

Accurate Iris Segmentation in Non-cooperative Environments Using Fully Convolutional Networks

Nianfeng Liu, Haiqing Li, Man Zhang, Jing Liu, Zhenan Sun and Tieniu Tan
Center for Research on Intelligent Perception and Computing,
Institute of Automation, Chinese Academy of Sciences, Beijing, P.R. China, 100190
{nianfeng.liu, hqli, manzhang, jingliu, znsun, tnt}@nlpr.ia.ac.cn

Abstract

Conventional iris recognition requires controlled conditions (e.g., close acquisition distance and stop-and-stare scheme) and high user cooperation for image acquisition. Non-cooperative acquisition environments introduce many adverse factors such as blur, off-axis, occlusions and specular reflections, which challenge existing iris segmentation approaches. In this paper, we present two iris segmentation models, namely hierarchical convolutional neural networks (HCNNs) and multi-scale fully convolutional network (MFCNs), for noisy iris images acquired at-a-distance and on-the-move. Both models automatically locate iris pixels without handcrafted features or rules. Moreover, the features and classifiers are jointly optimized. They are end-to-end models which require no further pre- and post-processing and outperform other state-of-the-art methods. Compared with HCNNs, MFCNs take input of arbitrary size and produces correspondingly-sized output without sliding window prediction, which makes MFCNs more efficient. The shallow, fine layers and deep, global layers are combined in MFCNs to capture both the texture details and global structure of iris patterns. Experimental results show that MFCNs are more robust than HCNNs to noises, and can greatly improve the current state-of-the-arts by 25.62% and 13.24% on the UBIRIS.v2 and CASIA.v4-distance databases, respectively.

1. Introduction

Iris segmentation aims to locate valid iris texture regions from noisy background. As the beginning process of iris recognition algorithms, its accuracy has a great impact on the accuracy of subsequent feature extraction and classification. In order to reduce the user cooperation in iris recognition, significant efforts have been devoted to develop at-a-distance and on-the-move systems in the past two decades [8, 17]. The accuracy of traditional iris seg-

mentation algorithms decrease dramatically in such non-cooperative environments due to many adverse factors such as blur, off-axis, occlusions and specular reflections.

Existing iris segmentation methods can be roughly divided into two categories. The first category is boundary-based methods which locate pupillary, limbic and eyelid boundaries to isolate iris texture regions. The second category is pixel-based methods which directly distinguish between iris pixels and non-iris pixels according to the appearance information around pixels.

Two classic boundary-based methods are based on integro-differential operators [5] and Hough transforms [31]. Integro-differential operators exhaustively search over a parameter space for the maximum of contour integral derivative. Hough transforms find optimal curve parameters by a voting procedure in binary edge maps. In cooperated environments, pupillary and limbic boundaries can be fitted as circles or ellipses, while upper and lower eyelid boundaries can be fitted as spline [6] or parabolic curves [31]. Boundary-based methods highly rely on image gradients. The limbic boundaries in near-infrared (NIR) images and pupillary boundaries in visible-light images are often low-contrast, but occlusions and specular reflections present a large number of high-contrast edge points. Consequently, original integro-differential operators and Hough transforms perform poorly in non-cooperated environments. Many noise removal [10, 33], coarse iris localization [28], edge detection [13] and boundary fitting methods [3, 29] have been proposed to improve the robustness and efficiency of boundary-based iris segmentation methods. Active contours are utilized to fit irregular iris shapes but they tend to be trapped by iris textures, occlusions and specular reflections [7, 23].

Iris images contain not only gradient and geometry information, but also plentiful other appearance information, such as intensity, color and texture. Pixel-based methods first extract discriminative appearance features in the neighborhood of pixels, and then construct classifiers for iris pixel classification. Pundlik et al. [21] separate eyelash,

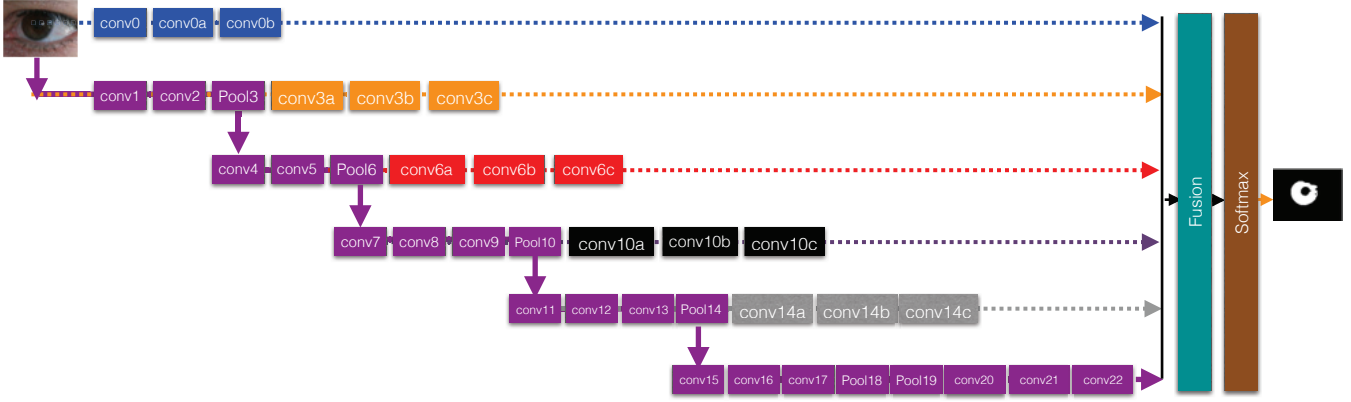


Figure 1. The architecture of MFCNs. The “conv*” is denoted as convolutional layer. The “Pool*” is denoted as pooling layer. The details of the parameters of each layer are shown in Tables 2.

