A Geometric Approach to Face Detector Combining*

Nikolay Degtyarev and Oleg Seredin

Tula State University
http://lda.tsu.tula.ru
n.a.degtyarev@gmail.com

Abstract. In this paper, a method of combining face detectors is proposed, which is based on the geometry of the competing face detection results. The main idea of the method consists in finding groups of similar face detection results obtained by several algorithms and further averaging them. The combination result essentially depends on the number of algorithms that have fallen in each of the groups. The experimental evaluation of the method is based on seven algorithms: Viola-Jones (OpenCV 1.0), Luxand© FaceSDK, Face Detection Library, SIFinder, Algorithm of the University of Surrey, FaceOnIt, Neurotechnology© VeriLook. The paper contains practical results of their combination and a discussion of future improvements.

Key words: combining classifiers, face detection, clustering of detector outputs, combination of face detectors, comparative test

1 Introduction

The state-of-the-art algorithms of face detection (FD) have excellent performance for many tasks [1–3, 8, 13]. However, even the best of them still have significant error rates, e.g., $5-6\%$ False Rejection Rate, separating them from the desired error-free result. At the same time, it was shown by Degtyarev et al. [3] that the percentage of challenging images incorrectly processed by all tested algorithms is much smaller, only 0.13%, whereas each of the remaining 99.87% of images is correctly processed by at least one of the algorithms.

This fact reveals the possibility of reducing the error rate of face detection through harnessing several diverse algorithms in parallel. We call such principle the *combination* or *fusion of face detectors* on the analogy of the commonly adopted term of *combination/fusion of classifiers*, introduced by J. Kittler [11] and R. Duin [5].

In classifier combining, an object submitted to analysis is supposed to be indivisible, and the final output is result of jointly processing a number of elementary decisions on its class membership – voting, optimal weighting, etc. However, as

* This work is supported by the Russian Foundation for Basic Research, Grant No. 09-07-00394.

to face detector combining, images under processing are not atomic, and what is to be fused is an ensemble of diverse suggestions on the position of the face in the given image in addition to the information of its presence.

In this paper, we propose a quite naive and obvious *geometric approach* to fusing several face detectors, which takes into account only geometric properties of their outputs and ignores, for the computational simplicity sake, other individual properties like False Rejection/Acceptance Rates (FRR/FAR), Confidence Rates, etc. The main principle of combining consists in clustering of the detected face represented by the centers of the eyesand further averaging the cluster centers with respect to the portions of the detectors that have fallen in each of the clusters.

The most tangible disadvantage of the very idea of combining several diverse face detectors is the increasing computational time. However, the recent advances in the multi-core CPU technology (Central Processing Unit) allows, in principle, for a natural parallelization by the scheme "one detector – one core".

2 Models of Face Representation and Localization Accuracy

To combine or correctly compare face detectors, they should represent faces in a unified form. Coordinates of the eye centers (i.e., centers of the pupils) are the most suitable description of faces for these tasks. The reasons for this proposition are, first, the convenience of this kind of representation from the viewpoint of comparing the results, second, the necessity of matching the eye centers as an inevitable step in the majority of learning algorithms, and, third, the fact that ground-truthing eyes by a human is faster, easier and can be done more confidently than locating faces by rectangles.

Thus, we consider *all faces* as represented by their eye centers. If some FD returns a face location in the rectangular form, we first additionally estimate the coordinates of the eye centers by the eye reconstruction algorithm proposed in [3] and examined in [4].

If a detected face is represented by the centers of the eyes (Fig. 1.a), we consider them as correctly detected, if and only if the detected eyes belong to the pair of circles D^A around the true locations of the eyes. The common diameter of the circles $D_{Eyes} = 2\alpha \times l_{Eyes}$ depends on the distance l_{Eyes} between the centers of the eyes with coefficient α taken equal to 0.25. This criterion was originally used by Jesorsky et al. [7]).

