Detecting visible areas of iris by classifier of textures with support set

Solomatin Ivan,
Matveev Ivan.

Moscow Institute of Physics and Technology
Federal Research Centre "Computing Centre" of
Russian Academy of Sciences

October 11, 2016

Purpose of the study

Purpose:

To build automatic algorithm localizing visible areas of iris using classifier, which is trained on the pixels of unoccluded sector on the processed image.
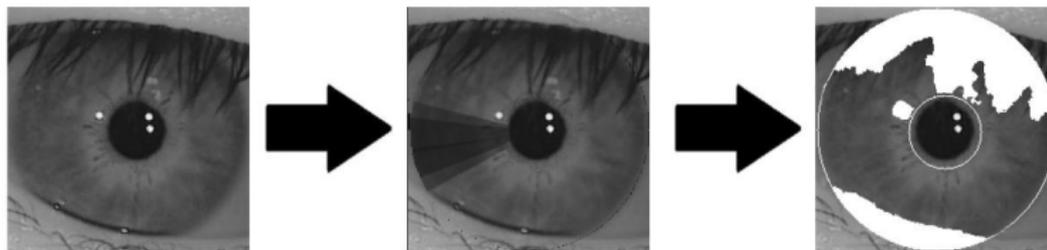
Problem statement

Input:

- I - grayscale bitmap sized $W \times H$. Every pixel is encoded in one byte.
- x_P, y_P, r_P - coordinates of the center and radius of the circle that approximates the pupil-iris boundary.
- x_I, y_I, r_I - coordinates of the center and radius of the circle that approximates the sclera-iris boundary.

Output:

Binary matrix J, sized $W \times H$. Every pixel of the matrix shows if the corresponding pixel of the source image contains occlusion.

$$J \in \mathbb{B}^{[1;W] \times [1;H]}, \text{ where } \mathbb{B} = \{0, 1\}$$

Problem statement

In the formal way:

$$\Omega \subset [1; W] \times [1; H];$$

$$\Omega = \left\{ (x, y) : \left| \begin{array}{l} (x - x_P)^2 + (y - y_P)^2 \geq r_P^2 \\ (x - x_I)^2 + (y - y_I)^2 \leq r_I^2 \end{array} \right. \right\}.$$

Ω is an annular region of the iris localization.
The purpose is to classify all the pixels of Ω into two classes, it means to build a classifier:

$$Q(x, y) : \Omega \to \{0, 1\}. \qquad Q(x, y) = \begin{cases} 1, & \text{occlusion} \\ 0, & \text{iris} \end{cases}$$

The results of the classification are compiled into the binary matrix $J \in \mathbb{B}^{[1; W] \times [1; H]}$, where $\mathbb{B} = \{0, 1\}$.

Related works

1. J. Daugman, *How iris recognition works*, ICIP (1). 2002. pp. 33-36.
   In this work, boundaries of eyelids are detected using **integro-differential operators**.

2. J. Daugman, *New methods in iris recognition*, Systems, Man, and Cybernetics, Part B, IEEE Transactions on, vol. 37, no. 5, pp. 1167–1175, Oct. 2007.
   In this work, **active contour** approach is used to detect eyelids boundaries.

Related works

6 / 27

Detecting visible areas of iris

I. A. Solomatin, I. A. Matveev

Purpose of the study

Problem statement

Related works

Proposed method
1. Polar transformation
2. Feature vector.
3. Basic set
4. Training the classifier
5. Classification
6. Morphological post-processing
7. Lacunas detection

Examples

Experiment
Step 1
Step 2

Conclusion

3. D. Zhang, D.M. Monro, and S. Rakshit, *Eyelash removal method for human iris recognition*, Image Processing, 2006 IEEE International Conference on, pp. 285–288, Oct. 2006. In this work **Sobel filter** is used for eyelashes detection. After detection they are removed from the image using median filter.

4. Yung hui Li and Marios Savvides. *A pixel-wise, learning-based approach for occlusion estimation of iris images in polar domain.* In ICASSP, pages 1357-1360. IEEE, 2009. In this work algorthm, localizing occlusions is implemented using **classifier based on Gaussian Mixtures**, which is trained on **human-made training set**.

