Robust Detection of Iris Region Using an Adapted SSD Framework

Saksham Jain[1(✉)] and Indu Sreedevi[2]

[1] Netaji Subhas Institute of Technology, Delhi, India
jain.saksham01@gmail.com
[2] Delhi Technological University, Delhi, India
s.indu@dce.ac.in

Abstract. Accurate detection of the iris is a crucial step in several biometric tasks, such as iris recognition and spoofing detection, among others. In this paper, we consider the detection task to be the delineation of the smallest square bounding box that surrounds the iris region. To overcome the various challenges of the iris detection task, we present an efficient iris detection method that leverages the SSD (Single Shot multibox Detector) model. The architecture of SSD is modified to give a lighter and simpler framework capable of performing fast and accurate detection on the relatively smaller sized iris biometric datasets. Our method is evaluated on 4 datasets taken from different biometric applications and from the literature. It is also compared with baseline methods, such as Daugman's algorithm, HOG+SVM and YOLO. Experimental results show that our modified SSD outperforms these other techniques in terms of speed and accuracy. Moreover, we introduce our own near-infrared image dataset for iris biometric applications, containing a robust range of samples in terms of age, gender, contact lens presence, and lighting conditions. The models are tested on this dataset, and shown to generalise well. We also release this dataset for use by the scientific community.

Keywords: Biometrics · Iris detection · SSD

1 Introduction

Iris recognition plays a major role in modern biometry because the muscular pattern of the iris is unique for all humans, and remains unchanged over time. Detection of the iris region is the first step in iris-based biometric systems, and is important in the performance [6] of the entire pipeline. However, it is still a challenging and time consuming task, with much scope for improvement. As such, we focus only on iris region detection in this paper.

In most iris recognition systems, the subsequent step after iris localisation is normalisation of the isolated iris region, with further processing done on this image [6,11]. Several current methods involve localising the iris of the eye with

C. Arora and K. Mitra (Eds.): WCVA 2018, CCIS 1019, pp. 51–64, 2019.
https://doi.org/10.1007/978-981-15-1387-9_5

a circular boundary. These algorithms are time and computationally inefficient. Moreover, many biometric images usually have partial occlusion of the iris region by either eyelids or eyelashes. This leads to a noisy image, as can be seen in Fig. 1, leading to a restriction in the overall performance.

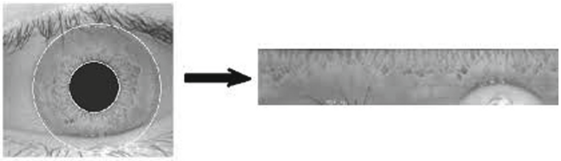

Fig. 1. Noisy normalised image due to partial occlusion by lower eyelid, taken from [11]

However, with the advent of powerful deep learning techniques that have dominated computer vision problems in recent years, CNN-based methodologies can be leveraged to overcome the limitations of the traditional techniques surveyed above. CNNs have shown themselves to be excellent at feature extraction, detection, and recognition [12,14,18], and have already proven their worth in several biometric applications, including iris-based [16].

These techniques perform well under noisy conditions [10,27], and show potential for applications where easier iris region detection might be facilitated by localising the entire iris region, including occlusions and additional noise such as the pupil. Thus, for detection in this paper, we use the smallest square bounding box that completely encapsulates the iris. A novel, lighter and faster framework for detecting the iris region based on the Single Shot Multibox Detector [15] is proposed. It is evaluated and compared with Daugman's [6] algorithm, HOG+linear SVM methodology with sliding window approach [5], and the YOLO [21] network. The proposed framework shows very promising results and speedy, accurate detection despite challenges like noise, occlusion and specular reflections.

The rest of this paper is organised as follows: Sect. 2 presents related work, Sect. 3 summarises the problem, Sect. 4 briefly describes the datasets, Sect. 5 describes the methodology, and Sect. 6 presents the experiments and results. Finally, Sect. 7 presents the concluding remarks of this paper.

2 Related Works

The most well-established technique for iris recognition in use today was given in a seminal paper by Daugman [6]. For iris region detection, it defines an integro-differential operator which fits the circular boundaries of the iris and the pupil by maximising the radial Gaussian via gradient ascent.