3 A Geometric Method of Face Detectors Combining

As mentioned above, the proposed method of FD combining is based on the *geometric approach*. This means that the method takes into account only the positions of detected faces and disregards any additional information related to or provided by FD e.g. false rejection or acceptance error rate, confidence rate of detection, etc. Before describing the method, let us introduce some definitions.

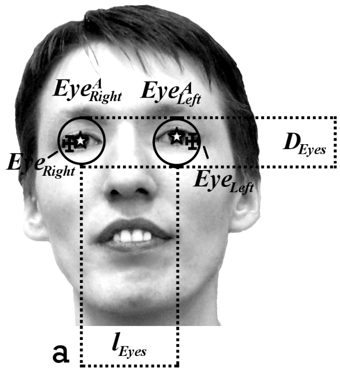

Fig. 1. Schematic face representation. Eye_{Left} and Eye_{Right} – absolute coordinates of detected left and right eyes respectively; l_{Eyes} – distance between eye centers; D_{Eyes} – diameter of the area of acceptable eyes' coordinates deviation from the true eyes location Eye^A_{Right} and Eye^A_{Left};

Definition 1. *The distance between the faces of a given pair (g, h) each represented by the centers of the eyes is the greatest of the Euclidean distances between the left eyes of the pair and the right ones:*

$$d_{Faces}(g,h) = \max\Big(\big\|Eye^g_{Left} - Eye^h_{Left}\big\|, \big\|Eye^g_{Right} - Eye^h_{Right}\big\|\Big). \tag{1}$$

Hereafter, Eye^g_{Right} and Eye^g_{Left} stand for the coordinates of, respectively, the left and the right eye of the given face g.

Definition 2. *The merged face is a synthetic pair of the coordinates of eye centers, averaged among the given group of K faces:*

$$\begin{gathered} Eye^{Merge}_{Left} = \frac{1}{K}\sum_{i=1}^{K} Eye^i_{Left}, \quad Eye^{Merge}_{Right} = \frac{1}{K}\sum_{i=1}^{K} Eye^i_{Right}, \\ l^{Merge}_{Eyes} = \frac{1}{K}\Big\|\sum_{i=1}^{K} Eye^i_{Left} - \sum_{i=1}^{K} Eye^i_{Right}\Big\|. \end{gathered} \tag{2}$$

The model of eyes localization accuracy described in Section 2 implies that if each algorithm of a group of algorithms has correctly detected a face, then the distances between the detected faces are smaller or equal to the diameter D_{Eyes} of the respective area D^A, i.e.:

$$d_{Faces}(g,h) \leq D_{Eyes} \leq 2\alpha \times l_{Eyes}. \tag{3}$$

A merged face based on a group of accurate algorithms may be treated as a correctly detected face, too. If some algorithms in the group have incorrectly detected a face, the merged face based on all results of the group may still be a correctly detected face, depending on the number of incorrectly estimated positions and the errors of estimates.

Definition 3. *A given pair of faces (g, h) is mergeable if and only if the distance between them is at least* 2α *times smaller than the interocular distance of the corresponding merged face* $l_{Eyes}^{Merge}(g, h)$*, i.e.* $d_{Faces}(g, h) \leq 2\alpha\big(l_{Eyes}^{Merge}(g, h)\big)$.

In other words, the pair of *mergeable* faces will be correctly detected, if position of the corresponding merged face on the image is the true face location. Examples of *mergeable* and *nonmergeable* faces are given in Fig. 2.

