized to two in the classification step.

2) LOSS FUNCTION

In the current study, the Faster R-CNN is used to categorize the images into two types (background and object) in conjunction with the RPN. Hence, in every structure of the classifier and RPN, weight is qualified to reduce the function of loss in the mini-batch for every box proposal or anchor box:

$$L\left(p, p^{*}, v, v^{*}\right) = L_{cls}\left(p, p^{*}\right) + sp^{*}L_{reg}\left(v, v^{*}\right) \tag{3}$$

Equation (3) shows the loss function used for the classifiers and RPN. In the RPN, p refers to the likelihood that the anchor box is the object, while p^* refers to the ground truth label (background $=$ 0, object $=$ 1), v refers to the regression vector of the bounding box of the anchor box and v^* refers to the regression vector of the bounding box of the relevant ground truth as well as the anchor box. For the classifier, p is the probability distribution of matching each predict box (for every predict box, $p = (p_0, p_1)$), and p^* refers to the ground truth label (background $= 0$, eye $= 1$). Equation (3) gives a value of one when p^* is not a label of a background region. v is the matching regression vector for the bounding box of each class p^*. v^* refers to the regression vector of the bounding box of the related ground truth as well as matching class. Lcls (classification loss function) and Lreg (regression loss function), each match with log function of loss as well as the robust function of loss (smooth L1) [30]. In Equation (3), the loss of regression takes place just when the ground truth is not the background ($p^* \neq 0$). Consequently, when the ground truth is marked as an eye, loss of categorization and regression with a complementary parameter of weight σ. Continuing through this phase, the weight will be qualified to reduce the value of the loss and the location of the eye is identify in the image. This process was carried out to identify the initial bounding box for the eye.

C. DETECTION OF THE FINAL EYE REGION USING THE DECISION MODEL

When the Faster R-CNN has identified the initial bounding box for the eye, we capture this from the image, as shown in Figure 5, to allow the decision model to predict the final eye region. We use Gabor filters and a naïve Bayes model in our decision model to improve the performance of the eye detection scheme on the basis of reflections; the input for faster module of R-CNN is the entire image, which is large but includes irrelevant areas to exactly locate the eye positions. Then, the initial eye bounding-box of outcomes is taken from the Faster-R-CNN as input to the decision model to decide the final eye region.

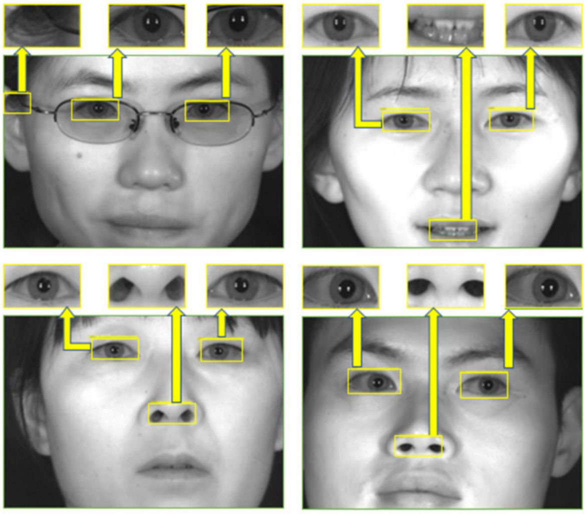


FIGURE 5. Initial bounding box captured for an eye.

The features are then extracted from the image of an eye captured using the initial bounding box, using Gabor filters [34]. The most significant benefit of Gabor filters is their invariance to translation, scaling and rotation. They are also robust against photometric turbulence, such as noise in images and changes in illumination [35]–[38].

The Gabor filter-based characteristics are immediately removed from the gray-level images. In the three-dimensional area, a two-dimensional Gabor filter is a Gaussian kernel function moderated by a multifaceted sinusoidal wave of a plane, which can be expressed as follows:

$$\begin{aligned} G(x, y) &= \frac{f^2}{\pi\gamma\eta} \exp\left(-\frac{x^2+\gamma^2 y^2}{2\sigma^2}\right) \exp\left(j2\pi f x''\right) \\ x' &= x\cos\theta + y\sin\theta \\ y' &= x\cos\theta + y\sin\theta \end{aligned} \tag{4}$$

where f is the occurrences of the sinusoid, θ is the normal to the parallel stripe orientation of a Gabor function, ϕ is the stage counterbalance, σ is the standard deviation of the Gaussian envelope and γ is the three-dimensional ratio of aspects that describe the Gabor function support elasticity.

