A Benchmark for Iris Location and a Deep Learning Detector Evaluation

Evair Severo*, Rayson Laroca*, Cides S. Bezerra*, Luiz A. Zanlorensi*,
Daniel Weingaertner*, Gladston Moreira† and David Menotti*

*Postgraduate Program in Informatics, Federal University of Paraná (UFPR), Curitiba, Paraná, Brazil
†Computing Department, Federal University of Ouro Preto (UFOP), Ouro Preto, Minas Gerais, Brazil

Email: {ebsevero, rblsantos, csbezerra, lazjunior, danielw, menotti}@inf.ufpr.br   gladston@iceb.ufop.br

Abstract—The iris is considered as the biometric trait with the highest unique probability. The iris location is an important task for biometrics systems, affecting directly the results obtained in specific applications such as iris recognition, spoofing and contact lenses detection, among others. This work defines the iris location problem as the delimitation of the smallest squared window that encompasses the iris region. In order to build a benchmark for iris location we annotate (iris squared bounding boxes) four databases from different biometric applications and make them publicly available to the community. Besides these 4 annotated databases, we include 2 others from the literature. We perform experiments on these six databases, five obtained with near-infrared sensors and one with visible light sensor. We compare the classical and outstanding Daugman iris location approach with two window based detectors: 1) a sliding window detector based on features from Histogram of Oriented Gradients (HOG) and a linear Support Vector Machines (SVM) classifier; 2) a deep learning based detector fine-tuned from YOLO object detector. Experimental results showed that the deep learning based detector outperforms the other ones in terms of accuracy and runtime (GPUs version) and should be chosen whenever possible.

Index Terms—Iris location; Daugman detector; HOG & linear SVM; YOLO; Deep Learning.

I. INTRODUCTION

Biometrics systems have significantly improved person identification and authentication, performing an important role in personal, national and global security [1]. In biometry, the iris appears as one of the main biological characteristics, since it remains unchanged over time and is unique for each person [2]. Furthermore, the identification process is non-invasive, in other words, there is no need of physical contact to obtain an iris image and analyze it [3]. Figure 1a illustrates the iris and other structures of a human eye.

Iris location is usually the initial step in recognition, authentication and identification systems [4] and thus can directly affect their performance [5], [6]. In this sense, how the iris location step influences those systems is an interesting question to be studied. For achieving such aim, here, we propose to benchmark/evaluate baseline methods that can be applied to iris location. Initially, we survey some methods in the literature.

The pioneer and maybe the most well known methods for iris location is the one proposed by Daugman [6], which defines an integro-differential operator to identify the circular
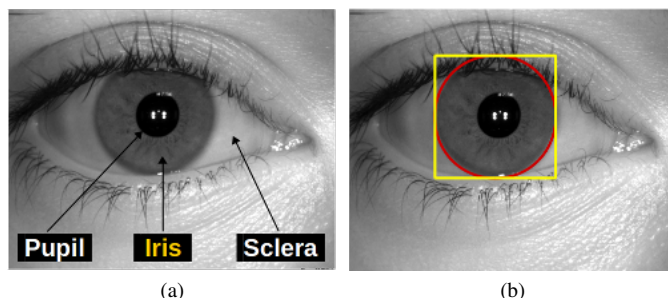

Figure 1. (a) Periocular region and its main structures. (b) Manual iris location through a bounding box and a circle.

borders present in the images. This operator takes into account the circular shape of the iris in order to find the correct position, by maximizing the partial derivative with respect to the radius.

Wildes [5] proposed another relevant method for iris location by using border detection and the Hough transform. First, the iris is isolated by using Gaussian filters of low pass followed by a spatial sub-sampling. Subsequently, the Hough transform is applied and those elements that better fit a circle according to a defined condition are selected.

Tisse et al. [7], present a modification of Daugman's algorithm. This approach applies a Hough transform on a gradient decomposition to find an approximation of the pupil center. Then, the integro-differential operator is applied to locate the iris boundaries. It has the advantage of eliminating the errors caused by specular reflections.

Rodríguez & Rubio [8] used two strategies to locate inner and outer iris contours. For locating the inner contour of the iris, the operator proposed by Daugman is used. Then, for determining the outer boundary of the iris, three points are detected, which represent the vertexes of a triangle inscribed in a circumference that models the iris boundary. This approach presented no better accuracy than the Daugman method, but makes full use of the local texture variation and does not use any optimization procedure. For this reason, it can reduce the computational cost [8].