pupil, iris and background through graph cut based energy minimization. Proença [19] design location and color features and employ neural networks with one hidden layer for iris pixel classification. Tan and Kumar [26] extract Zernike moments around pixels and then use support vector machines (SVMs) for classification. Rather than detecting occlusions in original iris images, Li and Savvides [14] use Gabor filters and Gaussian mixture models (GMMs) to classify iris pixels and occlusions in normalized iris texture maps.

Current boundary-based and pixel-based methods are designed mainly based on prior knowledge and need lots of pre- and post-processing. Moreover, their handcrafted features lack flexibility to search optimal descriptors. Deep convolutional neural networks (CNNs) provide a powerful end-to-end model which can automatically learn optimal features and classifiers. CNNs have recently achieved a great success in image classification [12, 24, 25], biometrics spoofing detection [18], face recognition [22] and iris recognition [15]. However, there are two difficulties in applying CNNs to image segmentation. The first one is limited localization accuracy caused by the invariance of CNNs to local spatial transformations. The second one is high computational complexity if we classify each pixel in a sliding window manner. In order to solve these problems, hierarchical CNNs [9, 32] can integrate both high-level and low-level vision cues to improve the localization accuracy. Fully convolutional networks (FCNs) [4, 16] enable pixels-to-pixels prediction and therefore speed up both learning and inference.

In this paper, we propose two models, namely hierarchical convolutional neural networks (HCNNs) and multi-scale fully convolutional networks (MFCNs), to address noisy iris segmentation. We compare these two models and show the priority of MFCNs in both accuracy and efficiency. The proposed MFCNs conduct a dense prediction and gener-

ate labeled maps from input iris images, which avoids repeated computation in overlaps of sliding windows in HCNNs. Both models are end-to-end prediction where features and classifiers are jointly optimized. Note that both models contain no further pre- and post-processing. To combine the fine details of local iris textures and the global structure of eyes, MFCNs fuse multi layers from shallow-and-fine layers to deep-and-coarse layers. We conduct experiments on two well known datasets: UBIRIS.v2 [20] and the CASIA.v4-distance [2] databases to evaluate the performance of the proposed HCNNs and MFCNs. Experimental results suggest that our models can achieve the significant improvements on both iris images acquired under visible light and NIR illumination. The average segmentation errors obtained by MFCNs are improved over the current state-of-the-arts by 25.62% and 13.24% on UBIRIS.v2 and CASIA.v4-distance, respectively.

The remainder of this paper is organized as follows. In Section 2, the MFCNs and HCNNs are described in detail. Section 3 describes our experiments and discusses our results. Finally, the paper is concluded in Section 4.

2. Technical details

Two networks are proposed to address the problem of iris segmentation including hierarchical convolutional neural networks (HCNNs) and multi-stage fully convolutional networks (MFCNs). We compare and analyze two models to show the priority of both the efficiency and effectiveness of MFCNs.

2.1. Hierarchical convolutional neural networks (HCNNs) for iris segmentation

Typical convolutional neural networks are constructed by convolutional layers, pooling layers and fully connected layers. The most famous networks include AlexNet [12], VGGNet [24] and GoogLeNet [25].

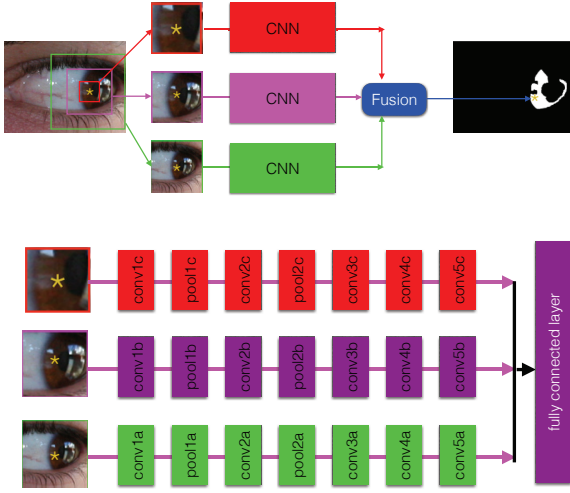


Figure 2. The overall architecture of HCNNs. For each pixel from the image, three patches at three different scales 56×56 , 112×112 and 224×224 are cropped from small to large. Each pixel is separated trained or tested, the predicted label of each pixel is finally merged to an iris mask.

Convolutional layers convolve input images with convolutional kernels, which could be denoted as follows:

$$(Y^k)_{ij} = (W^k * x)_{i,j} + b^k \quad (1)$$

where $*$ is denoted as convolution to an input image x . And k is the index of the convolutional kernels. $(Y^k)_{ij}$ denotes the response of convolution between k -th kernel and pixels centered at (i,j) .