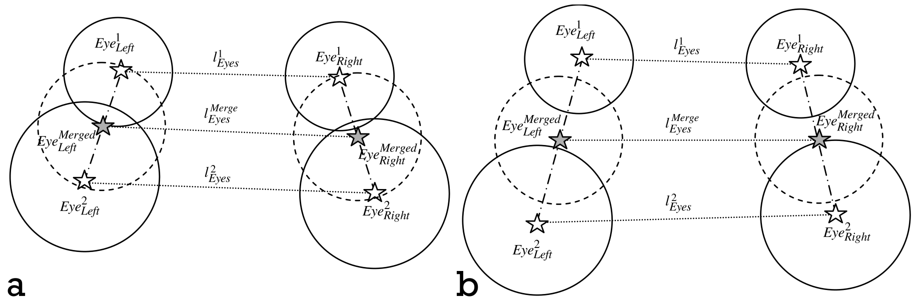

Fig. 2. Examples of mergeable (a) and non mergeable (b) pairs of faces.

In practice, we don't know whether an algorithm has detected a face correctly or incorrectly, nor the true location of the face in the image, and even nor whether a face is in the image at all. Therefore, we suppose that correctly detected faces form clusters around the "true face location", whereas incorrectly detected faces must be scattered in the image. Such clusters may be defined as follows.

Definition 4. *A group of faces forms a cluster if and only if there is at least one face (further called the "center") among the group, that is meargeable with all other faces (in the group).*

Such definition of a cluster of faces is less strict than (3), and allows for intersecting clusters of detected faces, i.e., one face can be member of several different clusters. However, the merged face based on the largest cluster is more likely to be result of correct detection than one based on all other clusters. This consideration is the essence of Algorithm 1. The algorithm consists in repeatedly replacing all faces in the largest cluster by corresponding merged faces, until there exists at least one non-trivial cluster (i.e. with size greater than 1). When at some step the remaining clusters become trivial, namely, each of them contains one face, the algorithm selects the merged face produced by the greatest number of originally detected faces, but not less than the preset *Threshold*, which is the only parameter of the algorithm, otherwise, the algorithm makes the decision that the image contains no face.

It should be emphasized that after replacing clusters by corresponding merged faces, each merged face can become a part of other clusters, etc. A simulated example of combining Face Detectors by this algorithm is shown in Fig. 3. The proposed method is discussed in Section 6.

Algorithm 1 A Geometrical Method of Face Detectors Combining.

Require: α, Threshold;
DetFaces = $\{(Eye_{Left}, Eye_{Right}, merge_{count} = 1), \ldots\}$;
Ensure: $(Eye_{Left}, Eye_{Right})$
  loop
    Determinate the largest cluster of the faces (see Definition 4);
    if the size of the found cluster is > 1 **then**
      Replace all faces in the cluster by the corresponding merged face in DetFaces
    else
      Since there are no clusters consisting of more than one face, select merged face ($Face_m$), that has been originated by the greatest, but not least than the Threshold number of initial faces.
      if such merged face is founded **then**
        return the merged face ($Face_m$);
      else
        return NOT_FACE;
      end if
    end if
  end loop