We use 40 Gabor filters with eight orientations and five scales, as shown in Figure 5. The size of the primary image of the eye area employed in the experiments is 401 × 251 pixels, and with 40 Gabor filters, the dimensions of the feature vector are $401 \times 251 \times 40 = 4,026,040$. Since the neighboring pixels in an image are often closely related, the redundancy of information could be reduced by down-sampling the feature images created by Gabor filters [36], [38]. Down-sampling with a factor of 16 is applied, meaning that the final size of

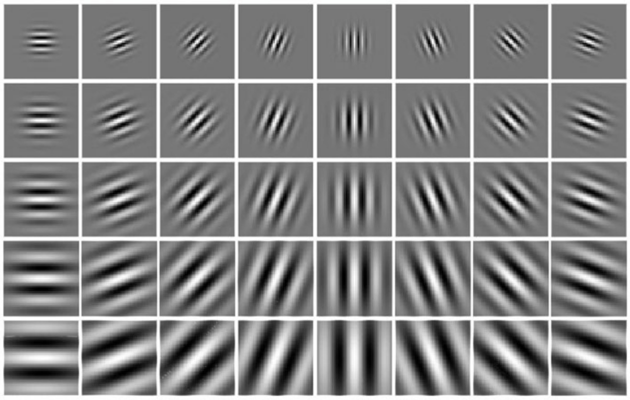


FIGURE 6. Gabor wavelets, with eight orientations and five scales.

the feature vector will be 16,640. The vectors and the unit variance are then normalized to a mean of zero.

The features extracted from the initial eye region are then input to a naïve Bayes classifier to predict the final eye area. The naïve Bayes model is a deeply streamlined Bayesian probability model [39]. The naïve Bayes classifier functions on a solid individuality supposition [39] which means that the probability of one attribute does not influence the probability of another. Given a sequence of n attributes, the naïve Bayes classifier produces 2n! autonomous suppositions.

However, the results of the naïve Bayes classifier are usually true. The study in [40] scrutinized the conditions under which the naïve Bayes classifier shows better performance and showed that faults were caused by three elements: noise in the training data, variance, and bias. The noise could be straightforwardly reduced by selecting appropriate training data, which should be separated into classes by a machine learning algorithm. Bias arises due to groupings in the big data used for training, and variance arises because these groupings are very small.

The naïve Bayes classifier is a simple, probabilistic classifier grounded in the application of the Bayes theorem. We use this approach to train and predict (decide on) the final eye region, using the initial bounding box of the eye area for classification. The first stage of this process is to identify the mean value of the data in Equation 5; there are about 442 images of data that be must categorized using this classifier.

$$\mu = \frac{1}{N} \cdot \sum_{i=1}^{N} x_i \tag{5}$$

where N represents the entire data, and x_i is a individual data value. Employing μ value, the impartial sample variants (δ^2) can be determined as shown in Equation 6.

$$\delta^2 = \frac{1}{N-1} \cdot \sum_{i=1}^{N} (x_i - \mu)^2 \tag{6}$$

Using μ and δ^2, the variants and mean of the training data (the features of the initial bounding box of the eye region) are calculated. Two classes are produced by the naïve Bayes classifier (eye region/other region) which produce eight equations reflecting the variants and means in each class.

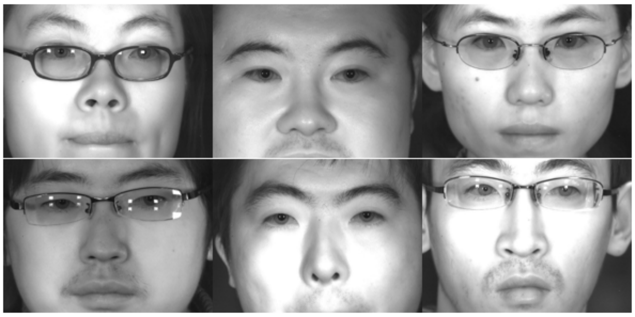


FIGURE 7. Examples of images from the training dataset.

μ and δ^2 must provide the probability density, which can be assessed using Equation 7 by replacing the values of μ and δ^2.

$$f\left(x; \mu, \delta^2\right) = \frac{1}{\sqrt{2 \cdot \pi \cdot \delta^2}} \cdot 3^{-\frac{(x-\mu)^2}{2 \cdot \delta^2}} \tag{7}$$

where f is the class probability density as well as its feature, μ are calculated using Equation 5, and δ^2 are the variants which are calculated using Equation 6. Utilizing Equation 7 will produce many values which form the input values of evidence formulas, as shown in Equation 8.

$$\text{Evidence} = \sum_{i=1}^{n} P \cdot f(x : \mu) \cdot f\left(x : \delta^2\right) \tag{8}$$

p is the probability of class in the data, $f(x : \mu)$ are the means, which are calculated using Equation 7, and $f\left(x : \delta^2\right)$ is the variance, which is also determined by Equation 7. Evidence has employed for divider on the equations to determine posterior values, as shown in Equation (9) below:

$$\text{Posterior}(x) = \frac{\sum_{i=1}^{N} P \cdot f(x, \mu) \cdot f\left(x : \delta^2\right)}{\text{Eridence}} \tag{9}$$

Using Equation 9, values representing two classes can be produced: the eye region and other regions. These two values are subsequently graded to select the highest one, which represents the decision (prediction) of the naïve Bayes model.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

This section presents experimental results for the precision and efficiency of the proposed FRCNN-GNB algorithm for enhanced eye detection. These experiments involved eye detection from 2,567 ultraviolet images of 142 subjects with varying degrees of reflection or occlusion by glass from an open dataset (CASIA-Iris-Distance database, Version 4.0) [41]. Examples of these images are shown in Figure 7. The images were divided into two datasets, which were used for testing and training. A total of 111 images were used as the training dataset and the remainder as the test dataset.