Alvarez-Betancourt & Garcia-Silvente [9] presented an iris location method based on the detection of circular boundaries under an approach of gradient analysis in points of interest

of successive arcs. The quantified majority operator QMA-OWA [10] was used in order to obtain a representative value for each successive arc. The identification of the iris boundary is given by obtaining the arc with the greatest representative value. The authors reported similar results to those achieved by the Daugman method, with improvements in processing time.

In the method proposed by ZhuYu & Cui [11], the first step is to remove the eyelashes by dual-threshold method, which can be an advantage over other iris location approaches. Next, the facula is removed through erosion method. Finally, the accurate location is obtained through Hough Transform and least-squares method.

Zhou et al. [12] presented a method for iris location based on Vector Field Convolution (VFC), which is used to estimate the initial location of the iris. This initial estimate makes pupil location much closer to the real boundary instead of circle fitting, improving location accuracy and reducing computational cost. The final result is obtained using the algorithm proposed by Daugman [6].

Zhang et al. [13] used an algorithm which adopts a momentum based level set method [14], [15] to locate the pupil boundary. Finally, the Daugman's method was used in order to locate the iris. Determine the initial contour for momentum based level set by minimum average gray level method decreases the time consumption and improves the results obtained by the Daugman's method. This improvement happens because this initial contour, as well as the Zhou et al. [12] approach, is generally close to the real iris inner boundary [13].

Su et al. [16] proposed an iris location algorithm based on regional property and iterative searching. The pupil area is extracted using the regional attribute of the iris image, and the iris inner edge is fitted by iterating, comparing and sorting the pupil edge points. The outer edge location is completed in an iterative searching method on the basis of the extracted pupil centre and radius.

As can be seen, several works in the literature have proposed methods to perform iris location by determining a circle that delimits it (as shown in *red* in Figure 1b), since in many applications it is necessary to perform the iris normalization. Normalization consists in transforming the circular region of the iris from the Cartesian space into a polar coordinate system, so that the iris is represented by a rectangle. Usually, representations and characteristics used on further processes are extracted from the transformed image.

In contrast, with the increasing success of deep learning techniques and Convolutional Neural Networks (CNNs) in computer vision problems [1], [17]–[21], it has become interesting also in iris-related biometrics problems (besides faces) the use of the entire iris region, including the pupil and some sclera region, without the need for normalization.

In this sense, this work defines the iris location task as the determination of the smallest *squared* bounding box that encompasses the entire region of the iris as show in *yellow* in Figure 1b. Thus we propose to evaluate, as baselines, the

following window-based detectors: 1) a sliding window detector based on features from Histogram of Oriented Gradients (HOG) and a linear Support Vector Machines (SVM) classifier, i.e., an adaptation from the human detection method proposed by Dalal & Triggs [22]; 2) a deep learning based detector fine-tuned from YOLO object detector [23], [24].

We compare our results with the well-known method of Daugman [4], since its notoriety and one fair implementation can be publicly found[1]. The experiments were performed in six databases and the reported results show that the use of deep learning to iris location is promising. The fine-tuned model from YOLO object detector yielded real-time location with high accuracy, overcoming problems such as noise, eyelids, eyelashes and reflections.

This paper is structured as follows: Section II presents the databases used in the experiments; Section III describes the baseline methods used in this work; Section IV reports our experiments and discusses our results; Finally, Section V concludes the work.

II. DATABASES

Six databases were used for the experiments performed in this work: IIIT-Delhi Contact Lens Iris (IIIT-D CLI) [25], Notre Dame Contact Lens Detection 2015 (NDCLD15) [26], MobBIOfake [27], Notre Dame Cosmetic Contact Lenses (NDCCL) [28], CASIA-IrisV3 Interval [29] and BERC mobile-iris database [30].

Except the NDCLD15, all other databases were manually annotated from a single annotator[2]. The NDCLD15 annotations were provided by the database authors [26].

Bellow we present a brief description of these databases and how they were used in the experiments.

IIIT-Delhi Contact Lens Iris: The IIIT-D CLI database consists of 6570 iris images of 101 individuals. Three classes of images were used for the composition of the database: individuals who are not using contact lenses, individuals using transparent lenses and individuals using color cosmetic lenses. In order to study the effect of the acquisition device, iris images were captured using two sensors: Cogent iris sensor and VistaFA2E single iris sensor [25].

For the training set, 1500 images of each sensor were randomly selected. The remaining images (3570) were used to compose the test set. All images have resolution of 640×480 pixels and were manually annotated. Figure 2a and Figure 2b show, respectively, examples of images obtained by VistaFA2E and Cogent sensors.