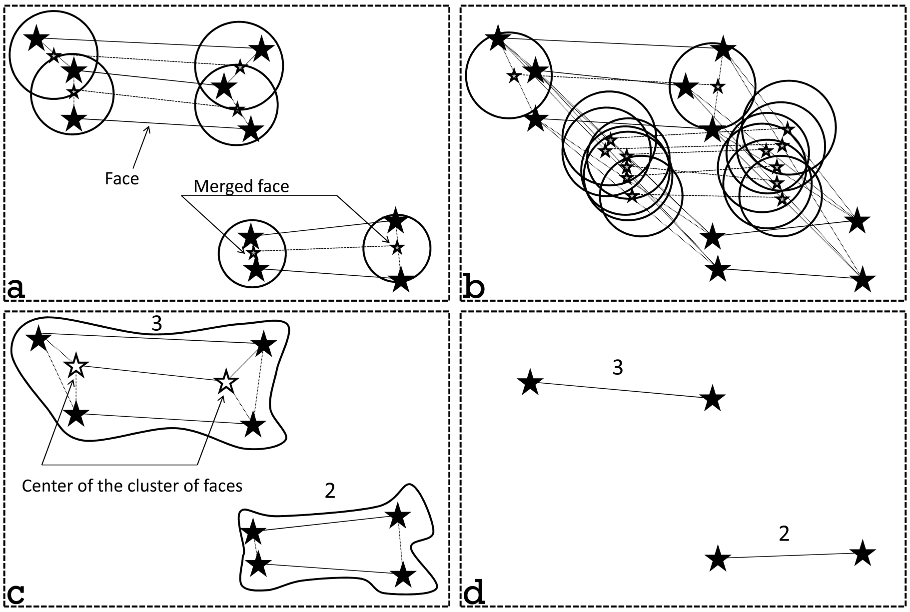

Fig. 3. Successive steps of combining simulated outputs of five Face Detectors by Algorithm 1: mergeable (a) and nonmergeable (b) pairs of faces; clusters of faces (c); substitution of clusters by the corresponding merged faces (d).

4 Experimental Procedure

In this work, we combined the following implementations of different algorithms: Viola-Jones [14] (OpenCV 1.0, OCV); Luxand FaceSDK (FSDK, `http://www.luxand.com`); Face Detection Lib. (FDLib) [9]; SIFinder (SIF) [10]; Algorithm of the University of Surrey (UniS); FaceOnIt [12] (FoI, `http://www.faceonit.ch`); Neurotechnology VeriLook (VL, `http://www.neurotechnology.com`).

The result of each algorithm was evaluated by the following parameters:

- *False Rejection Rate* (FRR) — Ratio of type I errors, which indicates the probability of misclassification of the images containing a face;
- *False Acceptance Rate* (FAR) — Ratio of type II error, which indicates the probability of misclassification of the images not containing a face.

The total size of the test dataset is 59 888 images, namely, 11 677 faces and 48 211 non-faces. More information on experimental data and comparative testing of FD algorithms can be found in [3].

5 Results

The proposed face detector combining algorithm had been routinely tested in accordance with the procedure described in Degtyarev et al. [3]. The main idea of the procedure consists in computing FRR, FAR and vectors of algorithm's errors for each *Threshold* of the FD combining algorithm.

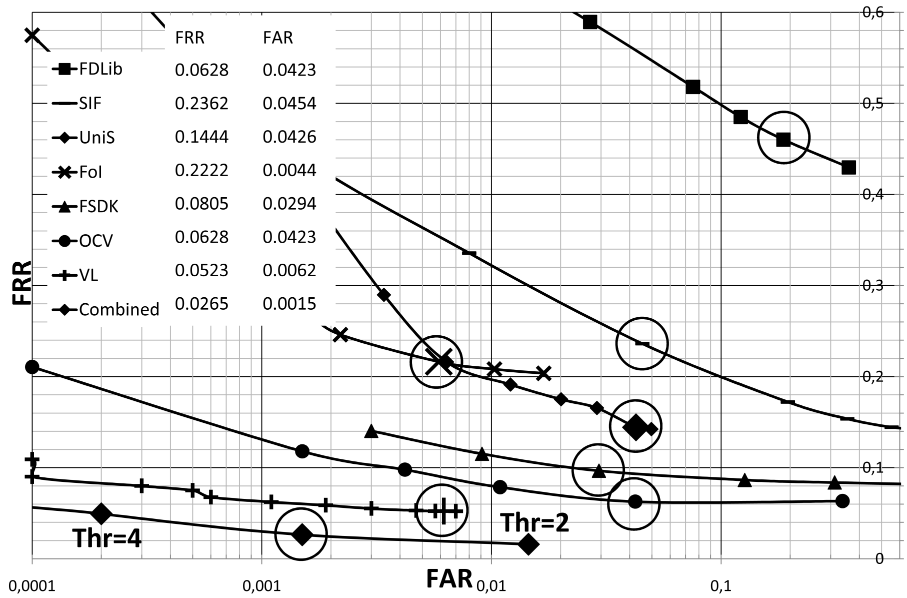

Fig. 4. The ROC plots "FAR in the log scale against FRR" as functions of the tuning parameter *Threshold*. The perfect performance would be the bottom left corner: $FRR = FAR = 0$. Circled points correspond to the tuning parameter value that delivers the minimal detection error for each algorithm.