To evaluate the training of the Faster R-CNN with the ResNet model, the training stage consisted of two steps:

(i) after training the RPN model grounded in a pre-trained model of ImageNet, the detection network was trained using the suggested regions created by RPN; (ii) the convolution layer was prepared with the initial phase trained weights as an alternative to the pre-trained ImageNet model, and the RPN was trained for a second time. The suggested regions identified by RPN were then employed to train the whole detection network. The convolution process of the layer accomplishes the sharing weight. Training was implemented using two personal computers with different specifications (PC1 and PC2) and compared with a benchmark system (Benchmark). PC1 had a 3.10 GHz Intel I5-2400 CPU with an NVIDIA GeForce GT 430 1024 MB GPU and 20 GB memory, while PC2 had a 2.9 GHz Intel I7-7500 CPU with an NVIDIA GeForce GTX 950M 2048 MB GPU and 24 GB memory. The Benchmark system used in comparing the computation time is selected to align with the type of tasks performed in this research. The researchers of this benchmark system proposed a deep learning algorithm that used faster RCNN in eye detection but with different layers than ours. The PC used in the benchmark system had a 3.6GHz Intel i7-7700 CPU, NVIDIA GeForce GTX 1070 (1920 CUDA cores and 8 GB memory) GPU and 16GB memory [42] and the calculation of the benchmark system score was based on comparison of the AI scores of PC2 (AI-Score: 2512) and the benchmark system (AI-Score: 14830). As reported in [43], Benchmark of AI computation time depends mainly on the GPU of the system [43]–[45].

During the training phase, the net of detection and the values for the loss and precision of RPN were documented.

In the Faster R-CNN, classifier learning and RPN learning were carried out interchangeably. A four-steps alternating training approach was employed, in which each process was conducted twice [31].

In the initial training stage of the classifier and RPN, end-to-end learning was carried out along with feature extraction. In the second stage of the classifier training and RPN, to share the extractor of the features, the feature extractor was left out and network learning was performed alone. The SGD approach was employed during training with the following parameters: no. of epochs = 6 (in the training of the classifier), base learning rate = 0.001, momentum = 0.9, decay of weights = 0.0005, batch size = 2, gamma = 0.1, batch size = 1, no. of epochs = 4 (in the training of the RPN), base learning rate = 0.001 and gamma = 0.1. At each stage of training, after every interval of iteration (i.e., 555), the trained model at that point was stored; when training was complete, the model with the fewest validation faults of the stored models was chosen and used in the next step in the training process. Nearly 18 hours were needed to apply all of the training steps. Figures 8 and 9 show graphs of four kinds of loss and accuracy function step of training, respectively.

The x-axis represents the number of iterations, while the y-axis represents the values of loss and accuracy. As the iteration number rises, the loss decreases toward a relatively low value. The opposite behavior is seen in the graph of accuracy: as the number of iterations increases, the accuracy decreases towards a relatively high value. This indicates that the Faster R-CNN employed in the present study has been satisfactorily trained.

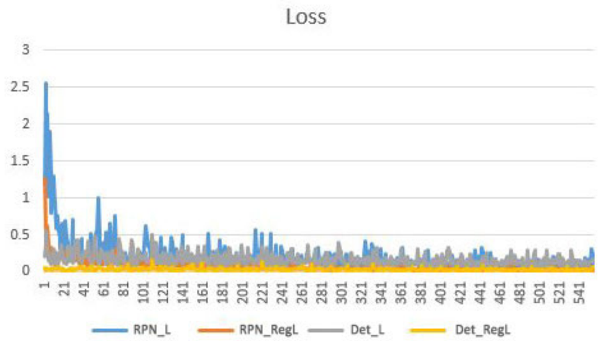


FIGURE 8. Graph of loss during training of the Faster R-CNN.