CASIA-IrisV3 Interval - This database consists of 2639 iris images with resolution of 320×280 pixels, obtained in two sections. The images were captured with their own developed camera and an example can be seen in Figure 2f. The main characteristic of this database is that a circular near-infrared led illumination was used when the images were captured, thus this database can be used for studies on the detailing of

---

[1]https://github.com/Qingbao/iris

[2]The iris location annotations are publicly available to the research community at https://web.inf.ufpr.br/vri/databases/iris-location-annotations/

(a) IIIT-D CLI
(VistaFA2E sensor)

(b) IIIT-D CLI
(Cogent sensor)

(c) BERC

(d) MobBIO (Fake)

(e) MobBIO (Real)

(f) CASIA-IrisV3
Interval

(g) NDCCL
(AD100 sensor)

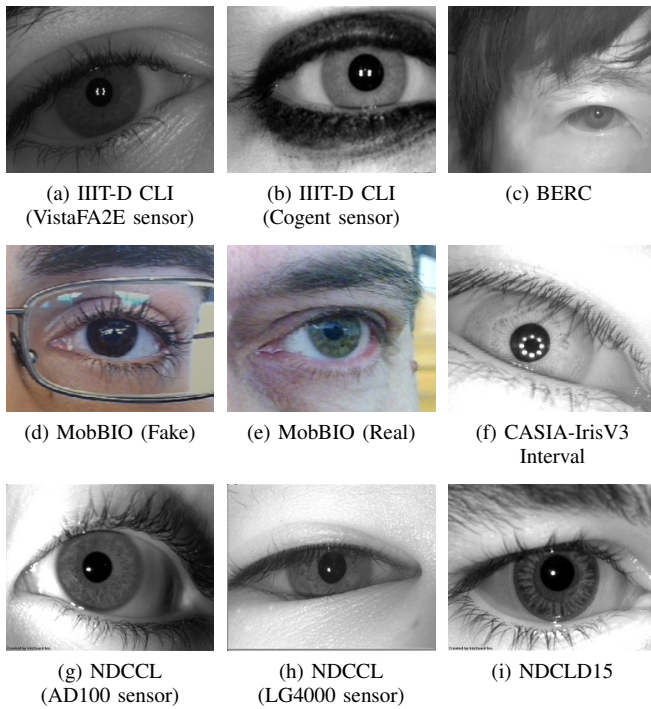(h) NDCCL
(LG4000 sensor)

(i) NDCLD15

Figure 2. Examples of images from the databases used.

texture features in iris images [29]. For training, 1500 images were randomly selected. The remaining images were used for testing.

Notre Dame Cosmetic Contact Lenses - The images from the NDCCL database have resolution of 640×480 pixels and were captured under near-infrared illumination. Two iris cameras were used: IrisGuard AD100 (Figure 2g and IrisAccess LG4000 sensor (Figure 2h), composing two subsets. The IrisAccess LG4000 subset has a training set with 3000 images and a test set of 1200 images. IrisGuard AD100 subset has 600 images for training and 300 for testing [31], [32]. The database contains images of individuals divided into three classes: no contact lenses, non-textured contact lenses and textured contact lenses.

MobBIOfake - The MobBIOfake database was created with the purpose of studying the liveliness detection in iris images obtained from mobile devices in uncontrolled environments [27]. This database is composed of 1600 fake iris images of 250×200 pixels, obtained from a subset of 800 images belonging to the MobBIO database [33].

For the creation of the fake images, the original images were grouped by each subject and a pre-processing was performed in order to improve the contrast. The images were then printed using a professional printer in a high quality photo paper and recaptured using the same device. Finally, the images were cropped and resized to unify the dimensions. The database is equally divided into training and test sets, in other words, 400 real images and 400 fake images were destined for the training sets. Figure 2d and Figure 2e are examples of fake and real images, respectively.

Notre Dame Contact Lens Detection 2015 - The NDCLD15 database is composed of 7300 iris images with resolution of 640×480 pixels. This database is composed of 6000 images for training and 1300 images for evaluation. Images were acquired using either IrisAccess LG4000 sensor or Iris-Guard AD100 sensor. All iris images were captured in a windowless indoor lab under consistent lighting conditions. This database was created with the purpose of studying the classification of iris images between types of contact lenses [26]. Therefore, the database contains images of individuals divided into three classes: no contact lenses, non-textured contact lenses and textured contact lenses. An example image of this database can be seen in Figure 2i.