Pooling layers perform non-linear transformations by different down-sampling strategies including max-pooling, mean pooling, and stochastic pooling. The pooling operation outputs a value for each sliding window. It reduces computation for subsequent layers and alleviates overfitting with decreased number of parameters. Max-pooling is frequently used in computer vision tasks.

Fully connected layers connect all the neurons in its previous layer to every neuron it has. It could be denoted as:

$$(y^k) = \sum_f (W^{kf} * x^f)_{i,j} + b^k \quad (2)$$

where f is the index of the f -th neuron of the input. y^k is the k -th output neuron, W^{kf} and b^k is the weights and bias between the two neurons x^f and y^k .

Iris segmentation using HCNNs. In iris segmentation, the label (iris or non-iris) of each pixel is necessarily predicted across the whole input image. Traditional convolutional neural networks could achieve this by patchwise model [9, 32]. Patchwise model takes in the local patches around the pixel to extract features. Each pixel is separately

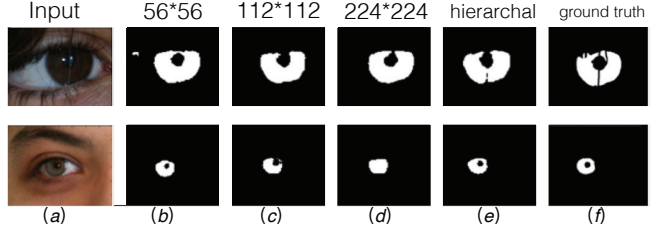


Figure 3. Segmentation results from HCNNs with different size of input patches. Figure (a), (b) and (c) are the results of single-input CNNs with the input patch size of 56×56 , 112×112 and 224×224 , respectively. Figure (e) and (f) are the results of HCNNs and ground truth, respectively.

Table 1. Implementation details of the pooling layers and convolutional layers of HCNNs, where the ‘type’ is the pooling strategy; ‘NO.’ is the number of kernels; ‘size’ is the size of convolution or pooling kernels; ‘stride’ is the stride of convolution or pooling; ‘max’ is max pooling and ‘avg’ is average pooling.

layer	NO.	size	stride	layer	NO.	size	stride	layer	NO.	size	stride
conv1a	48	5	1	conv1b	48	5	1	conv1c	48	5	1
conv2a	128	5	1	conv2b	128	5	1	conv2c	128	5	1
conv3a	192	3	1	conv3b	192	3	1	conv3c	192	3	1
conv4a	192	3	1	conv4b	192	3	1	conv4c	192	3	1
conv5a	64	3	1	conv5b	64	3	1	conv5c	64	3	1
pool1a	max	3	2	pool1b	max	3	2	pool1c	max	3	1
pool2a	max	3	2	pool2b	max	3	2	pool2c	max	3	1

labeled by the final fully connected layer and each label is finally merged together to generate the iris mask. Noticing that smaller input patches (i.e. 56×56) capture more local information while introduce more noisy as shown in Figure 3. Larger input patches (i.e. 224×224) capture more global information and ignore more details. To capture multi-scale context information, the patches around the centered pixel can be extracted at hierarchical scales from small to large. In this way, the model has the ability to capture both the local and global information.

The overall architecture of the proposed HCNNs is shown in Figure 2. It is a binary classification model which takes into hierarchical patches as input to classify iris and non-iris pixel. Three hierarchal layers are fused in a fully connected layer. Each pixel of the input is separately predicted, the results are merged into a mask after every pixel of an input is labeled. Implementation details of HCNNs are shown in Table 1.

However, HCNNs are lack of efficiency and effectiveness for the reasons below: 1) The overlap region of each patch is repeatedly computed in both feedforward and back-propagation. 2) the respective field of the neurons are limited by the size of input patch which is thus suboptimal for capturing more global information.

2.2. Multi-scale fully convolutional networks (MFCNs) for iris segmentation

Fully convolutional networks (FCNs) are consisted of convolutional layers and pooling layers. Different from traditional CNNs, it does not contain fully connected layers. FCNs are essentially special cases of CNNs because fully connected layers are specific forms of convolutional layers with every kernel covers the entire input region. Thus, FCNs could take arbitrary-sized input [24]. The output of the final convolutional layer can be upsampled to the size of the input so that the FCNs is a natural choice for dense segmentation prediction. All the labels of pixels in input iris images can be predicted in the same time. It is the main difference from HCNs where only a label of a pixel can be at each sliding window. In this case, MFCNs input a whole iris image and contain no repeated computation in overlapped regions. The respective fields of neurons are not limited by the size of patches. Thus it avoids two bottlenecks of conventional CNNs and therefore makes iris segmentation faster and more accurate.

The final output heat map is O with dimensions of $m \times n \times c$, where m, n, c are the height, width and number of label classes of input images. Thus we could formulate the loss between final output and the ground truth map.