This method is modified [28] by applying Hough Transform to a gradient decomposition to approximate the centre of the pupil, while the integro-differential operator fits the iris boundary. In [23] the inner boundary is localised by using the Daugman integro-differential operator, and the outer boundary is modelled using points considered to be the vertices of a triangle inscribed in the circular boundary. This is a faster and computationally cheaper technique than Daugman as it does not involve optimisation.

In [19] the Gabor filter is used to roughly identify the pupil centre, and subsequently, the intero-differential operator localises the iris such that the real centre is in the near vicinity of the rough position of the pupil centre. In, [29], Gabor filters are used for feature extraction and generating a descriptor. Then the proposed probabilistic fuzzy matching scheme is used to compute similarity scores.

In [20], the pupil region is isolated via application of the kNN algorithm on formulated function, and the outer iris region is detected by contrast enhancement and thresholding. In [26], authors present an algorithm which uses the regional properties of the pupil to extract its area and determines the inner iris contour by iterating points, and then comparing and sorting them. Similarly the outer iris contour is determined by an iterative searching methodology, using the pupil centre and approximate radius.

Recently, deep learning based methods have been effectively used in iris recognition systems and related tasks. The authors, in [17], explore the application of pre-trained CNNs to the problem of iris recognition, and demonstrate the effectiveness of their off-the-shelf features for the task. In [2], the authors discuss in detail the network design of a Fully Convolutional Deep Neural Network for iris segmentation, and provide comprehensive comparisons with other methods.

In [1], the authors investigate iris recognition in a visible light environment, and propose a CNN-based method for iris segmentation in the presence of environmental noise of visible light. In [4], a multi-task CNN is proposed to carry out iris localisation, and compute the probability of a presentation attack from the input ocular image. In [24], the authors evaluate baselines for square bounding box location of the iris, and set a benchmark for deep learning-based detectors for the problem.

3 Problem Formulation

Even though Iris based biometric systems are popular, lot of constraints exist. The main challenges faced by current techniques are high computational cost and time consumption for iris detection, inability to deal with scale change, poor performance due to occlusion by eyelashes/eyelid, requirement of iris centring etc. More limitations arise due to ambient conditions such as noise, light reflections etc. In this paper, we propose a modified SSD model uniquely suited for addressing these issues, described in the following sections.

4 Datasets

Four established datasets were chosen from existing biometric applications and literature for this study, namely: Notre Dame Cosmetic Contact Lenses 2013 (NDCLD13) [8], Notre Dame Contact Lens Detection 2015 (NDCLD15) [7], IIIT Delhi Contact Lens Iris (IIITD CLI) [13,30] and CASIA-Iris V3 Interval [3]. An original dataset is also introduced as a part of this study: the IrisDet dataset, which has also been evaluated in this paper.

The NDCLD13 data set contains near-IR images, taken under two sensors. 4200 images (3000 in training set and 1200 in test set) are under the LG4000 sensor and 900 images (600 in training set and 300 in test set) are under the AD100 sensor. The NDCLD15 is an expanded dataset that comprises 7300 images, with 6000 images in the training set and 1300 images in the test set. The IIITD CLI dataset contains a total of 6570 near-IR illumination images taken from 101 subjects using either the Cogent iris sensor or the VistaFA2E sensor. 3000 images comprise the training set (1500 images each corresponding to the two sensors) with the rest comprising the test set for validation and testing. The CASIA-Iris V3 Interval dataset contains 2639 iris images, acquired using a camera that uses circular near-IR LED illumination. All images have a distinctive circular pattern visible in the pupil region. The training set consists of 1500 images, with the remainder being used as the test set. Figure 2 shows sample images.

IrisDet Dataset: This dataset, created during the course of this study, contains 1893 images of the ocular region, taken from 175 subjects, and acquired under near-IR illumination. All images were taken using the IriShield MK2120U single iris camera, and have a resolution of 640 × 480 pixels (Fig. 3). Although not classified, subjects satisfy either of three conditions: no contact lenses, clear contact lenses, and coloured contact lenses. This dataset differs from all others in this study, in that, about half the images are off-centre, and have been taken in varied lighting conditions, depending on the usage of a goggle. This adds more diversity to the training samples, and helps to train more robust models, as demonstrated in Sect. 5. The training set contains 1300 images, with the remaining 593 being used as the test set. We make this dataset and its annotations available to the scientific community.[1]

5 Proposed Methodology

5.1 Network Framework

The proposed iris detection framework is based on the Single Shot MultiBox Detector(SSD) [15], which can be broken down into two simple major steps: extraction of multi-scale feature maps, and application of small convolution filters for object detection. The starting point is the SSD300 variant, wherein the

[1] Print and sign the license agreement available at Saksham Jain's website. Scan and email it to both authors, upon which the download link to the dataset will be sent to the interested researcher.