The receiver operating characteristic (ROC curves) "FAR in the log scale against FRR in the standard biometric sense" [6, 15] for all the tested face detectors and their combination by our algorithm are presented in Fig. 4. These curves let us to identify the algorithm with the best overall performance, because the closer the curve to the perfect-performance-point $FRR = FAR = 0$ (the bottom left corner), the better the performance. As we can see, the proposed method of face detector combining does improve the performance of each of the FD algorithms to be combined for 2-3%. Nevertheless, there is still room for future development.

It is obvious that each of the algorithms have unique peculiarities of detection. One way to perform their numerical evaluation is to compare the number of images uniquely classified by each algorithm, as well as the numbers of "challenging" and "easy" images (see Table 1). Here the term *easy images* means the images detected by all algorithms, in the opposite case images are considered as *challenging*.

Table 1. Peculiar images distribution on the datasets.

Cases	**Number (faces)**	**% in DB**
easy images	38 478 (4 385)	64,25
challenging images	78 (78)	0,13
only OCV	5 (5)	< 0,01
only SIF	5 (5)	< 0,01
only FDL	3 (3)	< 0,01
only FSDK	10 (10)	0,02
only UniS	20 (20)	0,04
only FoI	22 (22)	0,04
only VL	49 (49)	0,08
only Comb	3 (3)	< 0,01

For a better understanding of the potential of the proposed FD combining method, let us take a look to some exemplary cases. As we can see in Fig. 5, all seven algorithms and their combination by the proposed method correctly detected the given faces. In Fig. 6 we can see two examples of non-face images where the two best algorithms (OCV and VL) falsely detected faces, whereas the proposed FD combining method did not find any faces in them, because there were no clusters of faces containing at least 3 elements. In Fig. 7, a much more interesting case is presented – the face correctly detected by the proposed FD combining method, whereas all the original algorithms failed, each of them correctly marked no more than one eye in this face. This case is noteworthy, because our algorithm does not detect any new faces, it only successively finds and merges the largest cluster of faces found by other face detectors. According Table. 1, there are also two cases like this one.

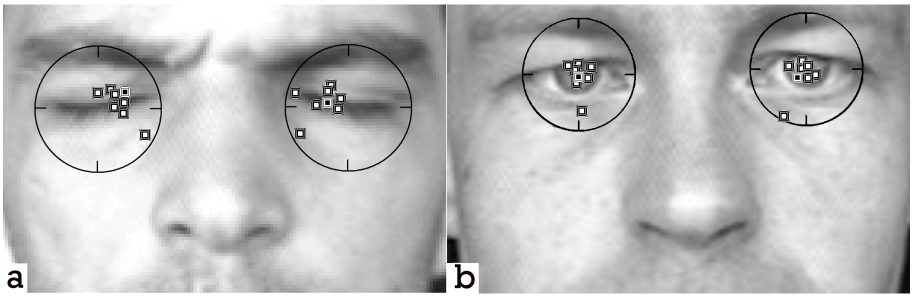

Fig. 5. Results of face detector combining on easy images containing faces; (a) – the pupils of the eyes are not visible; (b) – the pupils are visible.