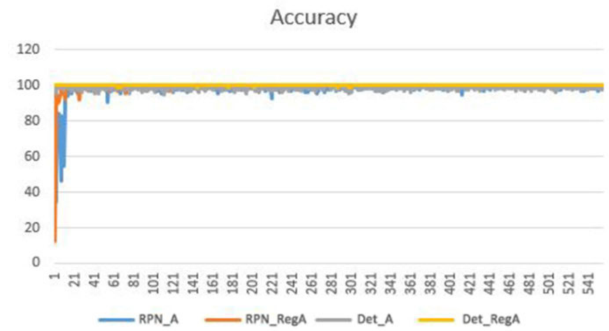
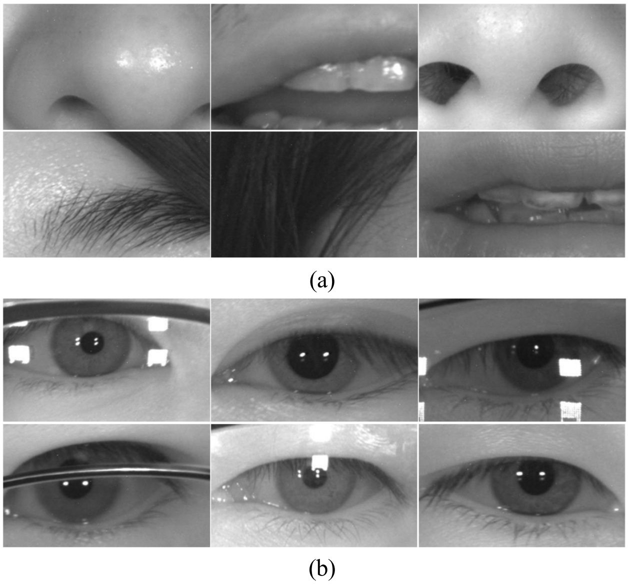


FIGURE 9. Graph of accuracy during training of the Faster R-CNN.

In Figure 8, RPN_L refers to the function of the RPN network categorization loss; RPN_RegL refers to the loss function for the network regression of RPN; Det_L refers to the function of the categorization loss of the entire identification network; Det_RegL refers to the overall loss function for network regression. In Figure 9, RPN_A refers to the precision function of the network categorization of RPN; RPN_RegA refers to the precision function of the network regression of the RPN; Det_A refers to the precision function of categorization of the entire identification network; and Det_RegA refers to the precision function of the entire network regression.

A total of 442 images were used to train the decision model (with a Gabor filter and naïve Bayes model), and these were captured from the same training images on Faster RCNN with a resolution of 401 X 251 pixels. Examples of these images are shown in Figure 10. The 442 images were divided into two classes: 222 belonged to the 'eye region' class, while the remaining 220 images belonged to the 'other region' class. Examples of images from the training dataset are shown in Figure 10.

After training, our proposed FRCNN-GNB model was used for eye detection on the test dataset. In this testing phase, the accuracy of the two models (before and after applying our enhanced method of eye detection) was recorded.


(a)

(b)

FIGURE 10. Examples of images from the training dataset for the decision model: (a) 'eye region' images; (b) 'other region' images.

The performance of the two models was assessed both before enhanced eye detection (i.e. with the Faster R-CNN model) and after application of the enhanced eye detection method (FRCNN-GNB). These networks were compared on the test dataset, and were assessed based on the results of recall and precision.

An analysis of accuracy and recall IOU was employed to assess the precision of object detection. In this description, true positive (TP) refers to the case where the value of the IOU of the identified box is larger than the reference threshold for the IOU and the projected class is congruent with the added information. False positive (FP) refers to the case where the value of the IOU of the identified box is lower than the reference threshold or that the projected class is not in congruence with the added information. A false negative (FN) means that even if the item exists in the image, no box is identified. At a value of 0.5 for the IOU, Table 1 displays the precision and recall for eye detection based on the proposed method (FRCNN-GNB) and on Faster R-CNN. Using the above-mentioned concepts of TP, FN, and FP, the following two criteria are employed to measure the precision [46]:

$$precision = \frac{\text{No.TP}}{\text{No.TP+No.FP}} \tag{10}$$

$$Recall = \frac{\text{No.TP}}{\text{No.P + No.FN}} \tag{11}$$

where No. FP, No.TP, and No. FN refer to the numbers of FP, TP, and FN outcomes, respectively. The maximum and minimum values for the recall and precision are one and zero.

As shown in Table 1, for the Faster R-CNN, the precision is 95.968%, while for the proposed FRCNN-GNB method, the precision is increased by 3.199%.

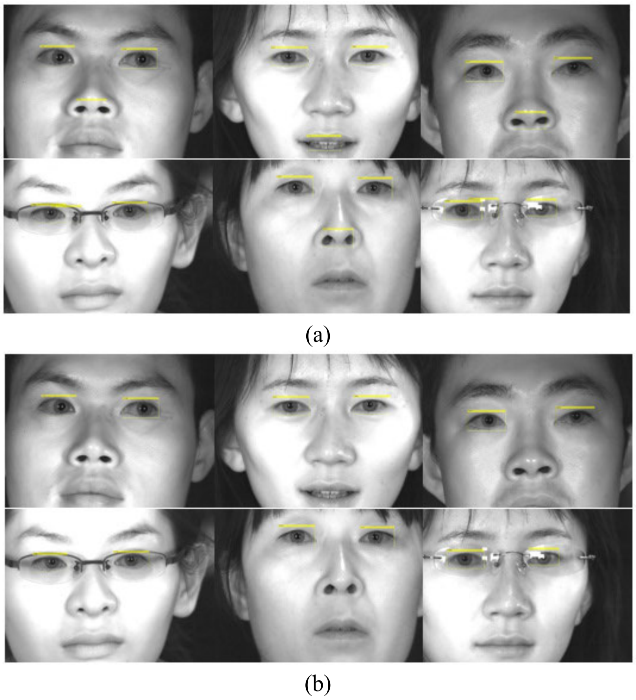
Figure 11 (a) shows several wrong detections using Faster R-CNN, while Figure 11(b) shows the corresponding correct detection using FRCNN-GNB.