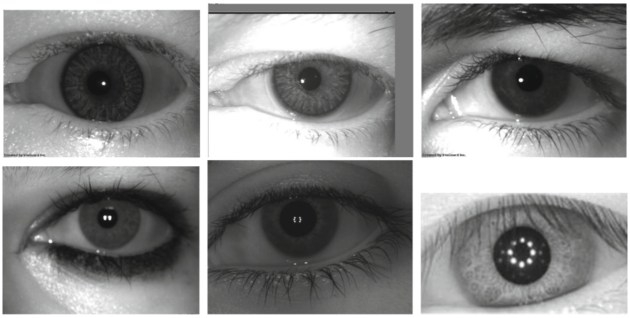

Fig. 2. Dataset sample images (left to right, top to bottom) (i)-NDCLD:AD100, (ii)-NDCLD:LG4000, (iii)-NDCLD15, (iv)-IIITDCLI:Cogent, (v)-IIITDCLI:VistaFA2E, (vi)-Casia-IrisV3

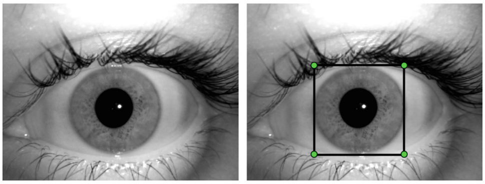

Fig. 3. (i) Shows an image from the introduced IrisDet and (ii) shows manual annotation (https://github.com/tzutalin/labelIm)

image input resolution is 300 × 300, after which the convolutional layers are applied to the image. The SSD benefits from transfer learning, and uses the VGG16 model [25], trained on ImageNet [14], to do so. The architecture of the network is demonstrated in Fig. 4. The early network layers upto Conv 5_3 form the base of the network, and have the transferred VGG16 pre-learned weights. Transfer learning is used because it enables the model to directly obtain the learned "objectness" [9], from the pre-trained network, and thus allows it to successfully learn the iris features from smaller-sized training sets, despite noise or partial occlusion.

The SSD uses multi-scale feature maps for object detection [15], to better handle variation in location, scale and aspect ratio. Different resolution layers are better at detecting objects at different scales. This eliminates the need for the eye to be at a set distance from the camera, except due to inherent camera constraints. These constraints mean however, that biometric cameras capture images of the entire ocular region, meaning that the iris size itself is constrained. This allows for the removal of lower resolution feature maps which are primarily for detecting large-sized objects. Therefore, in the proposed variant (Fig. 4), only the 38 × 38, 19 × 19 and 10 × 10 feature maps are taken as the prediction source layers. This has the added effect of making the model lighter [31], giving it greater speed without much loss of accuracy for this application. All feature

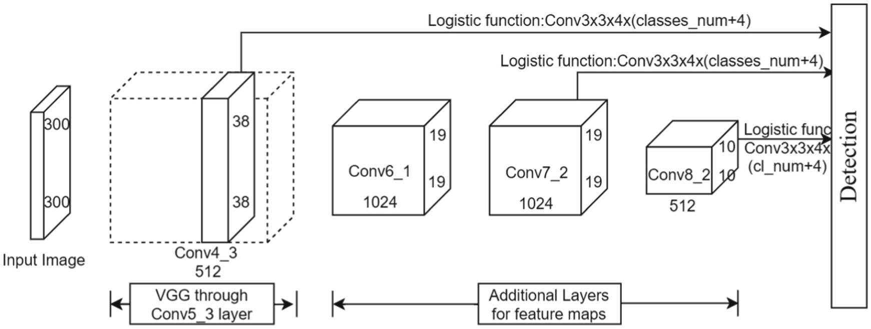

Fig. 4. Architecture of the proposed model

maps contain a certain number of default bounding boxes (discussed in Sect. 5.2) to start off, and bounding box predictions based on them are made.