BERC Mobile-iris Database - The BERC database is composed of images obtained in near-infrared wavelength with a resolution of 1280×960 pixels. The images were captured by a mobile device under vertical position, in sequences composed of 90 images [30]. In order to simulate the situation where the user moves the mobile phone back and forth to adjust the focus, the sequences of images were obtained by moving the mobile phone to the iris at 3 distances: 40 to 15 cm, 15 to 25 cm and 25 to 15 cm. The best images of each sequence were selected, totaling 500 iris images of 100 subjects. An example image of this database can be seen in Figure 2c. In this database, 400 images were randomly selected for training and 100 for testing.

III. BASELINES

In this work, we use two approaches to perform iris location. One of them is based on HOG and SVM, which is an adaptation of the human detection method proposed by Dalal & Triggs [22]. We use this approach together with the sliding window technique presented on the face detection method, proposed by Viola & Jones [34], [35]. The other approach is based on deep learning, using YOLO CNNs [23].

A. Histogram of Oriented Gradients and Support Vector Machines

Despite image acquisition with different devices, lighting conditions, variations of translation, rotation and scale [2], the iris presents a common structure, following patterns of texture, shape and edge orientations, which can be described by a feature descriptor and interpreted by a classifier.

HOG is a feature descriptor used in computer vision for object detection. This method quantizes the gradient orientation occurrences in regions of an image, extracting shape information from objects [22]. Figure 3 illustrates an image described by HOG.

In this work, each window was divided into cells of 8×8 pixels. For each cell, the horizontal and vertical gradients in all pixels are calculated. Thus, the orientations and magnitudes of the gradient are obtained. The gradient orientations are then quantified in nine directions.

In order to avoid effects of light and contrast variation, the histograms of all cells on blocks (2×2 cells) are normalized. The HOG feature vector that describes each iris window is then
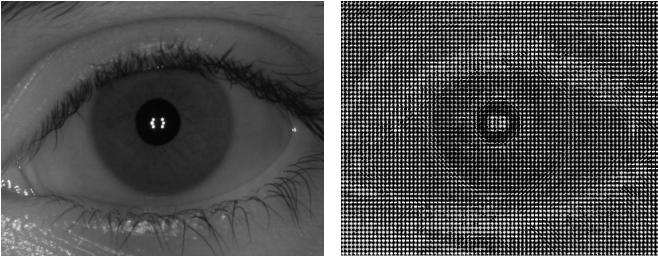
Figure 3. Exemple of image described by HOG.

constructed by concatenating the normalized cell histograms for all blocks. Finally, a feature vector (2×2 blocks $\times$ 8 cells $\times$ 9 orientations) is obtained to describe each iris candidate window.

The window containing the iris region (ground truth) from each training image is extracted and used to compose the examples of positive windows. Furthermore, windows that are completely outside or have only a small intersection with the iris region are extracted and considered negative windows. We created 10 negative windows for each positive window. Figures 4a and 4b illustrate, respectively, positive and negative samples used for the training of the proposed approach.

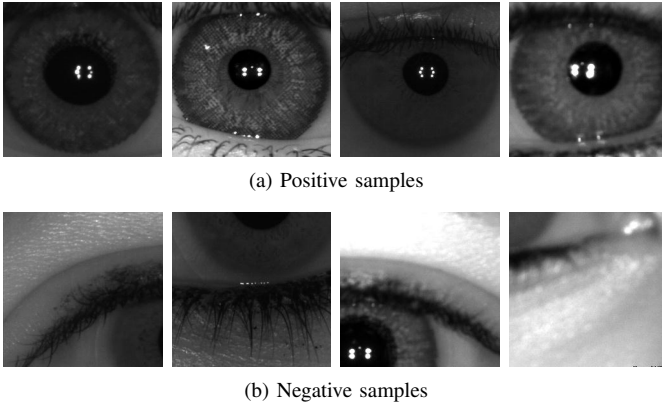
(a) Positive samples

(b) Negative samples

Figure 4. Training samples used by SVM.

From these positive and negative samples, the SVM classifier is trained using a linear kernel and the constant is determined by grid-search in the training set.

The SVM was first presented by Vladimir Vapknik [36], and is one of the most used classification methods in recent years [37], [38]. To find the decision boundary, the SVM minimizes the upper limit of the generalization error, which is obtained by maximizing the margin distance from the training data.

In order to perform the iris location, a sliding window approach with different scales is applied in each test image. We adopted windows with size 50×50 pixels as canonical scale. From this scale, we used 6 lower scales and 8 higher scales by a factor of 5%. The image region that presents the greatest similarity with the iris can be found through the decision border generated by the SVM, which will return the highest positive response for the best estimated iris location.