Proposed method. Basic stages

1. Applying the polar transformation.

2. Feature vector calculation.

3. Finding the basic set S'.

4. Training the classifier, using S' as training set.

5. Classification.

6. Morphological post-processing.

7. Lacunas detection.

8. Applying the reverse polar transformation.
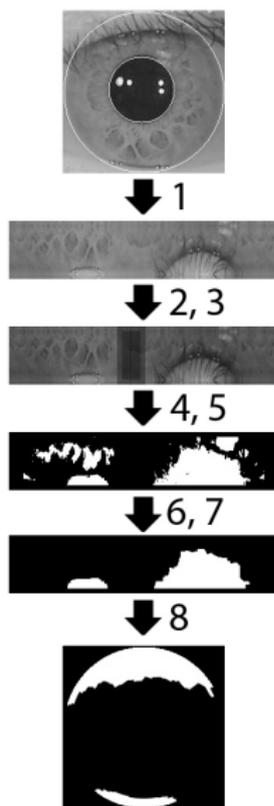
Stage 1. Polar transformation

Coordinates of the pixels of the new region are set in (ρ, φ) variables: $\rho \in [1; h], \varphi \in [1; w]$. Every point of the polar domain has a prototype in the Cartesian domain according to the following rule:

$$
\begin{cases}
\hat{x}(\rho, \varphi) = x_P + \left(R_P + \frac{\rho_0(\frac{2\pi\varphi}{w})\rho}{h} \right) \cos\left(\frac{2\pi\varphi}{w} \right) \\
\hat{y}(\rho, \varphi) = y_P + \left(R_P + \frac{\rho_0(\frac{2\pi\varphi}{w})\rho}{h} \right) \sin\left(\frac{2\pi\varphi}{w} \right)
\end{cases}
$$

The Ω in the polar domain becomes a rectangle
$\Omega' = [1; w] \times [1; h]$
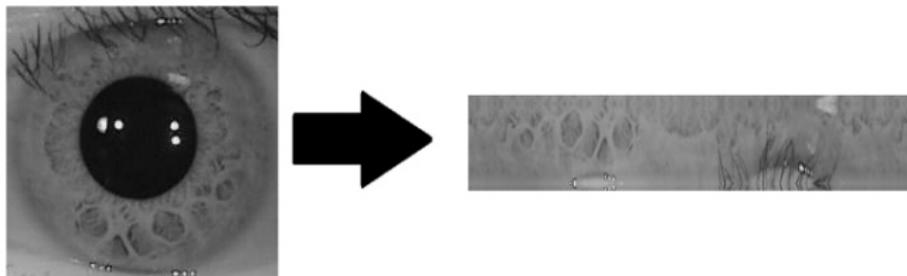
Stage 2. Calculating the feature vector.

The feature vector consists of the following $K = 12$ components:

- $B(\vec{x})$ — brightness in the point $\vec{x} = (\varphi, \rho)^{\mathrm{T}}$
- $\overline{B}(\vec{x})$ — average brightness in the neighbourhood of the point $\vec{x}$.
- $\sigma(\vec{x})$ — the standard deviation of brightness in the neighbourhood of $\vec{x}$.
- $\vec{C}(\vec{x})$ — vector of five components of discrete cosine transform of neighbourhood of $\vec{x}$. DCT is calculating in the neighbourhood sized 8×8.
- $\vec{M}(\vec{x})$ — vector of four components of co-occurrence matrix of neighbourhood of $\vec{x}$, which is binarized by Otsu's threshold.

