MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan
3-25

Fast Detection of Multi-View Face and Eye Based on Cascaded Classifier

Jung-Bae Kim, Seok-Cheol Kee and Ji-Yeon Kim
Computing Lab., Samsung Advanced Institute of Technology
P.O.BOX. 111, Suwon, Gyeonggi-do 440-600 Korea
{jung-bae.kim, sckee, jiyeon.kim}@samsung.com

Abstract

In our multi-view face and eye detection, we use a cascaded classifier trained by gentle AdaBoost algorithm, one of the appearance-based pattern learning method. Specifically, in order to detect multi-view face, we propose a special cascaded classifier using coarse-to-fine search, simple-to-complex search, and parallel-to-separated search. In order to detect eye, we propose a four-step eye detection method. Using proposed methods, we got face and eye detection ratios as 99.5%, 88.3%, respectively for 7 different DBs including 17,018 various multi-view faces.

1 Introduction

Multi-view face and eye detection technology is a core technology to face recognition and image retrieval. There can be three kinds of head rotation. X-axis rotation is up-down nodding, y-axis rotation is out-of-plane rotation (profile view), and z-axis rotation is in-plane rotation (left-right head leaning).

Rowley et al. detected z-axis rotated faces [1]. Schneiderman et al. detected y-axis rotated [2]. In order to detect y-axis rotated faces and z-axis rotated faces at the same time, Li et al. rotated the input image with three different angles with respect to z-axis and applied detector-pyramid detecting y-axis rotated faces [3]. Jones and Viola made y-axis rotated face detector and z-axis rotated face detector independently [4]. Wu et al. used a pose estimator in early stage and applied narrow view face detectors according to the result of the pose estimator [5]. [1] and [4] also used pose estimator method. However, errors from pose estimator deteriorated the face detection rate. So far, various methods on multi-view face detection are proposed. However, they show only limited performance.

To solve these problems, we use basically a classifier trained by gentle AdaBoost algorithm, one of the appearance-based pattern learning method. In order to detect multi-view face, we propose a special cascaded classifier structure using coarse-to-fine search, simple-to-complex search and parallel-to-separated search.

On the other hand, it is a very challenging work to detect eye pairs in various real environments. There may be many varying factors such as scale, pose, rotation, closed eye, illumination, glass reflection, occlusion, etc.

Recent researches proposed various methods to detect eye pairs. However, most of all consider only some part of variation. Kawaguchi et al. and Baskan et al. used face structure knowledge such as Hough transform, symmetry detector, projection analysis [6][7]. These methods do not consider physical features that eyes have much variation. In addition, these methods use a binarized image with threshold value. Eye candidate set resulted from thresholding has real eye with low probability. As our experiment, it is only 90.8%. It means that any classifier following the binarization process cannot have eye detection ratio over than 90.3%. Lucey et al. tried to use a learning method on eye detection [8]. However, it could not discriminate between a thick glass frame and a closed eye since it used only eye information. So far, various eye detection methods are proposed. However, they also show limited performance.

In this paper, we propose a four-step eye detection method. In first step, we restrict the both eyes' candidate areas from an input image. In second step, we detect both eyes' candidates using an eye classifier. In third step, we extract eye pair candidates' feature using an eye pair classifier. In fourth step, we decide an eye pair based on the three features extracted from previous two steps.

2 Gentle AdaBoost

We adopt an AdaBoost algorithm to implement multi-view face detection and eye detection. This algorithm was first proposed by Freund and Schapire [9] and it became popular in the face detection field after Viola and Jones [10]. AdaBoost is a very effective learning algorithm that organizes simple and fast weak classifiers as a weighted sum and renders a fast strong classifier having a high success rate. Specifically, we use a gentle AdaBoost [9], a gentle version of real AdaBoost, as follows:

1) Given N examples $(x_1, y_1), \ldots, (x_N, y_N)$ with $x \in R^k$, $y_i \in \{-1, 1\}$

2) Start with weights $w_i = 1/N$, $i = 1, \ldots, N$.

3) Repeat for $k = 1, \ldots, K$.

(a) Find a weak classifier $f_k(x)$, which minimizes

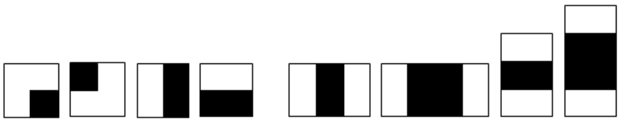
$$\varepsilon = \sum_{i=1}^{N} w_i (y_i - f_k(x_i))^2$$

(b) Set $w_i \leftarrow w_i \cdot \exp(-y_i \cdot f_k(x_i)), i = 1, \cdots, N$ and renormalize weights so that $\sum_{i=1}^{N} w_i = 1$

4) Output the classifier $\text{sign}\left\{\sum_{k=1}^{K} f_k(x)\right\}$

A weak classifier consists of a threshold value and a simple feature, as shown in Fig. 1. The specific values are determined according to an iterative learning. The simple features are designed to detect an edge or a line of the face easily. Several supplementary features are added to Viola and Jones' features [10].

In the learning stage, all possible positions, sizes, and types of features are considered within a 24×24 window. There are a total number of 117,400 features with some restrictions to their freedom such as minimum area. A face/non-face strong classifier is applied to the input image in all possible positions with all possible sizes in order to detect all the faces with various positions and sizes.



(a) (b)

Fig. 1. Haar-like simple features: (a) edge features; (b) line features.