The loss at each pixel (p, q) is defined as follows:

$$J_{(p,q)}^{(\theta)} = - \sum_{j=1}^K 1\{y_{(p,q)} = j\} \log \frac{\exp(\theta_j^T x_{(p,q)})}{\sum_{l=1}^K \exp(\theta_l^T x_{(p,q)})} \quad (3)$$

$$loss = \sum_{(p,q) \in O} J_{(p,q)}^{(\theta)} \quad (4)$$

where K is the number of classes; θ is the model parameter. $x_{(p,q)}$ represents the response at coordinate (p, q) . $1\{y_{(p,q)} = j\}$ equals to 1 when $y_{(p,q)} = j$ else 0. The soft-max loss is to summarize all the $loss$ at each pixel to measure the overall prediction accuracy.

Overview of MFCNs. The structure of MFCNs is illustrated in Figure 1. It is composed of 31 convolutional layers and 6 pooling layers. Six layers from shallow to deep are fused to formulate a multi-scale network to better capture both the coarse and fine details. Our model is an end-to-end model which requires no pre- and post-processing. It directly predicts the iris and non-iris pixels over the entire input image. Implementation details of the convolutional layers and pooling layers are shown in Tables 2.

Multi-scale within-net fusion for iris segmentation. In FCNs, the outputs are typically reduced by subsampling. The outputs are coarsen and more global, which are useful for capturing global structure or context information. However, it is difficult to predict the fine details of the iris images including the sharp iris boundary and non-iris pixels occluded by hair, eyelash, or reflection. Figure 4 (d) is the output of conv14c, we could easily notice the boundaries of

Table 2. Implementation details of convolutional layers of MFCNs, where ‘layer’ denotes the names of the layers in the overall model in Figure 1; ‘NO.’ is the number of kernels; ‘size’ is the size of convolution or pooling kernels; ‘stride’ is the stride of convolution or pooling; ‘max’ is max pooling and ‘avg’ is average pooling.

layer	NO.	size	stride	layer	NO.	size	stride	layer	NO.	size	stride
conv0	128	3	8	conv6b	128	1	1	conv14a	128	3	1
conv0a	128	1	1	conv6c	2	1	1	conv14b	128	1	1
conv0b	2	1	1	conv7	256	2	1	conv14c	2	1	1
conv1	64	3	1	conv8	256	3	1	conv15	512	3	2
conv2	64	3	1	conv9	256	3	1	conv16	512	3	2
conv3a	128	3	4	conv10a	128	3	1	conv17	512	3	2
conv3b	128	1	1	conv10b	128	3	1	conv20	1024	3	12
conv3c	2	1	1	conv10c	1	1	1	conv21	1024	1	1
conv4	128	3	1	conv11	512	3	1	conv22	2	1	1
conv5	128	3	1	conv12	512	3	1				
conv6a	128	3	2	conv13	512	3	1				
layer	type	size	stride	layer	type	size	stride	layer	type	size	stride
pool3	max	3	2	pool10	max	3	2	pool18	max	3	1
pool6	max	3	2	pool15	max	3	1	pool19	avg	3	1

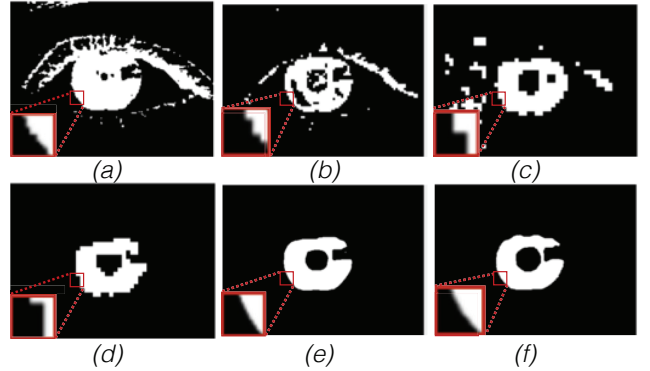


Figure 4. Images (a) to (d) show the outputs of layers conv3c, conv6c, conv10c, conv14c from shallow to deep in MFCNs. Images (e) and (f) are our results and the ground truth, respectively. From the selected square, we could find that the details are from fine to coarse when going deeper. The fusion layer fuses the deep-coarse layer and shallow-fine layer.

iris are highly coarse and the reflection region is badly labeled. In shallow layers, the respective field are much more smaller which enable them to predict fine and sharp texture details. Figure 4 (a) is the output of a shallow layer conv3c. It can predict all the fine boundaries of iris region, reflections and occlusions. However, it introduces too much noise to the segmentation results. In order to capture both the local, texture details and global, structural information, six layers are combined including conv0b, conv3c, conv6c, conv10c, conv14c and conv22 from shallow to deep. Figure 4 (e) are the output of the fusion layer, which is sharp and accurate when compared with the ground truth (Figure 4 (f)). The fine information including iris boundaries, reflec-

tions, occlusions are accurately labeled and the noises are correctly removed. The *Fusion layer* in our model conducts a sum operation to the output of these six layers after they are upsampled to the size of the input.

3. Experiments

3.1. Datasets

UBIRIS.v2 [20] is an iris dataset which is acquired in visible light illumination. It consists of 11102 images from 261 subjects. The images are acquired under unconstrained conditions with subjects at-a-distance and on-the-move. Thus the images often comprise serious occlusions, specular reflections, off-axis and blur. In NICE I competition, a subset of 500 UBIRIS.v2 images was used for training and another disjoint test set of 500 images was used for test. The ground-truth segmentation masks are manually labeled [1]. Our experimental settings are in consistent with the protocol of NICE I competition. But there were 55 test images missing when we downloaded the test images from the organizers of NICE I competition.