The objectness scores [9] and the bounding box location offsets (offset between predicted and default boxes) are determined by applying the objectness and location filters respectively to the feature maps. The default boxes are matched to the groundtruth using objectness, and the model is further modified and simplified by a logistic regression layer to binarise the objectness scores.

5.2 Default Bounding Boxes

The HOG+SVM [5] pipeline uses a sliding window strategy for detection, which is limited in terms of speed and is computational cost. Use of region proposals (eg. Faster R-CNN) [22] is much better in terms of both, but prone to mistaking background patches as objects due to inability to contextualise the whole picture. YOLO [21] overcomes this by working in a global context. However YOLO itself has limitations, in that it is spatially constrained on boundary box predictions [21] and somewhat struggles when different scales are involved. SSD, however, overcomes all of these limitations, since it "sees" the whole picture, and adds several feature layers after the base network, after which manually pre-selected bounding boxes are used, as per the requirements of the application.

In the proposed approach, the three feature maps are broken down into a grid formed by 1×1 cells, with the default bounding boxes centred on these cells. There is a single prediction per default box, to keep the number of predictions manageable. These centres are given by [15]:

$$(cx_{def}, cy_{def}) = (\frac{i + 0.5}{|s_n|}, \frac{j + 0.5}{|s_n|})$$

where s_n is the size of the n^{th} feature map, and $i, j \in [0, 1, 2, 3....|s_n|)$. i and j represent the indexes for the default box and matched groundtruth box, respectively. The bounding box location offset, ie. offsets of the predicted bounding boxes to the default boxes for each cell (discussed in Sect. 5.3), is used for box location in place of a global coordinate system.

According to the defined problem, the iris bounding box must be the smallest square bounding the iris. Thus, the aspect ratio of the default bounding boxes is chosen as 1, with the side length of the square determined by the scales, $sc_k \in \{0.1, 0.18, 0.33\}$ selected as per the requirement [15], from the feature map layers. An extra default box with scale, sc $= \sqrt{sc_k \cdot sc_{k+1}}$ is also added for each cell.

5.3 Matching the Default Boxes to the Groundtruth

This step is required so that the groundtruth box can be assigned to a specific default box with which it has the highest IoU (Intersection over Union) value. If the two have a higher IoU value than 0.5 (taken as the threshold in this paper), the default boundary box is considered to be a positive match (the box label is set to 1) otherwise it is a negative match (the box label for is set to 0), due to the logistic decision boundary. Simultaneously, the actual objectness score, and the location offset are also recorded. The objectness score, p always lies between 0 and 1, and since the detection task is solved involving a logistic function layer, the bounding box label x is either 0 or 1. The bounding box location offset is given [15] by:

$$g_j^{cx} = \frac{cx_{gt} - cx_{def}}{a_{def}}$$

$$g_j^{cy} = \frac{cy_{gt} - cy_{def}}{a_{def}}$$

$$a = \log(\frac{a_{gt}}{a_{def}})$$

where (cx, cy) is the matched bounding box centre, j means the same as above, and a is the side of the box. The indexes def and gt respectively denote the default bounding box and groundtruth bounding box.

In case there is conflict where two default boxes are matched with the same groundtruth, the one with the higher IoU value is chosen. Once the positive matches are finalised, the calculated cost function (described in Sect. 5.4) for the corresponding predicted bounding boxes is minimised.

It is natural that far more negative matches are present than positive ones. This can lead to unstable training, due to the resulting class imbalance. Thus, hard negative mining is carried out [15], keeping the ratio of positive matches to negatives at 1:3, for stabler training. This way, the class imbalance can be taken advantage of by having the model learn which predictions are poor. Thus, negative samples during the training phase have a positive impact on actual performance. This is also an added advantage over YOLO, since incorrect localisation is described to be the main source of errors for it [21].

5.4 Loss Function

The loss function or the training objective is a weighted combination of the individual loss functions for the confidence of class prediction, ie. the objectness score, and the bounding box location offsets. The confidence loss can be described by:

$$L_{conf} = -\sum_{i \in Pos}^{N} x_i \log(p_i) - \sum_{j \in Neg}^{N} (1 - x_j) \log(1 - p_j)$$

where, p, x, i, and j hold the same meanings as in the above sections. The location loss can be described by:

$$L_{loc} = \sum_{i \in Pos}^{N} \sum_{m \in (cx,cy,a)} x^m Smooth_{L1}(l_i^m - g_j^m)$$

The location loss is calculated using the Smooth L1 loss, ie. the absolute value loss which is less sensitive to anomalies than L2. Here, l denotes the predicted box offsets, and g denotes the matched groundtruth box parameters. The final loss function [15] is given bye:

$$L_{net} = \frac{1}{N}(\alpha L_{loc} + L_{conf})$$

The value of α is determined via cross validation, and is taken as 1 here.