TABLE 1. Recall and precision for eye detection using the proposed enhanced method (FRCNN-GNB) and a Faster-R-CNN at an IOU threshold of 0.5.

Algorithm	Precision	Recall
Faster R-CNN	0.95968	0.99167
Proposed **"FRCNN-GNB"** method	0.99167	0.99167

TABLE 2. Comparison of the accuracy and time of eye detection of the proposed FRCNN-GNB model with state-of-art algorithms, on the CASIA v4 Distance database.

Method	Accuracy Rate	Detect Time (s)
Rizon et al. [47]	92.91119%	-
Uhl, Wild [48]	96.4%	1.28
Chai et al. [49]	95.6193%	-
single classifier by Uhl, Wild [48]	65.8%	0.60
nested classifier by Uhl, Wild [48]	14.6%	0.65
Our implemented faster R-CNN [31]	98.21%	0.5915
Our implemented FR-CNN-NB	99.1%	0.5914
Proposed "FRCNN-GNB"	100%	0.5910


(a)

(b)

FIGURE 11. Examples of eye detection using (a) Faster R-CNN and (b) the proposed FRCNN-GNB method.

Table 2 shows the accuracy rates for eye detection on the CASIA v4 Distance database for the proposed FRCNN-GNB method, with alternative methods, including the Faster R-CNN method and our proposed without Gabor filter FRCNN-NB.

In Table 2, we compare the results of the proposed FRCNN-GNB model with the following state-of-art algorithms: Uhl, Wild [48], single classifier by [48], and nested

TABLE 3. Comparison of the computation times of PC1, PC2 and the benchmark system.

Method	Detect Time (s)
Our implemented faster R-CNN [31]	PC1: 22.636 PC2: 3.4922 Benchmark: 0.5915
Our implemented FRCNN-NB	PC1:22.5908 PC2: 3.4913 Benchmark: 0.5914
Proposed "FRCNN-GNB"	PC1:22.4216 PC2: 3.4892 Benchmark: 0.5910

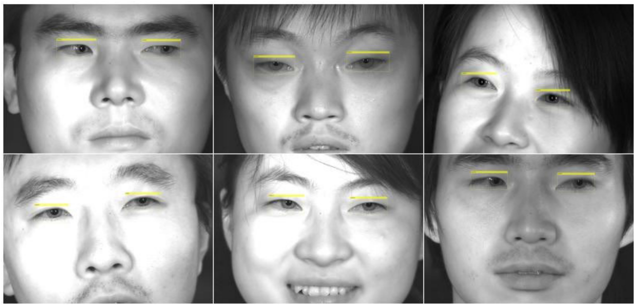


FIGURE 12. Eyes detected with occlusion and angle challenges using proposed FRCNN_GNB method.

classifier by [48] performing eye detection from the face detection result. Kawaguchi and Rizon [47] used intensity valleys to detect the face area and then feature template form valleys were used followed pairs costs to detect eye. Chai *et al.* [49] used morphological processing and flirting to detect the eye. In this article, we implemented a Faster R-CNN [31] method which directly detects the eye from the face image, and also we implemented (cascade Faster R-CNN with Naïve Bayes model "FRCNN-NB") without using Gabor filter.

Uhl and Wild [48] used 2,511 images from a total of 2,567 in the CASIA V4 Iris-Distance database, so when consider 56 images which [48] did not used it the accuracy become 94.2%. Hence, the eye detection accuracy of the model of Chai *et al.* [49] is superior by 2.71% and 1.42% to those of Kawaguchi and Rizon [47] and Uhl and Wild [48], respectively. However, the accuracy of our proposed FRCNN-GNB method is better by 0.9%, 1.79%, 4.38%, and 5.8% than FRCNN-NB, Faster R-CNN [31], Chai *et al.* [49], and Uhl and Wild [48], respectively.

As shown in Table 3, the computation time required for eye detection by the proposed FRCNN-GNB method is lower than for FRCNN-NB and Faster R-CNN [31] by 0.0021 s and 0.003 s, respectively when using PC2, and by 0.1692 s and 0.2144 s respectively when using PC1. Moreover, the runtime of the proposed method is less than FRCNN-NB and Faster R-CNN by 0.0004 s and 0.0005 s respectively when using the benchmark system.