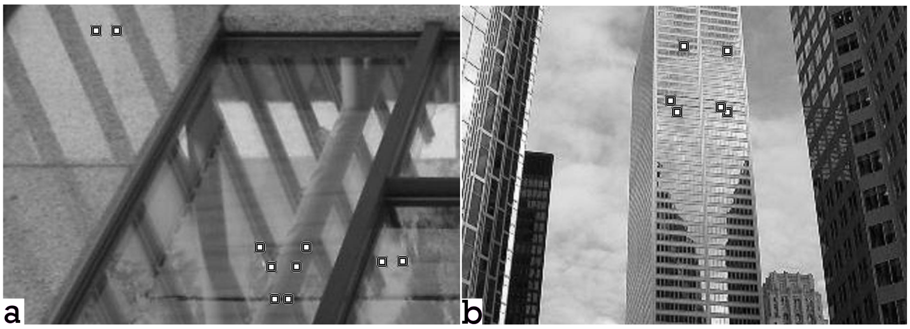

Fig. 6. Face detectors combining results on images not containing faces; two leading FD algorithms (OCV and VL) incorrectly detected faces; Combined algorithm did not find any suitable clusters of faces in given images, thus this images were considered to be *non faces*.

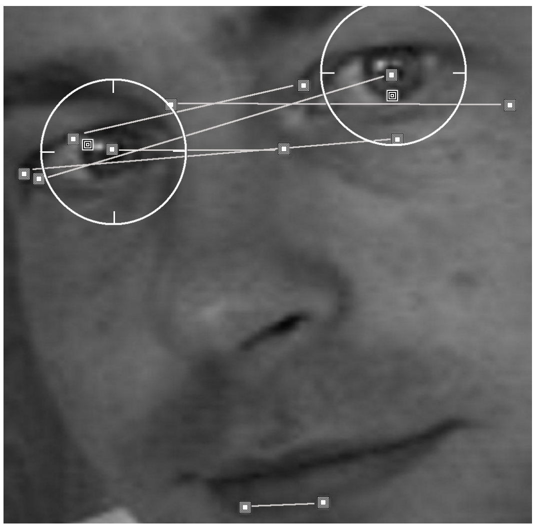

Fig. 7. "Challenging" case of face detectors combining; face correctly detected only by FD combining, because 5 algorithms' outputs (faces) formed a cluster.

6 Discussion and Conclusion

We have demonstrated in our experiments that the proposed method of FD combining has better performance (FRR 2.65%, FAR 0.15%) than each of its component algorithms. For comparison, VL and OCV, which are known to merge candidate windows according to some criteria, give, respectively, (FRR 5.23%, FAR 0.62%) and (FRR 6.54%, FAR 2.01%).

The method also has ability to correct some of detection errors made by all "elementary" algorithms (see Fig. 7). Nevertheless, there is a sufficient performance gap of $2-3\%$ FRR that separats it from the desired error-free result. This gap can be eliminated through further development, tuning of the FD combining algorithm and/or adding additional FD algorithms to the collection.

Perhaps, the most significant open question at this stage is the choice of the method of tuning free parameters in single algorithms to be combined. In this work, free parameters in each of the algorithms were chosen to deliver the minimal detection error $\left(\sqrt{FRR^2 + FAR^2} \to \min\right)$, that are not proven to be optimal for the FD combining task. It even may be better to combine a mix of the algorithms with two values of tuning parameters, delivering one the minimal FRR and the other the minimal FAR, because this would prevent forming false clusters and allow to lower the *Threshold* of cluster acceptance.

Exactly the same motivation leads us to another interesting and prospective idea – *self-combining*. It consists in combining results of only one algorithm, but with several different values of tuning parameters. Such an approach would allow us to eliminate misdetected faces, because they must not remain steady as the parameters will be changing, whereas true faces are expected to have fixed intervals of FD tuning parameters outside which they must disappear from the output of the respective algorithm.

Another interesting method of FD combining may consists in selecting a "most likely to be a face" region among originally detected alleged face regions using the decision tree learning approach. As features might be used, for instance distances between the results of algorithms, whether originally detected face regions were detected as faces by other FD algorithms or not, etc. Such method would return only originally detected face positions and not eliminate detection errors made simultaneously by all algorithms (see Fig. 7) in contrast to the proposed geometric method.