6 Experiments and Results

In this paper, we evaluate the proposed SSD-based framework, and compare it with the well-established Daugman [6] technique[2], as well as the HOG+SVM and YOLO baselines described in [24]. The implementation of our methodology uses the popular Keras library and is done in python. The experiments are performed on the five datasets mentioned in Sect. 3, on a single Nvidia GeForce GTX 960M GPU accelerated system.

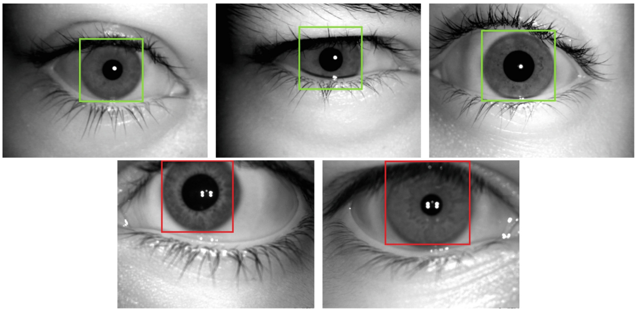

Fig. 5. Positive (green bounding boxes) and Negative (red bonding boxes) Results (Color figure online)

[2] Implementation: https://github.com/Qingbao/iris.

The training and testing splits for the datasets are given in Table 1. We use four standard metrics, namely Accuracy, Precision, Recall, and Intersection over Union (IoU), for the evaluation.

Table 1. Train/test splits for the five datasets

Dataset training/ test split	NDCLD13		NDCLD15	IIIT CLI		Casia	IrisDet
	AD100	LG4000		VistaFA2E	Cogent	IrisV3	
Training set	600	3000	6000	1500	1500	1500	1300
Test set	300	1200	1300	1530	2040	1139	593

Since we are not using any large datasets, we utilise data augmentation such as random expansion, flipping and random cropping the images for improved performance. The proposed method makes use of an Non-Maximum Suppression threshold of 0.5, and only those iris bounding squares are accepted, which have a greater than 0.5 objectness score.

Examples of the detection results obtained with our method are shown in Fig. 5. Most failure cases seem to occur in images where the iris is too close to the image border. We perform four experiments, as described below:

Individually Trained and Tested: In this experiment, all models are trained and tested on images corresponding to the same sensor with the Daugman method being applied to the test sets. The proposed approach gave the best results in all the metrics as well as lesser processing time than the other methods (with only YOLO being the faster). The state of the art results obtained are demonstrated in Table 2.

Collectively Trained and Tested: In this experiment, the models are trained on a combined training set taken from all the datasets, and consequently tested on a similarly combined test set, so as to check the ability of the methods to generalise when more varied training samples are provided. Once again, the proposed scheme outstrips the other methods across all metrics (with YOLO being the closest in performance), showing more robustness. Table 3 shows the results obtained, and Fig. 6 portrays the precision vs recall curve for both our proposed SSD variant and YOLO, and highlights the superiority of the proposed scheme.

Trained on Four, Tested on One: In this experiment, the models are trained on a combined training set (containing 11800 images) taken from any four datasets, and subsequently tested on a single test set taken separately from the remaining dataset, one at a time. Due to much greater variation and amount of training samples, all models generalise well, with our proposed method outperforming the others, as shown in Table 4. However, when tested against the IrisDet dataset, the values across all metrics are relatively lower, which may be attributed to the fact that all other datasets have a very low representation of images in which