B. YOLO Object Detector

Currently, deep CNNs are one of the most efficient ways to perform image classification, segmentation and object detection. In this work, we use the Darknet [39], which is an open source neural network framework used to implement YOLO, a state-of-the-art real-time object detection system [23].

The YOLO network, as most CNNs, is composed of three main operation layers to object detection, which are: convolution, max pooling and classification, the latter occurs through fully connected layers.

On Darknet, convolutional layers work as feature extraction, in other words, a convolutional kernel is sliding in the input image. The network architecture is inspired by the GoogLeNet model for image classification [40]. The original YOLO has 24 convolutional layers that produce different feature maps from the input.

The feature maps are then processed by max pooling layers, which dimensionally reduces the previously obtained feature map. max pooling divides the feature map into blocks and reduces each block into one value. Instead of the inception modules used by GoogLeNet, YOLO uses 1×1 reduction layers followed by 3×3 convolutional layers, similar to Lin et al. [41].

However, in this work we use an fast version of YOLO, based on a neural network with fewer convolutional layers (9 instead of 24) and fewer filters in those layers. Other than the size of the network, all training and testing parameters are the same for both YOLO and Fast-YOLO.

IV. RESULTS AND DISCUSSION

In this work, we evaluate both HOG-SVM and YOLO approaches, applied to iris location and compare them to the well-known Daugman method. The experiments were performed in the six databases described in Section II. All experiments were performed on a NVIDIA Titan XP GPU (3840 CUDA cores and 12 GB of RAM) and also using an Intel (R) Core i7-5820K CPU @ 3.30GHz 12 core, 64GB of DDR4 RAM.

In order to analyze the experiments, we employ the following metrics: Recall, Precision, Accuracy and Intersection over Union (IoU). These metrics are defined between the area of the ground truth and predicted bounding boxes in terms of False Positives (FP), False Negatives (FN), True Positives (TP), and True Negatives (TN) pixels, and can be formally expressed as:

$$Recall = \frac{TP}{TP + FN},$$

$$Precision = \frac{TP}{TP + FP},$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN},$$

$$IoU = \frac{TP}{FP + TP + FN}$$

In the following, we describe experiments in three different scenarios: intra-sensor, inter-sensor, multiple-sensors and mixing of databases.

Intra-sensor: Table I shows the results obtained by intra-sensor experiments, in other words, experiments in which the models were trained and tested with images from the same sensor. The YOLO CNN achieved the best averages in almost all analyzed metrics and required less processing time for iris location per image. The exception is for CASIA IrisV3 Interval database where Daugman method presented slightly better Precision (96.23% against 96.02%) and Accuracy (97.38% against 97.10%). This surprising result can be explained by the high level of cooperation and control in the image acquisition of such database. That is, the Daugman method take somehow advantage of the scenario. Anyway, the YOLO CNN locates the iris in real-time (0.02 seconds per image, on average) using our fast Titan XP GPU, whilst the Daugman method and the HOG-SVM approach demand, on average, 3.5 and 5.2 seconds, respectively, to locate the iris in each image using a single CPU core.

Inter-sensor: In addition, for databases containing images acquired with more than one sensor, inter-sensor experiments were performed and are presented in Table II. That is, we train the detectors with images of one sensor and test/evaluate then on the images from other sensor. These experiments show that in some cases YOLO CNN did not achieve promising results as previously shown. For example, in the database NDCCL, when fine tunning/training the detector with images from the AD100 sensor and testing with the ones from LG4000 sensor. The reason for the poor result might lie in the fact that the database for that specific sensor (AD100) has only 600 images, thus not allowing a good generalization of the trained CNN. In Figure 5a, we can observe some examples where the iris location obtained by the YOLO method did not achieved good results.
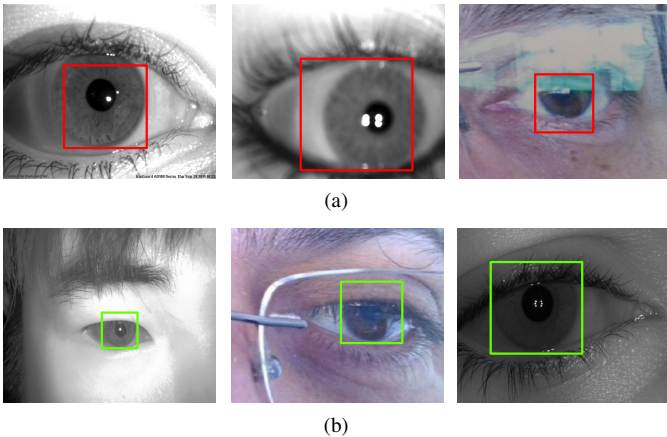

(a)

(b)

Figure 5. Samples of iris location obtained in the experiments: (a) poor results due to a homogeneous training set; (b) good results achieved with images of different sensors on training set.