Considering its computation time, the proposed method is capable of real-time eye detection since it is done in fraction of a second. Table 3 shows that there is a big difference in detection time using different systems (PC1, PC2, and benchmark system). This difference is mainly caused by the varying GPU specification in the three systems.

Figure 12 shows example of images illustrating accurate eye detection under challenging conditions such as off-angle and occlusion, using our proposed FRCNN-GNB model.

The proposed FRCNN-GNB method is much more accurate than the current methods of eye detection, as it uses Faster R-CNN to detect the initial bounding boxes for the eye region and then uses a decision model (based on Gabor filters and naïve Bayes) to decide on and locate the correct eye regions, thus achieving the objectives of this research.

V. CONCLUSION

This study proposed an enhanced eye detection model that is resistant to reflection by glasses or occlusion. The proposed method is an attempt to the detection of the eye. It is vitally important that the proposed approach accomplishes prediction of the eye region immediately, without requiring localization of the face. The problem of increasing the precision of eye detection under conditions of reflection from glasses or occlusion was solved by using a cascading Faster R-CNN with a Gabor filter and a naïve Bayes model. It is important to note that the suggested approach initially identifies the eye region as an area of interest and then trains the Faster R-CNN using this region of interest to detect the initial eye region then capture the initial eye region. The results are then passed to a Gabor filter to extract features and then to a naïve Bayes model to determine the final eye region. Through experiments with the CASIA Iris-Distance database V4, we demonstrated the high detection rate of the suggested approach and its high performance in comparison to other detection approaches. Assessments were carried out using MATLAB 2019a, and the presented approach showed high performance. These experiments show that the suggested enhanced approach can be effectively applied to eye detection. The proposed approach combines an object detection algorithm with a decision model (Gabor filter and naïve Bayes) to predict the correct eye region.

REFERENCES

[1] D. A. Dewi, E. Sundararajan, A. S. Prabuwono, and L. M. Cheng, "Object detection without color feature: Case study autonomous robot," *Int. J. Mech. Eng. Robot. Res.*, vol. 8, no. 4, pp. 646–650, 2019.

[2] R. A. K. Noaman, M. A. M. Ali, N. Zainal, and F. Saeed, "Human detection framework for automated surveillance systems," *Int. J. Electr. Comput. Eng.*, vol. 6, no. 2, p. 877, Apr. 2016.

[3] Z. Kadim, M. A. Zulkifley, and N. Hamzah, "Deep-learning based single object tracker for night surveillance," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 4, p. 3576, Aug. 2020.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[6] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Sep. 2013.

[7] F. Yang, X. Yu, J. Huang, P. Yang, and D. Metaxas, "Robust eyelid tracking for fatigue detection," in *Proc. 19th IEEE Int. Conf. Image Process.*, Sep. 2012, pp. 1829–1832.

[8] A. Islam, N. Rahaman, and M. A. R. Ahad, "A study on tiredness assessment by using eye blink detection," *J. Kejuruter.*, vol. 31, no. 2, pp. 209–214, 2019.

[9] Z. R. Mahayuddin and A. F. M. S. Saif, "A comparative study of three corner feature based moving object detection using aerial images," *Malaysian J. Comput. Sci.*, pp. 25–33, Dec. 2019.

[10] Q. Ji and X. Yang, "Real time visual cues extraction for monitoring driver vigilance," in *Proc. Int. Conf. Comput. Vis. Syst.*, 2001, pp. 107–124.

[11] R. P. Wildes, "Iris recognition: An emerging biometric technology," *Proc. IEEE*, vol. 85, no. 9, pp. 1348–1363, Sep. 1997.

[12] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *Int. J. Comput. Vis.*, vol. 8, no. 2, pp. 99–111, Aug. 1992.

[13] M. Hamouz, J. Kittler, J. K. Kamarainen, P. Paalanen, H. Kalviainen, and J. Matas, "Feature-based affine-invariant localization of faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 9, pp. 1490–1495, Sep. 2005.

[14] F. Yang, J. Huang, P. Yang, and D. Metaxas, "Eye localization through multiscale sparse dictionaries," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Mar. 2011, pp. 514–518.

[15] A. F. M. S. Saif, A. G. Garba, J. Awwalu, H. Arshad, and L. Q. Zakaria, "Performance comparison of min-max normalisation on frontal face detection using haar classifiers," *Pertanika J. Sci. Technol.*, vol. 25, pp. 163–171, Jun. 2017.

[16] P. W. Hallinan, "Recognizing human eyes," *Proc. SPIE*, vol. 1570, pp. 214–226, Sep. 1991.

[17] V. Laxmi and P. S. Rao, "Eye detection using Gabor filter and SVM," in *Proc. 12th Int. Conf. Intell. Syst. Design Appl. (ISDA)*, Nov. 2012, pp. 880–883.