CASIA.v4-distance [2] consists of a total of 2567 images from 142 subjects which are acquired using NIR imaging with the subjects 3 meters away from the camera. Note that the iris mask is not available publicly, a total of 400 iris images from the first 40 subjects are manually labeled by ourselves using Photoshop and GrowCut [30]. The first 300 images from the first 30 subjects are employed as the training set. The last 100 images from the last 10 subjects of the dataset are left for person-disjoint test in the experiments. The manually labeled iris regions will be released to the public.

3.2. Results

The implementation of MFCNs is based on caffe [11]. MFCNs are finetuned from a pre-trained VGG-21 model [24] which is trained for natural image classification.

The evaluation measure of the NICE I competition [1] are applied to evaluate the performance of the proposed iris segmentation method. The average segmentation error is denoted as follows:

$$error = \frac{1}{N \times m \times n} \sum_{i,j \in (m,n)} G(i,j) \oplus M(i,j) \quad (5)$$

where N, m, n are the number, length and width of test images, respectively. G and M are respectively the ground truth and the generated iris mask, and i, j are the coordinates of G and M . The \oplus in the equation represents the *XOR* operation to evaluate the disagreeing pixels between G and M .

The proposed MFCNs achieve average segmentation error of 0.90% and 0.59% on UBIRIS.v2 and CASIA.v4-distance databases, respectively. This is the first time to

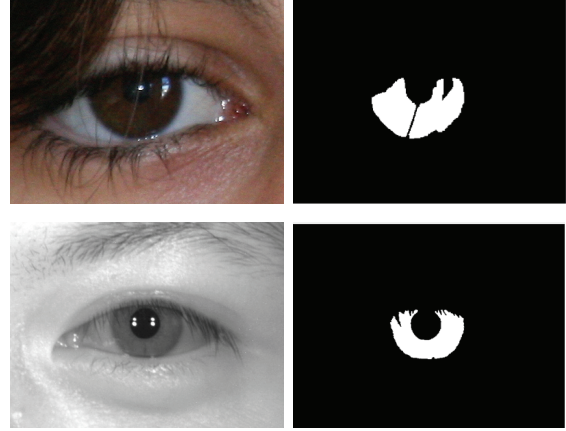


Figure 5. Examples from UBIRIS.v2 and CASIA.v4-distance. The first row is an image from the UBIRIS.v2 dataset and its corresponding ground truth. The second row an image from the CASIA.v4-distance dataset and its corresponding ground truth. It shows that the manually labeled ground truth are highly fine and sharp.

Table 3. Comparisons of average segmentation error to state-of-the-art approaches. Note that ‘CASIA.v4’ in the table is short for ‘CASIA.v4-distance’. The ‘-’ in the table indicates that the compared methods are not experimented in the dataset.

Method	UBIRIS.v2 error(%)	CASIA.v4 error(%)
Ours MFCNs	0.90	0.59
Ours HCNNs	1.11	1.08
Z. Zhao and A. Kumar, ICCV, 2015 [33]	1.21	0.68
T. Tan <i>et al.</i> , IVC, 2009 [28]	1.31	-
C. Tan and A. Kumar, T-IP, 2013 [27]	1.72	0.81
H. Proença, T-PAMI, 2010 [19]	1.87	-
C. Tan and A. Kumar, T-IP, 2012 [26]	1.90	1.13

achieve an average error which is lower than 1% on both datasets. Table 3 shows the summary of the performance of other state-of-the-art methods in recent publications. Even though the results cannot be directly compared because the training and test data of other methods and ours are different, it is still meaningful to show their differences.

Our experimental settings are almost the same as NICE I competition except 55 missing test images which should not change the statistic significantly, so it is relatively fair to compare our results to those in [19, 28]. The improvements of MFCNs over the latest publication [33] is respectively 25.62% and 13.24% on two datasets. Note that MFCNs have numerous parameters to be learned and use more training data than [26, 27, 33].

It demonstrates the effectiveness of MFCNs on both datasets acquired from visible light and NIR illumination. MFCNs are an end-to-end model which requires no

further pre- and post-processing. Most of the methods [19, 26, 27, 28, 33] employ pre-processing to remove illumination effects and noise. Compared to other methods, our model is robust to the illumination variations and noises. Figure 6 shows some segmentation results produced by MFCNs. Figures 8 and 9 show the false accept and false reject points that MFCNs generate. They demonstrate several successful iris segmentation examples in the presence of specular reflections, off-axis and occlusions in non-cooperative applications.

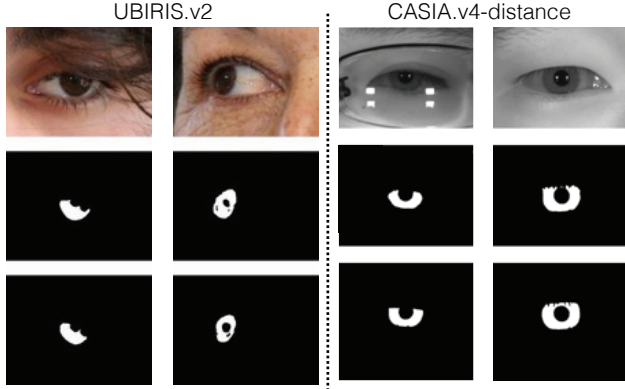


Figure 6. Segmentation results by MFCNs. The first, second and third rows are the input images, our results and the ground truth, respectively.