Multiple-sensors: In order to better analyze and understand the results of the inter-sensor experiments and to confirm our hypothesis that the YOLO's poor performance is due to few/homogeneous training samples, experiments were performed combining images from multiple sensors of the same databases. The figures obtained in this new experiment can be seen in Table III. It highlights the importance of a diverse collection of images for the training set in CNNs. With a larger number of images acquired from different sensors in the training set, the CNN was able to better generalize, increasing the correct iris location in most cases. Some examples of good iris location can be seen in Figure 5b.

Mixing databases: Table IV contains the results obtained by experiments where YOLO was trained with the training sets of all the databases and tested in the test images of all the databases. The results achieved by the Daugman method applied to all the test images are also presented, and we used specific parameters for each database. By analyzing these figures, we observe that YOLO strikingly outperforms the Daugman method in all analyzed metrics.

Figure 6 shows the behavior of the recall curve for the experiment reported in Table IV. It depicts how the percentage of images varies when we required a minimum Recall rate. These curve highlights how YOLO is a promising alternative to iris location, since all tested images achieved Recall values above 80%. That is, at least 80% of the required region of a iris is certainly located by the YOLO detector.
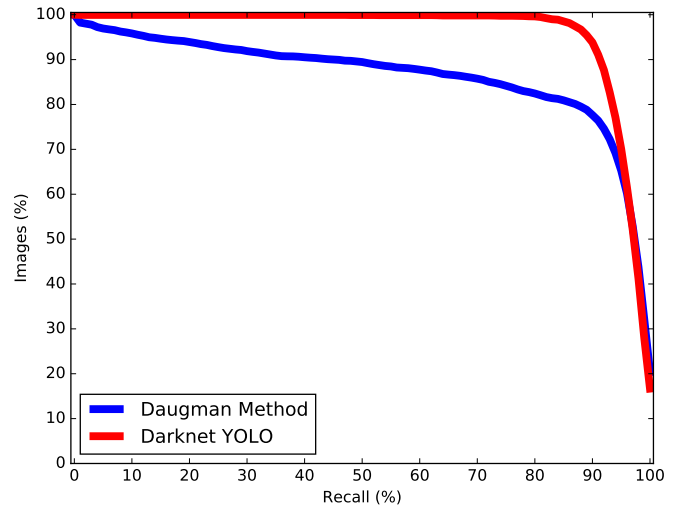

Figure 6. Recall curve of both Daugman and YOLO methods applied to all test sets.

V. CONCLUSION

The iris location is a preliminary but extremely important task in specific applications such as iris recognition, spoofing and liveness detection, as well as contact lens detection, among others. In this work, two object detection approaches were evaluated for the iris location. The experiments were performed in six databases. We manually annotated four of the six databases used in this work, and those annotations are publicly available to the research community.

The experiments showed that the use of the YOLO object detector, based on deep learning, applied to the iris location

TABLE I
INTRA-SENSOR RESULTS (%)

Database	Recall			Precision			Accuracy			IoU		
	Daugman [6]	HOG SVM	YOLO	Daugman [6]	HOG SVM	YOLO	Daugman [6]	HOG SVM	YOLO	Daugman [6]	HOG SVM	YOLO
NDCCL												
AD100	84.60	92.39	**98.78**	82.49	94.78	**95.03**	94.28	96.98	**98.49**	80.41	87.52	**93.84**
LG4000	93.41	96.72	**97.81**	92.15	90.80	**97.73**	97.53	97.24	**99.05**	89.67	87.76	**95.06**
IIIT-D CLI												
Vista	85.49	94.51	**97.85**	89.34	92.24	**93.71**	95.38	98.10	**98.28**	80.82	87.23	**91.76**
Cogent	86.24	**96.44**	96.02	92.82	87.99	**95.58**	96.34	96.67	**98.33**	82.61	84.76	**91.84**
MobBIO												
Real	76.32	95.77	**96.81**	74.71	72.26	**94.02**	85.26	95.33	**98.97**	70.79	68.76	**91.02**
Fake	75.81	93.28	**96.06**	73.45	74.33	**95.05**	84.81	95.26	**98.90**	70.12	68.99	**91.27**
BERC	88.19	92.83	**98.10**	85.64	87.95	**93.56**	98.72	98.49	**99.71**	79.10	85.10	**91.15**
CASIA IrisV3												
Interval	96.38	96.97	**97.79**	**96.23**	88.48	96.02	**97.38**	92.21	97.10	90.95	86.17	**91.24**
NDCLD15	91.63	96.04	**97.28**	89.76	90.29	**95.71**	96.67	97.14	**98.54**	85.34	86.85	**93.25**