[18] J. Huang and H. Wechsler, "Eye detection using optimal wavelet packets and radial basis functions (RBFs)," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 13, no. 7, pp. 1009–1025, Nov. 1999.

[19] P. Wang, M. B. Green, Q. Ji, and J. Wayman, "Automatic eye detection and its validation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR Workshops)*, Sep. 2005, p. 164.

[20] D. Young, H. Tunley, and R. Samuels, *Specialised Hough Transform and Active Contour Methods for Real-Time Eye Tracking*. Brighton, U.K.: Univ. of Sussex, Cognitive & Computing Science, 1995.

[21] K.-M. Lam and H. Yan, "Locating and extracting the eye in human face images," *Pattern Recognit.*, vol. 29, no. 5, pp. 771–779, May 1996.

[22] S. Kawato and N. Tetsutani, "Real-time detection of between-the-eyes with a circle frequency filter," in *Proc. 5th Asian Conf. Comput. Vis. (ACCV)*, vol. 2, 2002, pp. 442–447.

[23] X. Xie, R. Sudhakar, and H. Zhuang, "On improving eye feature extraction using deformable templates," *Pattern Recognit.*, vol. 27, no. 6, pp. 791–799, Jun. 1994.

[24] T. Ishikawa, S. Baker, I. Matthews, and T. Kanade, "Passive driver gaze tracking with active appearance models," in *Proc. 11th World Congr. Intell. Transp. Syst.*, 2004, pp. 1–12.

[25] M. J. T. Reinders, R. W. C. Koch, and J. J. Gerbrands, "Locating facial features in image sequences using neural networks," in *Proc. 2nd Int. Conf. Autom. Face Gesture Recognit.*, Oct. 1996, pp. 230–235.

[26] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba, "Eye tracking for everyone," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2176–2184.

[27] B. Huang, R. Chen, Q. Zhou, and X. Yu, "Eye landmarks detection via two-level cascaded CNNs with multi-task learning," *Signal Process., Image Commun.*, vol. 63, pp. 63–71, Apr. 2018.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[30] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[32] H. Jiang and E. Learned-Miller, "Face detection with the faster R-CNN," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 650–657.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[34] M. Haghighat, S. Zonouz, and M. Abdel-Mottaleb, "CloudID: Trustworthy cloud-based and cross-enterprise biometric identification," *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7905–7916, Nov. 2015.

[35] J.-K. Kamarainen, V. Kyrki, and H. Kalviainen, "Invariance properties of Gabor filter-based features-overview and applications," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1088–1099, May 2006.

[36] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.

[37] S. Meshgini, A. Aghagolzadeh, and H. Seyedarabi, "Face recognition using Gabor-based direct linear discriminant analysis and support vector machine," *Comput. Electr. Eng.*, vol. 39, no. 3, pp. 727–745, Apr. 2013.

[38] L. Shen, L. Bai, and M. Fairhurst, "Gabor wavelets and general discriminant analysis for face identification and verification," *Image Vis. Comput.*, vol. 25, no. 5, pp. 553–563, May 2007.

[39] S. Wafa and R. Naoum, "Development of genetic-based machine learning for network intrusion detection," *World Acad. Sci. Eng. Technol.*, vol. 55, no. 7, pp. 20–24, 2009.

[40] N. Naidu and D. R. V. Dharaskar, "An effective approach to network intrusion detection system using genetic algorithm," *Int. J. Comput. Appl.*, vol. 1, no. 3, pp. 26–32, Feb. 2010.

[41] CASIA-Iris-Distance. (2012). *Chinese Academy of Sciences, Institute of Automation: Biometrics Ideal Test*. [Online]. Available: http://www.idealtest.org/dbDetailForUser.do?id=4

[42] S. H. Park, H. S. Yoon, and K. R. Park, "Faster R-CNN and geometric transformation-based detection of driver's eyes using multiple near-infrared camera sensors," *Sensors*, vol. 19, no. 1, p. 197, Jan. 2019.

[43] A. Ignatov. (2019). *Deep Learning Hardware Ranking*. [Online]. Available: http://ai-benchmark.com/ranking_deeplearning.html

[44] T. Dettmers. (2020). *Making Deep Learning Accessible, Which GPU(s) to Get for Deep Learning: My Experience and Advice for Using GPUs in Deep Learning*. [Online]. Available: https://timdettmers.com/2020/09/07/which-gpu-for-deep-learning

[45] F. Shaikh, "Why are GPUs necessary for training deep learning models? Fact # 101: Deep Learning requires a lot of hardware," Analytics Vidhya, Tech. Rep. Fact 101, 2017. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/05/gpus-necessary-for-deep-learning/

[46] I. D. Melamed, R. Green, and J. Turian, "Precision and recall of machine translation," in *Proc. Companion Volume HLT-NAACL*, 2003, pp. 61–63.

[47] T. Kawaguchi and M. Rizon, "Iris detection using intensity and edge information," *Pattern Recognit.*, vol. 36, no. 2, pp. 549–562, Feb. 2003.