Table 2. Trained and tested on same sensor

Metric	Method	NDCLD13		NDCLD15	IIIT CLI		Casia	IrisDet
		AD100	LG4000		VistaFA2E	Cogent	IrisV3	
Accuracy	Daugman	94.28	97.53	96.67	95.38	96.34	97.38	94.74
	HOG+SVM	96.57	96.77	96.83	97.93	96.61	92.23	96.16
	YOLO	98.39	98.68	98.48	98.28	98.19	97.21	96.60
	Proposed	99.31	99.41	99.26	99.62	99.33	98.49	98.26
Precision	Daugman	82.49	92.15	89.80	89.34	92.82	96.23	90.06
	HOG+SVM	94.35	92.72	91.18	92.22	87.99	88.48	86.58
	YOLO	95.12	97.83	95.76	93.71	95.88	96.02	92.65
	Proposed	97.47	99.17	97.22	95.23	97.02	98.19	94.60
Recall	Daugman	84.60	93.41	91.63	85.49	86.24	96.38	95.92
	HOG+SVM	92.39	96.72	96.04	94.51	96.44	96.97	95.13
	YOLO	98.78	97.81	97.28	97.85	96.02	97.79	95.67
	Proposed	99.56	98.26	97.99	98.42	97.51	98.49	96.50
IoU	Daugman	80.41	89.67	85.34	80.82	82.61	90.95	90.12
	HOG+SVM	87.52	87.76	86.85	87.23	84.76	86.17	85.16
	YOLO	93.84	95.66	93.25	91.76	91.84	91.24	90.73
	Proposed	94.25	97.21	94.98	93.67	93.10	92.36	92.04

Table 3. Collectively trained and tested

Method	Training set	Test set	Accuracy	Precision	Recall	IoU
Daugman	–	All five	86.54	86.28	94.04	81.09
HOG+SVM	All five	All five	89.67	90.16	92.71	91.14
YOLO	All five	All five	98.32	95.20	97.13	92.54
Proposed	All five	All five	99.27	96.67	98.91	95.52

the iris is significantly off-centre. This lowers the robustness of such a model in a potential single iris biometric application where proper centering may not be a guarantee.

Trained on One, Tested on Four: In this experiment, we see how well the trained models generalise across datasets. This means that the models are trained on separate training sets consisting of 1800 randomly selected images from each of the five datasets (one at a time), and are subsequently tested on a single test set comprising 600 testing samples from the remaining datasets (each dataset having equal representation). For all methods, there is a fall in the values of each metric, which may be attributed to the differences in the way the images were taken, as well as the inherent differences in the camera sensors and environments. The results are presented in Table 5, and show that while all the metric values are higher under our proposed methodology, they are also relatively higher across all metrics when trained on the IrisDet dataset. Thus demonstrating that IrisDet allows for a higher generalisation capability regardless of the model.

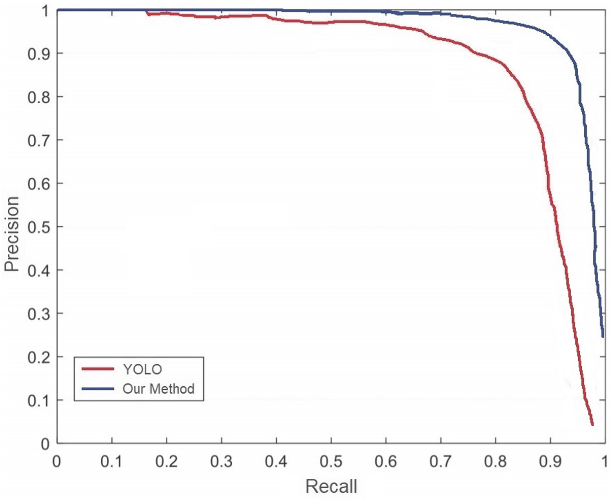

Fig. 6. Precision vs Recall curve comparison for our SSD-based model and YOLO

Table 4. Trained on four datasets and tested on the remaining one

Method	Training set	Test set	Accuracy	Precision	Recall	IoU
HOG+SVM	Others	NDCLD13	92.16	94.26	90.98	92.64
	Others	NDCLD15	92.16	93.12	91.60	94.25
	Others	IIITDCLI	93.45	93.62	94.20	93.56
	Others	CasiaV3	89.96	88.62	89.23	90.13
	Others	IrisDet	94.23	95.32	93.69	94.36
YOLO	Others	NDCLD13	97.68	97.15	98.32	96.98
	Others	NDCLD15	96.85	97.23	96.16	98.32
	Others	IIITDCLI	95.67	95.50	96.68	95.63
	Others	CasiaV3	96.85	98.23	97.96	95.64
	Others	IrisDet	94.13	94.51	93.65	95.67
Proposed	Others	NDCLD13	99.76	98.34	98.68	97.13
	Others	NDCLD15	99.27	98.58	97.24	96.86
	Others	IIITDCLI	99.38	98.95	97.78	96.67
	Others	CasiaV3	98.98	98.85	97.45	97.17
	Others	IrisDet	99.03	98.15	96.89	96.06