TABLE II
INTER-SENSOR RESULTS (%)

Database	Set		Recall		Precision		Accuracy		IoU	
	Train	Test	HOG-SVM	YOLO	HOG-SVM	YOLO	HOG-SVM	YOLO	HOG-SVM	YOLO
NDCCL	AD100	LG4000	**92.95**	79.25	**91.13**	89.18	**96.84**	92.67	**85.78**	68.71
	LG4000	AD100	93.22	**97.99**	93.15	**93.59**	96.78	**97.94**	86.76	**91.63**
IIIT-D CLI	Vista	Cogent	**96.89**	96.13	89.89	**94.21**	96.43	**97.98**	83.94	**90.57**
	Cogent	Vista	93.44	**98.26**	**93.61**	87.97	**97.08**	96.65	**87.55**	80.92

TABLE III
COMBINED SENSOR RESULTS (%), SAME DATABASES

Database	Set		Recall		Precision		Accuracy		IoU	
	Train	Test	HOG-SVM	YOLO	HOG-SVM	YOLO	HOG-SVM	YOLO	HOG-SVM	YOLO
NDCCL	AD100 & LG4000	LG4000	95.37	**99.29**	92.93	**99.68**	97.48	**99.77**	88.63	**98.91**
	AD100 & LG4000	AD100	91.77	**99.37**	94.77	**97.42**	96.85	**99.36**	86.91	**96.85**
IIIT-D CLI	Vista & Cogent	Cogent	96.73	**97.26**	87.15	**96.48**	96.50	**98.49**	84.17	**92.50**
	Vista & Cogent	Vista	94.20	**98.34**	92.74	**93.79**	97.01	**98.55**	87.41	**91.78**

TABLE IV
COMBINED SENSOR RESULTS (%), MIXED DATABASES

Method	Set		Recall	Precision	Accuracy	IoU	Time
	Train	Test					
YOLO	All training sets	All test sets	**97.13**	**95.20**	**98.32**	**92.54**	**0.02 s**
Daugman [6]	-	All test sets	86.45	86.28	94.04	81.09	3.50 s

presents promising results for all studied databases. Moreover, the iris location using this approach runs in real-time (0.02 seconds per image, on average) using a current and powerful GPU (NVIDIA GeForce Titan XP Pascal). Another relevant conclusion to be mentioned is that, similar to other deep learning approaches, it is important to have a sufficiently large number of images for training. The number and variety of images in the training set directly affects the generalization capability of the learned model.

As future work, we intend to perform experiments with more visible and cross-spectral iris databases. In addition, we intend to analyze the impact that iris location exerts on iris recognition, spoofing, liveness, and contact lens detection systems. Also, we plan to study how a short and shallow network than YOLO one can be designed for our single object detection problem, the iris location.

REFERENCES

[1] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 864–879, 2015.

[2] Y. Zhu, T. Tan, and Y. Wang, "Biometric personal identification based on iris patterns," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 2, 2000, pp. 801–804 vol.2.

[3] A. K. Jain, R. Bolle, and S. Pankanti, *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

[4] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148–1161, Nov 1993.

[5] R. P. Wildes, "Iris recognition: an emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, Sep 1997.

[6] J. Daugman, "How iris recognition works," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21–30, Jan 2004.

[7] C.-L. Tisse, L. Martin, L. Torres, and M. Robert, "Iris recognition system for person identification," in *PRIS: Pattern Recognition in Information Systems*, Alicante, Spain, 2002, pp. 71–75.

[8] J. L. G. Rodríguez and Y. D. Rubio, "A new method for iris pupil contour delimitation and its application in iris texture parameter estimation," in *Progress in Pattern Recognition, Image Analysis and Applications*, A. Sanfeliu and M. L. Cortés, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 631–641.

[9] Y. Alvarez-Betancourt and M. Garcia-Silvente, "A fast iris location based on aggregating gradient approximation using QMA-OWA operator," in *International Conference on Fuzzy Systems*, July 2010, pp. 1–8.

[10] J. Peláez and J. Doña, "A majority model in group decision making using QMA–OWA operators," *International Journal of Intelligent Systems*, vol. 21, no. 2, pp. 193–208.