Every point $\vec{x} \in \Omega'$ has corresponding feature vector:

$$\vec{p}(\vec{x}) = (B(\vec{x}), \overline{B}(\vec{x}), \sigma(\vec{x}), (\vec{C}(\vec{x}))^{\mathrm{T}}, (\vec{M}(\vec{x}))^{\mathrm{T}})^{\mathrm{T}}$$

Stage 3. Basic set

Then method finds unoccluded sector S, which is supposed to be the sector with minimum kurtosis coefficient:

$$\mu = \mathbb{E}\left[(X - \mathbb{E}X)^4\right].$$

Fixing the central angle of the sector and solving the problem in Ω', it is possible to find sector with minimum kurtosis coefficient spending $O(wh)$ time. Formal problem statement for sector with central angle $\Delta\alpha'$:

$$\alpha' = \operatorname*{argmin}_{\alpha' \in [0; \frac{w}{2}]} \sum_{x=\alpha'-\Delta\alpha'}^{\alpha'+\Delta\alpha'} \sum_{y=1}^{h} \left(I'(x,y) - \frac{1}{\Delta\alpha' h} \sum_{i=\alpha'-\Delta\alpha'}^{\alpha'+\Delta\alpha'} \sum_{j=1}^{h} I'(i,j) \right)^4.$$

Stage 3. Basic set

This figure illustrates kurtosis coefficient distribution on one of the images from CASIA database. The dark rectangle indicates the chosen sector.

Stage 3. Basic set

During the computation experiments two different sectors were used - sector with minimum variance of brightness and with mininum kurtosis coefficient of brightness. An experiment was conducted using images from ICE database with human-made masks of occlusion. Vectors of the errors were calculated for both of the methods (E_1 - variance, E_2 - kurtosis coefficient) and then analysed using statistical tests of Pearson and Wilcox. With alternate hypothesis $ME_1 > ME_2$:

$$\text{p-value}_{Pearson} = 0.038,$$

$$\text{p-value}_{Wilcox} = 0.0016.$$

So, the zero hypothesis should be rejected at level of significance $\alpha = 0.05$. So, it gives a reason to suppose that using kurtosis koefficient makes finding of S more accurate.

Stage 4. Training the classifier

To train classifier Q, the points of S are used: $S = \{\vec{x}_n\}_{n=1}^N$.

Local textural features of these points will be the training set for the classifier.

$$\vec{\mu} = \frac{1}{N} \sum_{n=1}^N \vec{p}(\vec{x}_n).$$

$$\vec{\bar{p}}(\vec{x}_n) = \vec{p}(\vec{x}_n) - \vec{\mu}.$$

$$M = \left(\vec{\bar{p}}(\vec{x_1}) \quad \vec{\bar{p}}(\vec{x_2}) \quad \cdots \quad \vec{\bar{p}}(\vec{x_n}) \right)$$

$$C = MM^{\mathrm{T}}.$$

Stage 5. Classification

- Mahalanobis distance between $\vec{x}$ and middle vector $\vec{\mu} = \frac{1}{|S|} \sum_{\vec{x_S} \in S} \vec{p}(\vec{x_S})$:

$$D(\vec{x}) = \sqrt{(\vec{p}(\vec{x}) - \vec{\mu})^{\mathrm{T}} C^{-1} (\vec{p}(\vec{x}) - \vec{\mu})}.$$

- Pixel $\vec{x}$ is supposed to be an element of "iris" class, if:

$$P(\vec{x}) = \exp\left(-D^2(\vec{x})\right) \geq P_{\text{threshold}}.$$

$$Q(\vec{x}) = \begin{bmatrix} 0, \text{ if } P(\vec{x}) \geq P_{\text{threshold}} \Longrightarrow \vec{x} \text{ --- iris} \\ 1, \text{ if } P(\vec{x}) < P_{\text{threshold}} \Longrightarrow \vec{x} \text{ --- occlusion} \end{bmatrix}$$

- The result of the classification is matrix J':

$$J'(\varphi, \rho) = Q(\vec{x})$$

Stage 6. Morphological post-processing

Morphological post-processing operator

$$M(J') : \mathbb{B}^{\Omega'} \to \mathbb{B}^{\Omega'}$$

is defined as:

$$M(J') = (J' \bullet B) \circ B,$$

where:
- $\bullet$ — morphological closing,
- $\circ$ — morphological opening,
- B — structure element (square 3×3).