As mentioned above, the geometric approach takes into account only face positions in the outputs of detectors. Contrariwise, additional information provided by or linked to some FD algorithms may be helpful for combining face detectors, in particular, for finding the optimal different weights for outputs of different algorithms (FRR and FAR look to be suitable for this role).

It should be emphasized, that the aim of this work is only to show the possibility of face detector combining, and the proposed method is only the first attempt. We believe that disadvantages of this method will lead to much better approaches in solving of the above-mentioned problems.

7 Acknowledgements

The research leading to these results has received funding from the Russian Foundation for Basic Research, Grant No. 09-07-00394. We also would like to acknowledge many colleagues for discussions about this problem, in particular Glen Ivin, Alexey Kostin and Vadim Mottl.

References

1. Beveridge J. R., Alvarez A., Saraf J. et al.: Face Detection Algorithm and Feature Performance on FRGC 2.0 Imagery, Proceedings of First International Conference on Biometrics, pp. 1–7, IEEE (2007)
2. Castrillon, M., Deniz, O., Hernandez, D., Lorenzo, J.: A comparison of face and facial feature detectors based on the ViolaJones general object detection framework, Machine Vision and Applications, pp. 1–14, DOI: 10.1007/s00138-010-0250-7, Springer, Heidelberg (2010)
3. Degtyarev, N., Seredin, O.: Comparative Testing of Face Detection Algorithms. In A. E lmoataz et al. (Eds.): ICISP 2010, LNCS 6134, pp. 200–209, DOI: 10.1007/978-3-642-13681-8_24, Springer, Heidelberg (2010)
4. Degtyarev, N., Seredin, O.: Effect of Eyes Detection and Position Estimation Methods on the Accuracy of Comparative Testing of Face Detection Algorithms. In 10th International Conference on Pattern Recognition and Image Analysis: New Information Technologies (PRIA-10-2010). St. Petersburg, December 5-12, 2010, pp. 261–264, POLITECHNIKA (2010)
5. Duin, R. and Tax, D.: Experiments with classifier combining rules, Multiple Classifier Systems, pp. 16–29, Springer (2000)
6. Fawcett T.: An introduction to ROC analysis, Pattern Recognition Letters, Volume 27, Issue 8., pp. 861–874 (2006)
7. Jesorsky, O., Kirchberg, K.J., Frischholz, R.W.: Robust face detection using the hausdorff distance, Lecture Notes in Computer Science, June, pp. 90-95 (2001)
8. Jones, M.J.:Face Recognition: Where We Are and Where To Go From Here, IEEJ Trans. on Elect., Information and Systems, Volume 129, Issue 5, pp. 770–777 (2009)
9. Kienzle, W., Bakir, G., Franz, M., Scholkopf, B.: Face detection – efficient and rank deficient, Advan. in neural inform. process. systems 17, 2005. – pp. 673–680 (2005)
10. Krestinin, I.A., Seredin, O.S.: Excluding cascading classifier for face detection. Proc. of the 19th Int. Conf. on Computer Graphics and Vision, pp. 380–381 (2009)
11. Kittler J., Combining classifiers: A theoretical framework, Pattern Analysis & Applications, Vol. 1, No. 1, pp. 18–27, (1998)
12. Marcel, S., Keomany, J., et al.: Robust-to-illumination face localisation using Active Shape Models and Local Binary Patterns, IDIAP-RR, V. 47, (2006)
13. Mohamed, N.M., Mahdi, H.: A simple evaluation of face detection algorithms using unpublished static images, in proc. of 10th Intern. Conference on Intelligent Systems Design and Applications, pp. 1–5, DOI: 10.1109/ISDA.2010.5687301, IEEE (2010)
14. Viola, Paul and Jones, Michael J.: Robust Real-Time Face Detection, International Journal of Computer Vision, Volume 57, Issue 2, pp. 137–154 (2004)
15. Wechsler H.: Reliable face recognition methods: system design, implementation and evaluation, Springer, 329 p. (2007)