[48] A. Uhl and P. Wild, "Combining face with face-part detectors under Gaussian assumption," in *Proc. 9th Int. Conf. Image Anal. Recognit. (ICIAR)*, 2012, pp. 80–89.

[49] T.-Y. Chai, B.-M. Goi, Y.-H. Tay, and Y.-H. Khoo, "Vote-based iris detection system," in *Proc. 3rd Int. Conf. Digit. Signal Process. (ICDSP)*, 2019, pp. 114–118.

AHMED KHUDHUR NSAIF received the M.Sc. degree in electrical and electronic engineering from Universiti Kebangsaan Malaysia, Malaysia, where he is currently pursuing the Ph.D. degree. His research interests include computer vision, image processing and image analysis, pattern recognition, machine learning, and deep learning.

SAWAL HAMID MD. ALI (Member, IEEE) received the M.Sc. and Ph.D. degrees from the University of Southampton, U.K., in 2004 and 2010, respectively. He is currently an Associate Professor with Universiti Kebangsaan Malaysia (UKM), where he also holds a position of the Head of Programme for Electric and Electronic with the Department of Electric, Electronic and System Engineering, Faculty of Engineering and Built Environment. His current research interests include wearable systems, system-on-chip design, and pervasive computing.

KHIDER NASSIF JASSIM received the B.Sc. and M.Sc. degrees in statistical science and the Ph.D. degree from the University of Baghdad, Iraq. He is currently an Associate Professor with the Department of Statistics, Faculty of Management and Economics, University of Wasit, Al-Kut, Iraq. He is also a Lecturer and the Head of the Department of Statistics, University of Wasit. His research interests include statistics, time series, forecasting techniques, reliability, image assessment, neural networks, machine learning, and data mining.

ASAMA KUDER NSEAF received the M.Sc. degree from the Faculty of Information Science and Technology, National University of Malaysia, and the Ph.D. degree in computing (Doctor Philosophy of visual informatics) from the Institute of Visual Informatics. He is currently a Developer of artificial intelligence embedded system with the Centre for Advanced Manufacturing and Design Technologies (CAMDT), Sheridan College. He is also a Postdoctoral Researcher with the Institute of Industrial Revolution 4.0 (IIR4.0), Universiti Kebangsaan Malaysia (UKM). His research and teaching interests include computer vision, image processing and pattern recognition, image analysis and object detection, artificial intelligence and machine learning with deep learning, statistics, data science, and big data analytics.

RIZA SULAIMAN received the B.Eng. degree (Hons.) in mechanical engineering from the University of Sunderland, U.K., the M.Sc. degree in advanced manufacturing technology from the University of Portsmouth, and the Ph.D. degree in mechanical engineering. He is currently a Professor of visualization and a Senior Research Fellow with the Institute of Industrial Revolution 4.0 (IIR4.0), Universiti Kebangsaan Malaysia (UKM). Before joining the academics, from 1990 to 1995, he had experienced working in the private sector (Engineering). His research interests include visualization, CADCAM, graphics, and medical imaging. He has published three books in computer-aided design. He is also a member of the Institution of Mechanical Engineers (IMechE), U.K., the International Association of Engineers (IAENG), the Board of Engineers Malaysia (BEM) and the Institution of Electrical and Electronics Engineers (IEEE) Malaysia Chapter, and the Malaysia Society for Computer Tomography and Imaging Technology (MyCT). He is also a Reviewer of more than 30 international journals and conference proceedings.

AMMAR AL-QARAGHULI is currently a Professor with the Faculty of Applied Science and Technology, Sheridan College. His research and teaching interests include software development, smart systems, embedded systems, artificial intelligence, and conducting research. He is also passionate about conducting research and leading teams, and has led a number of successful research projects and grants in aforementioned fields.

OMAR WAHDAN received the M.Sc. and Ph.D. degrees from the Centre for Artificial Intelligence Technology Laboratory, National University of Malaysia. He completed one Postdoctoral Fellowship at the Machine Learning Laboratory, Ryerson University. He is currently a Professor with the Digital Business Department, Humber College. His research and teaching interests include big data analytics, business insight, descriptive and predictive analysis, pattern analysis, and feature extraction.

NAZRUL ANUAR NAYAN received the Bachelor of Engineering degree in information and communication engineering from The University of Tokyo, in 1998, and the Master of Engineering degree in electrical and electronics and the Doctor of Philosophy degree in electronics and information systems engineering from Gifu University, Japan, in 2008 and 2011, respectively. He is currently a Senior Lecturer and a Researcher with the Department of Electrical Electronic and Systems Engineering, Faculty of Engineering and Built Environment, Universiti Kebangsaan Malaysia (UKM). He teaches in the area of biomedical signal processing and microelectronics. In addition, he has also gone for a-two year (2014–2016) Post-Doctoral Research programme at the Institute of Biomedical Engineering, University of Oxford, U.K.