Experiments are also conducted on both datasets to test HCNNs which does a pairwise prediction. It is deserved to notice that the performance of HCNNs outperforms other state-of-the-art methods on UBIRIS.v2 dataset as shown in Table 3. MFCNs outperform HCNNs significantly. Remind that the performance of HCNNs is inferior on the CASIA.v4-distance dataset compared with that on the UBIRIS.v2 dataset. The reason is that we only test equally spaced 100×100 pixels of input images due to unacceptable computational cost. The image size of CASIA.v4-distance (640×480) is larger than UBIRIS.v2 (300×400). So the upsampling process on the CASIA.v4-distance dataset degrades the accuracy more seriously.

MFCNs have much higher efficiency compared with HCNNs. For a $300 \times 400 \times 3$ input image, the test time of HCNNs is 30 seconds on a GTX Titan GPU. However, MFCNs take merely 0.2 second which is 150 times faster than HCNNs.

3.3. Challenging examples

Figure 7 shows some challenging examples obtained by MFCNs from UBIRIS.v2 and CASIA.v4-distance dataset. By carefully checking all the segmentation results from the UBIRIS.v2, we find out that most failure or suboptimal cases are coming from the ‘black skin’ subjects and those images which contain no eyes. This situation may be

caused by a lack of training examples. Because there are only 10 images which are coming from ‘black skin’ subjects in the training set. Note that all images in our training sets contain an eye. We believe that if given more training data, the results of these hard examples will be refined. Other suboptimal examples from UBIRIS.v2 are mislocalized and isolated small regions. Meanwhile, most of the hard examples from CASIA.v4-distance are also isolated small regions. Trivial post-processing approaches can remove these small regions. To further improve the accuracy, more sophisticated post-processing methods can ensure both smooth boundaries and consist labeling.

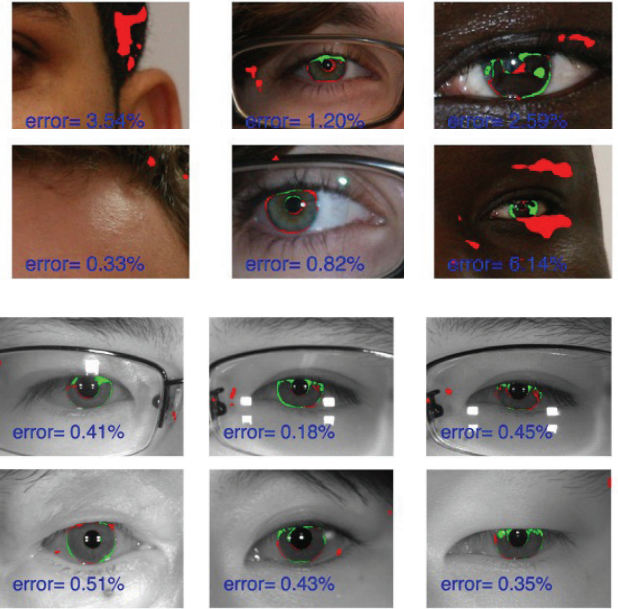


Figure 7. Hard examples from UBIRIS.v2 and CASIA.v4-distance dataset where the green points denote false accept points (i.e. points labeled as non-iris by the ground truth but iris by our method), the red points denote false reject points (i.e. points labeled as iris by the ground truth but non-iris by our method). The first two rows are results from UBIRIS.v2 and the last two rows are results from CASIA.v4-distance dataset. This image is better viewed in color.

4. Conclusions

Robust and accurate iris segmentation is a key procedure to ensure high iris recognition rate. In this paper, we have developed a much more accurate and efficient iris segmentation framework, namely MFCNs, to segment the iris images under non-cooperative environment in an end-to-end manner. MFCNs learn features and classifiers simultaneously from data and rely on no handcrafted features and sophisticated rules. In addition, no complex pre- and post-process are required during segmentation. Experimental results on