For all four metrics, our SSD-based method yields the best results. The processing speed of the proposed method (0.11 s per frame) is also much faster than that of Daugman's (5.20 s per frame) and HOG+SVM (6.72 s per frame), although it loses out to the YOLO detector (0.043 s per frame). However, if a more powerful GPU is used, the detection speed can be increased much further.

Table 5. Trained on a Single Dataset and Tested on Collection of the Remaining Four

Method	Training set	Test set	Accuracy	Precision	Recall	IoU
HOG+SVM	NDCLD13	Others	86.65	85.94	87.25	83.16
	NDCLD15	Others	88.16	87.64	89.32	86.00
	IIITDCLI	Others	89.38	90.42	91.03	87.97
	CasiaV3	Others	82.67	82.88	83.26	80.01
	IrisDet	Others	90.67	91.08	89.89	83.16
YOLO	NDCLD13	Others	94.64	94.23	93.46	91.09
	NDCLD15	Others	94.89	95.08	93.61	90.92
	IIITDCLI	Others	93.64	93.41	92.63	91.39
	CasiaV3	Others	91.08	90.86	90.31	89.84
	IrisDet	Others	94.36	94.03	92.86	91.81
Proposed	NDCLD13	Others	95.75	95.86	95.33	93.14
	NDCLD15	Others	96.95	95.83	96.22	93.10
	IIITDCLI	Others	97.08	95.90	96.04	93.63
	CasiaV3	Others	95.66	94.84	94.13	92.86
	IrisDet	Others	97.87	96.69	96.63	94.14

7 Conclusions

In this paper, we adapt the SSD model for the detection of the iris region, which overcomes several limitations of current techniques such as high computational cost for iris location, and inability to deal with noise, specular reflections, change in scale, poor performance due to occlusion by eyelashes/eyelid, etc. The proposed SSD-based model shows state-of-the-art results, and demonstrates its superiority over existing techniques. Additionally, we introduce and evaluate the IrisDet dataset, which provides the most robust training response. The model, trained on IrisDet, is capable of iris detection on varying scales of ocular images (depending on the eye to camera distance), and also handles off-centre irises. The model shows a lot of potential for extending it to end-to end iris recognition.

References

1. Arsalan, M., et al.: Deep learning-based iris segmentation for iris recognition in visible light environment. Symmetry **9**(11), 263 (2017). https://doi.org/10.3390/sym9110263. http://www.mdpi.com/2073-8994/9/11/263
2. Bazrafkan, S., Thavalengal, S., Corcoran, P.: An end to end deep neural network for iris segmentation in unconstrained scenarios. Neural Netw. **106**, 79–95 (2018)
3. CBSR: Casia-irisv3 image database. http://biometrics.idealtest.org
4. Chen, C., Ross, A.: A multi-task convolutional neural network for joint iris detection and presentation attack detection. In: 2018 IEEE Winter Applications of Computer Vision Workshops (WACVW), pp. 44–51, March 2018. https://doi.org/10.1109/WACVW.2018.00011