Stage 6. Morphological post-processing

(a) $M(J')$

(b) $L(M(J'))$
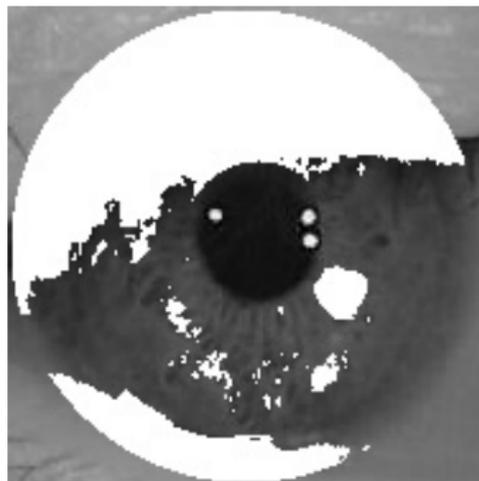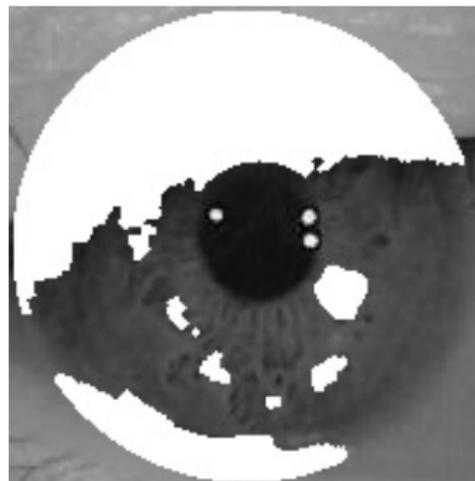
(c) $I \odot f_{reverse}(M(J'))$

(d) $I \odot f_{reverse}(L(M(J')))$

Stage 7. Lacunas detection

Morphological post-processing doesn't remove all the noise completely. Some irises have complex structure, e.g. lacunas. Lacunas detection operation

$$L(J') : \mathbb{B}^{\Omega'} \to \mathbb{B}^{\Omega'}$$

divides occlusion into connected components and removes the components which have average brightness lower than the unoccluded area. The exception are the components which are on the edge of the iris — they are not changed.

Stage 7. Lacunas detection

(e) $M(J')$

(f) $L(M(J'))$
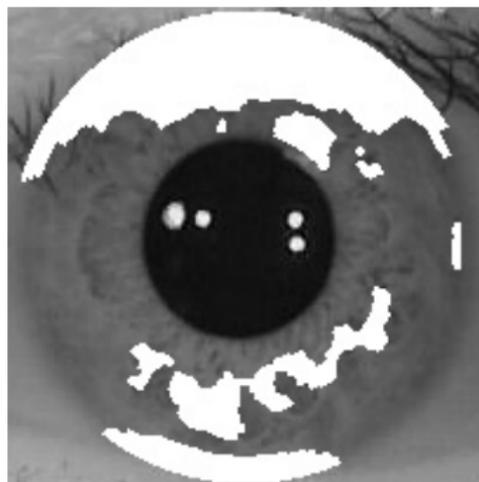
(g) $I \odot f_{reverse}(M(J'))$
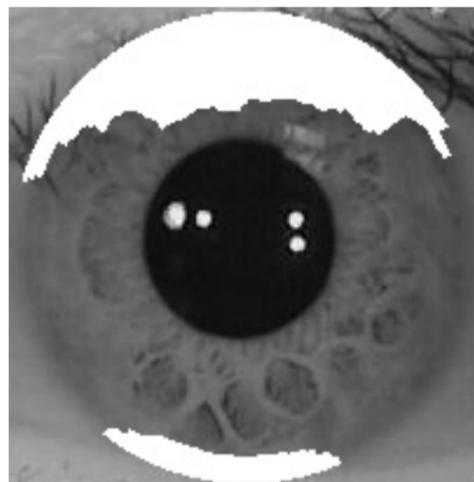
(h) $I \odot f_{reverse}(L(M(J')))$

Example 1

Source image:

Result:

Example 2

Source image:

Result:

Example 3

Source image: Result:

Experiment

An experiment was conducted using iris images from CASIA and ICE databases and consisted of **two steps**:

1. During the first step the output mask was **compared to the expert mask**. As the error function the sum of the relative errors of the first and second type was used.