[11] ZhuYu and W. Cui, "A rapid iris location algorithm based on embedded," in *2012 International Conference on Computer Science and Information Processing (CSIP)*, Aug 2012, pp. 233–236.

[12] L. Zhou, Y. Ma, J. Lian, and Z. Wang, "A new effective algorithm for iris location," in *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Dec 2013, pp. 1790–1795.

[13] W. Zhang and Y. D. Ma, "A new approach for iris localization based on an improved level set method," in *2014 11th International Computer Conference on Wavelet Actiev Media Technology and Information Processing (ICCWAMTIP)*, Dec 2014, pp. 309–312.

[14] G. Läthén, T. Andersson, R. Lenz, and M. Borga, "Momentum based optimization methods for level set segmentation," in *Scale Space and Variational Methods in Computer Vision*, X.-C. Tai, K. Mørken, M. Lysaker, and K.-A. Lie, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 124–136.

[15] Z. Wang, Y. Feng, and Q. Tao, "Momentum based level set segmentation for complex phase change thermography sequence," in *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 12, Oct 2010, pp. V12–257–V12–260.

[16] L. Su, J. Wu, Q. Li, and Z. Liu, "Iris location based on regional property and iterative searching," in *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug 2017, pp. 1064–1068.

[17] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[18] N. Pinto, Z. Stone, T. Zickler, and D. Cox, "Scaling up biologically-inspired computer vision: A case study in unconstrained face recognition on facebook," in *CVPR 2011 WORKSHOPS*, June 2011, pp. 35–42.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[20] Z. Mao, W. X. Yao, and Y. Huang, "EEG-based biometric identification with deep learning," in *2017 8th International IEEE/EMBS Conference on Neural Engineering (NER)*, May 2017, pp. 609–612.

[21] T. Y. Fan, Z. C. Mu, and R. Y. Yang, "Multi-modality recognition of human face and ear based on deep learning," in *2017 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR)*, July 2017, pp. 38–42.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2005, pp. 886–893 vol. 1.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 779–788.

[24] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 6517–6525.

[25] N. Kohli, D. Yadav, M. Vatsa, and R. Singh, "Revisiting iris recognition with color cosmetic contact lenses," in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–7.

[26] J. S. Doyle and K. W. Bowyer, "Robust detection of textured contact lenses in iris recognition using BSIF," *IEEE Access*, vol. 3, pp. 1672–1683, 2015.

[27] A. F. Sequeira, J. Murari, and J. S. Cardoso, "Iris liveness detection methods in mobile applications," in *2014 International Conference on Computer Vision Theory and Applications*, vol. 3, Jan 2014, pp. 22–33.

[28] J. Doyle and K. Bowyer, "Notre dame image database for contact lens detection in iris recognition," 2014.

[29] CASIA-IrisV3 Image Database Center for Biometrics and Security Research (CBSR). [Online]. Available: http://biometrics.idealtest.org/

[30] D. Kim, Y. Jung, K.-A. Toh, B. Son, and J. Kim, "An empirical study on iris recognition in a mobile phone," *Expert Systems with Applications*, vol. 54, pp. 328 – 339, 2016.

[31] J. S. Doyle, K. W. Bowyer, and P. J. Flynn, "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)*, Sept 2013, pp. 1–7.

[32] D. Yadav, N. Kohli, J. S. Doyle, R. Singh, M. Vatsa, and K. W. Bowyer, "Unraveling the effect of textured contact lenses on iris recognition," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 851–862, May 2014.

[33] A. F. Sequeira, J. C. Monteiro, A. Rebelo, and H. P. Oliveira, "MobBIO: A multimodal database captured with a portable handheld device," in *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, vol. 3, Jan 2014, pp. 133–139.

[34] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, 2001, pp. I–511–I–518 vol.1.

[35] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004.

[36] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, ser. COLT '92, 1992, pp. 144–152.

[37] G. Franchi, J. Angulo, and D. Sejdinović, "Hyperspectral image classification with support vector machines on kernel distribution embeddings," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 1898–1902.

[38] P. Ruiz, J. Mateos, G. Camps-Valls, R. Molina, and A. K. Katsaggelos, "Bayesian active remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 4, pp. 2186–2196, April 2014.

[39] J. Redmon, "Darknet: Open source neural networks in C," http://pjreddie.com/darknet/, 2013–2016.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 1–9.

[41] M. Lin, Q. Chen, and S. Yan, "Network in network," *CoRR*, vol. abs/1312.4400, 2013. [Online]. Available: http://arxiv.org/abs/1312.4400