5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893, June 2005. https://doi.org/10.1109/CVPR.2005.177
6. Daugman, J.: How iris recognition works. IEEE Trans. Circuits Syst. Video Technol. **14**(1), 21–30 (2004). https://doi.org/10.1109/TCSVT.2003.818350
7. Doyle, J.S., Bowyer, K.W.: Robust detection of textured contact lenses in iris recognition using BSIF. IEEE Access **3**, 1672–1683 (2015). https://doi.org/10.1109/ACCESS.2015.2477470
8. Doyle, J.S., Bowyer, K.W., Flynn, P.J.: Variation in accuracy of textured contact lens detection based on sensor and lens pattern. In: 2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 1–7, Sept 2013. https://doi.org/10.1109/BTAS.2013.6712745
9. Erhan, D., Szegedy, C., Toshev, A., Anguelov, D.: Scalable object detection using deep neural networks. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2155–2162. CVPR '14, IEEE Computer Society, Washington, DC, USA (2014). https://doi.org/10.1109/CVPR.2014.276
10. Fawzi, A., Moosavi-Dezfooli, S., Frossard, P.: The robustness of deep networks: a geometrical perspective. IEEE Signal Process. Mag. **34**(6), 50–62 (2017). https://doi.org/10.1109/MSP.2017.2740965
11. Han, M., Sun, W., Li, M.: Iris recognition based on a novel normalization method and contourlet transform. In: 2009 2nd International Congress on Image and Signal Processing, pp. 1–3, October 2009. https://doi.org/10.1109/CISP.2009.5304768
12. Hinton, G., Vinyals, O., Dean, J.: Distilling the Knowledge in a Neural Network. ArXiv e-prints, March 2015
13. Kohli, N., Yadav, D., Vatsa, M., Singh, R.: Revisiting iris recognition with color cosmetic contact lenses. In: 2013 International Conference on Biometrics (ICB), pp. 1–7, June 2013. https://doi.org/10.1109/ICB.2013.6613021
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, NIPS 2012, vol. 1., pp. 1097–1105. Curran Associates Inc., USA (2012). http://dl.acm.org/citation.cfm?id=2999134.2999257
15. Liu, W., et al.: SSD: Single Shot MultiBox Detector. ArXiv e-prints, December 2015
16. Menotti, D., et al.: Deep representations for iris, face, and fingerprint spoofing detection. IEEE Trans. Inf. Forensics Secur. **10**(4), 864–879 (2015). https://doi.org/10.1109/TIFS.2015.2398817
17. Nguyen, K., Fookes, C., Ross, A., Sridharan, S.: Iris recognition with off-the-shelf cnn features: a deep learning perspective. IEEE Access **6**, 18848–18855 (2018). https://doi.org/10.1109/ACCESS.2017.2784352
18. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1717–1724, June 2014. https://doi.org/10.1109/CVPR.2014.222
19. Radman, A., Jumari, K., Zainal, N.: Fast and reliable iris segmentation algorithm. IET Image Process. **7**(1), 42–49 (2013). https://doi.org/10.1049/iet-ipr.2012.0452
20. Ramkumar, R.P., Arumugam, S.: A novel iris recognition algorithm. In: 2012 Third International Conference on Computing, Communication and Networking Technologies, ICCCNT 2012, pp. 1–6, July 2012. https://doi.org/10.1109/ICCCNT.2012.6396075

21. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You Only Look Once: Unified. Real-Time Object Detection, ArXiv e-prints, June 2015
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031
23. Rodríguez, J.L.G., Rubio, Y.D.: A new method for iris pupil contour delimitation and its application in iris texture parameter estimation. In: Sanfeliu, A., Cortés, M.L. (eds.) Progress in Pattern Recognition, Image Analysis and Applications, pp. 631–641. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-76725-1
24. Severo, E., et al.: A Benchmark for Iris Location and a Deep Learning Detector Evaluation. ArXiv e-prints, March 2018
25. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. ArXiv e-prints, September 2014
26. Su, L., Wu, J., Li, Q., Liu, Z.: Iris location based on regional property and iterative searching. In: 2017 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 1064–1068, August 2017. https://doi.org/10.1109/ICMA.2017.8015964
27. Tang, Y., Eliasmith, C.: Deep networks for robust visual recognition. In: ICML (2010)
28. Tisse, C.L., Martin, L., Torres, L., Robert, M.: Person identification technique using human iris recognition. In: Proceedings of Vision Interface, pp. 294–299 (2002)
29. Tsai, C., Lin, H., Taur, J., Tao, C.: Iris recognition using possibilistic fuzzy matching on local features. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **42**(1), 150–162 (2012). https://doi.org/10.1109/TSMCB.2011.2163817
30. Yadav, D., Kohli, N., Doyle, J.S., Singh, R., Vatsa, M., Bowyer, K.W.: Unraveling the effect of textured contact lenses on iris recognition. IEEE Trans. Inf. Forensics Secur. **9**(5), 851–862 (2014). https://doi.org/10.1109/TIFS.2014.2313025
31. Yi, J., Wu, P., Hoeppner, D.J., Metaxas, D.: Fast neural cell detection using lightweight SSD neural network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 860–864, July 2017. https://doi.org/10.1109/CVPRW.2017.119