2. During the second step, the output mask was **used to identify human** by iris using Libor Masek algorithm.

Experiment. Step 1

The experiment was conducted using 2100 images of the CASIA database and 3000 images of the ICE database. For the error function calculation a human-made expert mask was used.

The accuracy during the experiment:

$$CASIA: \quad \overline{E} = 0.196$$
$$ICE\ DB: \quad \overline{E} = 0.321$$

Experiment. Step 1

This figure illustrates histogram of errors on 2100 images from CASIA database and on 3000 images from ICE database.
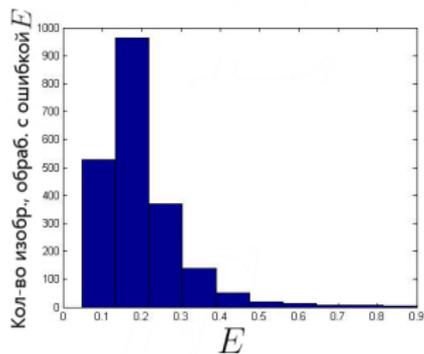
CASIA
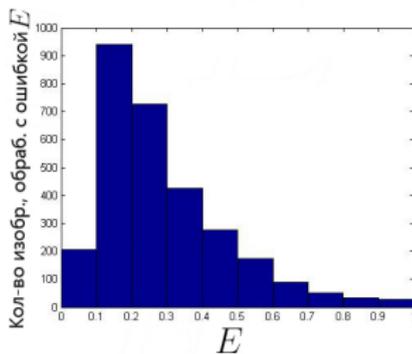
ICE DB

Experiment. Step 2

The experiment was conducted on 1000 images from CASIA and 1500 images from ICE. This figure contains the DET curves:

25 / 27

Detecting visible areas of iris

I. A. Solomatin,
I. A. Matveev

Purpose of the study

Problem statement

Related works

Proposed method
1. Polar transformation
2. Feature vector.
3. Basic set
4. Training the classifier
5. Classification
6. Morphological post-processing
7. Lacunas detection

Examples

Experiment
Step 1
Step 2

Conclusion
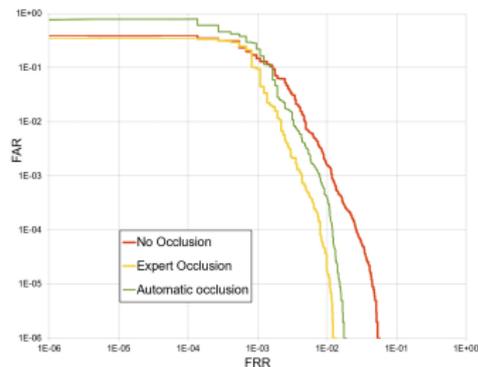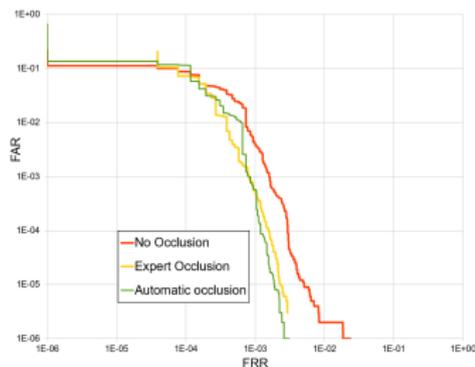
(i) CASIA

(j) ICE

This table shows *EER* for recognition without mask, with algorithm mask, and with expert mask:

Image base	EER_{nomask}	EER_{auto}	EER_{expert}
CASIA D	0.0061	0.0044	0.0030
ICE D	0.0014	0.0008	0.0008

Conclusion

The proposed method detects iris occlusions with high accuracy.

1. Main advantage of the method is that it is **automatic** — it doesn't need human made training sets.

2. Method significantly **increases the accuracy of human recognition**, but still worse than a human-made occlusion mask.

3. Method builds **new classifier for every image**. It provides robustness towards different iris structure, different illumination and so on.

4. Average processing time using AMD Opteron(TM) Processor 6272 with frequency 2100 Mhz is **0.7 seconds**

Thank you for your attention!