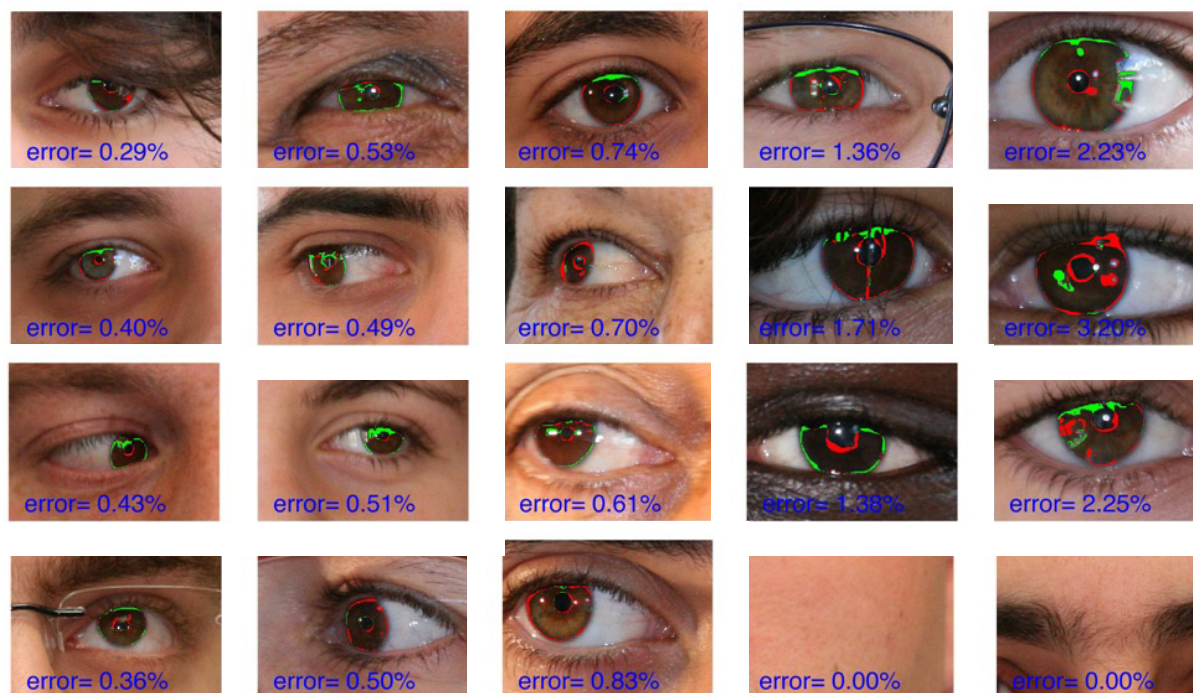


Figure 8. Segmentation results from UBIRIS.v2 dataset by MFCNs where the green points denote false accept points (i.e. points labeled as non-iris by the ground truth but iris by our method), the red points denote false reject points (i.e. points labeled as iris by the ground truth but non-iris by our method). This image is better viewed in color.

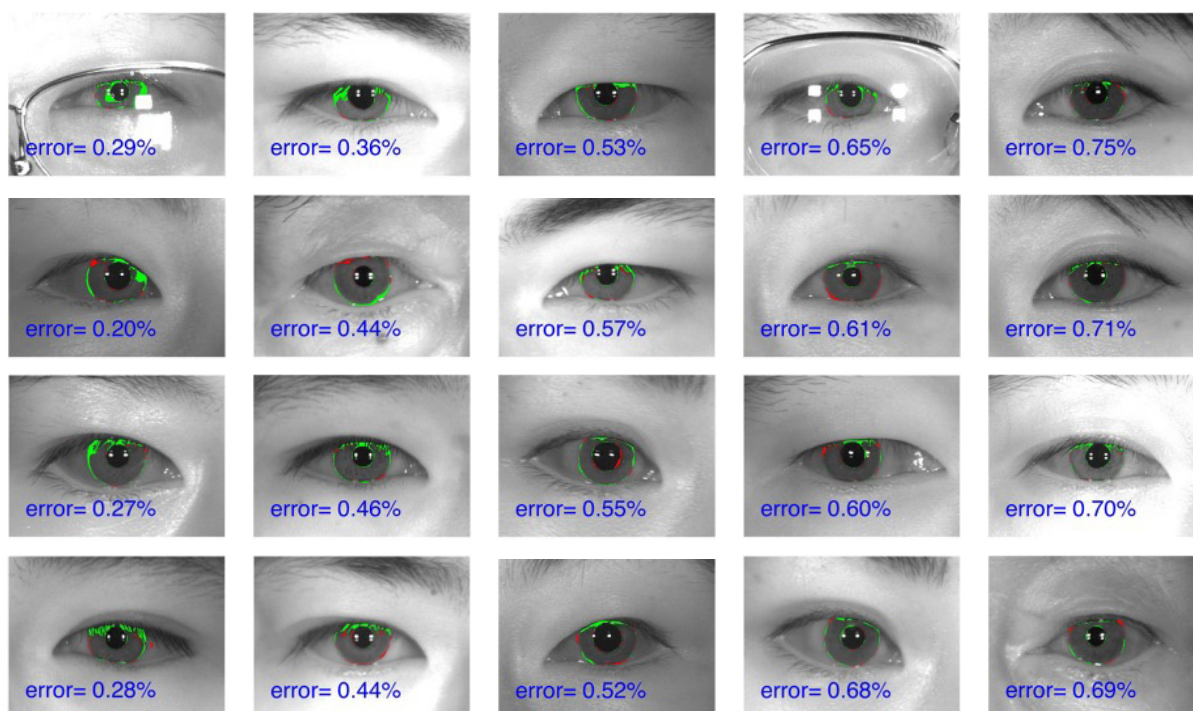


Figure 9. Segmentation results from CASIA.v4-distance dataset by MFCNs where the green points denote false accept points (i.e. points labeled as non-iris by the ground truth but iris by our method), the red points denote false reject points (i.e. points labeled as iris by the ground truth but non-iris by our method). This image is better viewed in color.

the challenging UBIRIS.v2 and CASIA.v4-distance iris image datasets have demonstrated that MFCNs achieve state-of-the-art iris segmentation accuracy.

5. Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 61403389, 61273272, 61573360) and the Beijing Talents Fund (Grant No. 2015000021223ZK30). The authors also acknowledge the SOCIA Laboratory of the University of Beira Interior for the training and test iris images of NICE I competition.

References

- [1] (2009). NICE.I-Noisy Iris Challenge Evaluation, Part I. <http://nice1.di.ubi.pt/index.html>.
- [2] Biometrics Ideal Test, CASIA.v4 database: <http://www.idealtest.org/dbDetailForUser.do?id=4>.
- [3] T. Camus, R. Wildes, et al. Reliable and fast eye finding in close-up images. In *ICPR*, 2002.
- [4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv*, 2014.
- [5] J. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE TPAMI*, 15(11):1148–1161, 1993.
- [6] J. Daugman. How iris recognition works. *IEEE TCSVT*, 14(1):21–30, 2004.
- [7] J. Daugman. New methods in iris recognition. *IEEE TSMC, Part B*, 37(5):1167–1175, 2007.
- [8] C. Fancourt, L. Bogoni, K. Hanna, Y. Guo, R. Wildes, N. Takahashi, and U. Jain. Iris recognition at a distance. In *Audio- and Video-Based Biometric Person Authentication*, 2005.
- [9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning Hierarchical Features for Scene Labeling. *IEEE TPAMI*, 35(8):1915–1929, 2013.
- [10] Z. He, T. Tan, Z. Sun, and X. Qiu. Toward Accurate and Fast Iris Segmentation for Iris Biometrics. *IEEE TPAMI*, 31(9):1670–1684, 2009.
- [11] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*, 2012.
- [13] H. Li, Z. Sun, and T. Tan. Robust iris segmentation based on learned boundary detectors. In *ICB*, 2012.
- [14] Y.-h. Li and M. Savvides. An Automatic Iris Occlusion Estimation Method Based on High-Dimensional Density Estimation. *IEEE TPAMI*, 35(4):784–796, Feb. 2013.
- [15] N. Liu, M. Zhang, H. Li, Z. Sun, and T. Tan. Deepiris: Learning pairwise filter bank for heterogeneous iris verification. *PRL*, 2015.
- [16] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.
- [17] J. Matey, O. Naroditsky, K. Hanna, R. Kolczynski, D. LoIacono, S. Mangru, M. Tinker, T. Zappia, and W.-Y. Zhao. Iris on the move: Acquisition of images for iris recognition in less constrained environments. *Proc. of the IEEE*, 94(11):1936–1947, 2006.
- [18] D. Menotti, G. Chiachia, A. Pinto, W. Robson Schwartz, H. Pedrini, A. Xavier Falcao, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *IEEE TIFS*, 10(4):864–879, 2015.
- [19] H. Proença. Iris Recognition: On the Segmentation of Degraded Images Acquired in the Visible Wavelength. *IEEE TPAMI*, 32(8):1502–1516, 2010.
- [20] H. Proença, S. Filipe, R. Santos, J. Oliveira, and L. A. Alexandre. The UBIRIS.v2: A Database of Visible Wavelength Iris Images Captured On-the-Move and At-a-Distance. *IEEE TPAMI*, 32(8):1529–1535, 2010.
- [21] S. Pundlik, D. Woodard, and S. Birchfield. Non-ideal iris segmentation using graph cuts. In *CVPR Workshop*, 2008.
- [22] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.
- [23] S. Shah and A. Ross. Iris segmentation using geodesic active contours. *IEEE TIFS*, 4(4):824–836, 2009.
- [24] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv*, 2014.
- [25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.
- [26] C.-W. Tan and A. Kumar. Unified Framework for Automated Iris Segmentation Using Distantly Acquired Face Images. *IEEE TIP*, 21(9):4068–4079, 2012.
- [27] C.-W. Tan and A. Kumar. Towards Online Iris and Periocular Recognition Under Relaxed Imaging Constraints. *IEEE TIP*, 22(10):3751–3765, Aug. 2013.
- [28] T. Tan, Z. He, and Z. Sun. Efficient and Robust Segmentation of Noisy Iris Images for Non-cooperative Iris Recognition. *IVC*, 28(2):223–230, Feb. 2010.
- [29] A. Uhl and P. Wild. Weighted adaptive hough and ellipsoidal transforms for real-time iris segmentation. In *ICB*, 2012.
- [30] V. Vezhnevets and V. Konouchine. Growcut: Interactive multi-label nd image segmentation by cellular automata. In *proc. of Graphicon*, pages 150–156, 2005.
- [31] R. Wildes. Iris recognition: an emerging biometric technology. *Proc. of the IEEE*, 85(9):1348–1363, 1997.
- [32] Z. Wu, Y. Huang, Y. Yu, L. Wang, and T. Tan. Early Hierarchical Contexts Learned by Convolutional Networks for Image Segmentation. In *ICPR*, 2014.
- [33] Z. Zhao and A. Kumar. An accurate iris segmentation framework under relaxed imaging constraints using total variation model. In *ICCV*, 2015.