3 Multi-View Face Detection

3.1 Detectable Rotation Angle

For x-axis rotation, we need only 2 detectors. They are a down-view face detector covering [-60°, -20°] and a frontal/upward face detector covering [-20°, 50°]. For y-axis rotation, we need 3 detectors including a left-view face detector, a frontal face detector and a right-view face detector. Their covering angles are [-90°, -20°], [-20°, 20°], [20°, 90°], respectively.

For z-axis rotation, we deal with all rotation covering [-180°, 180°]. However, during standing, a person can lean his/her head with [-45°, 45°]. We call "Basic mode" of z-axis rotation as [-45°, 45°] and "Extended mode" of z-axis rotation as [-180°, 180°].

When we design a detector covering 30° on z-axis rotation, 12 detectors covers the extended mode, [-180°, 180°]. For the basic mode, 3 detectors are sufficient. This is called as "Method I." When we design a detector covers 45°, 8 detectors can covers the extended mode. For the basic mode, 2 detectors are sufficient. This is called as "Method II." When we consider x, y, z-axis rotation, the number of face detectors needed is shown in Table I.

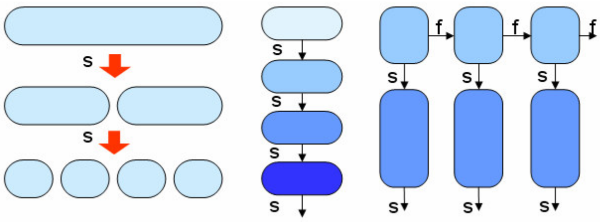
Table I. The numbers of individual face detectors needed in multi-view face detection.

		Number of face Detectors needed.	Number of face detectors to be trained
Method I	Basic mode	18=2×3×3	10=2×5
	Extended mode	72=2×3×12	10=2×5
Method II	Basic mode	12=2×3×2	6=2×3
	Extended mode	48=2×3×8	6=2×3

However, according as we rotate the simple feature of a detector with 90° or perform a mirroring operation on the simple feature, we get other detectors. For example, a left-view detector can be a right-view detector. Also, 12 frontal-view detectors can be made from only 2 detectors. As the result, the number of face detectors to be trained is shown in Table I.

3.2 The Structure of Multi-View Face Detector

In order to detect multi-view face, when we use 12 or 72 face detectors independently, we need 12 times or 72 times of computation comparing to a single face detector. If we adopt a pose estimator, there happens a pose estimator' error problem as we mentioned. We propose a new multi-view detector based on a special cascaded classifier structure using coarse-to-fine search, simple-to-complex search, and parallel-to-separated search as shown in Fig. 2.



(a) (b) (c)

Fig. 2. Three methods rendering a cascaded classifier used in the multi-view face detector; (a) Coarse-to-fine search; (b) Simple-to-complex search; (c) Parallel-to-separated search.

Coarse-to-fine search is that a whole-view classifier is located in early stage and narrower-view classifiers are located in late stage. Simple-to-complex search is that easier classifiers are located in early stage and more complex classifiers are located in late stage. Using these two stages, we can speed a multi-view detector up since most non-faces are eliminated in early stages.

Parallel-to-separated search is that all detectors are arranged in parallel until K stage and each detector is arranged separately from K+1 stage. Usually, when an input is entered and a face is detected successfully in a certain stage of a view, the input moves to next stage of the same view to be tested. Parallel arrangement means that when a face is not detected, the input moves to the same stage of next view. Separated arrangement means that when a face is not detected, no more procedure is needed. We just decide the input is a non-face. Using this method, an input image is decided as a certain view in early stage, and then we concentrate on whether the input is a face or not in late stage. In Fig. 2, s stands for success in face detection and f stands for failure.

We combine three cascading methods into a multi-view face detector as shown in Fig. 3. In 1~2 stage, whole-view face is detected. In 3~4 stage, face groups such as upright face, left-leaned face, right-leaned face are detected. In 5~M stage, all view face are detected. Also, parallel searches are implemented until K stage and separated searches are implemented from K+1 stage. In Fig. 3, NF stands for non-face, s stands for success in face detection, and f stands for failure.

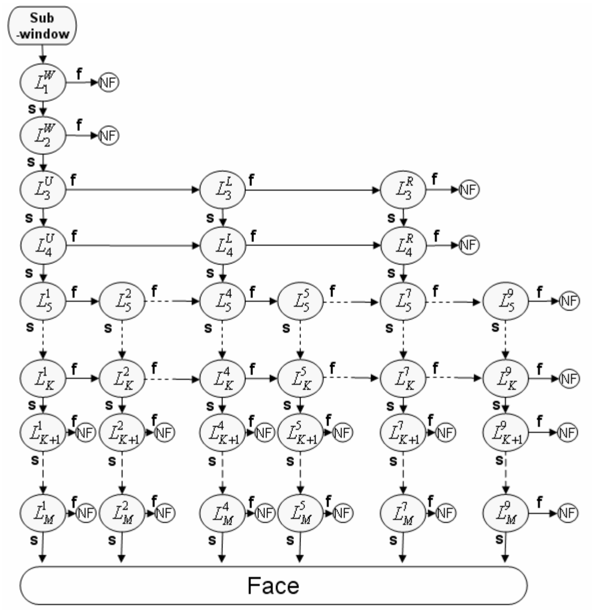


Fig. 3. The structure of multi-view face detector.

4 Eye Detection

Eye detector decides the center position of eye. We propose a four-step eye detection method as shown in Fig. 4.

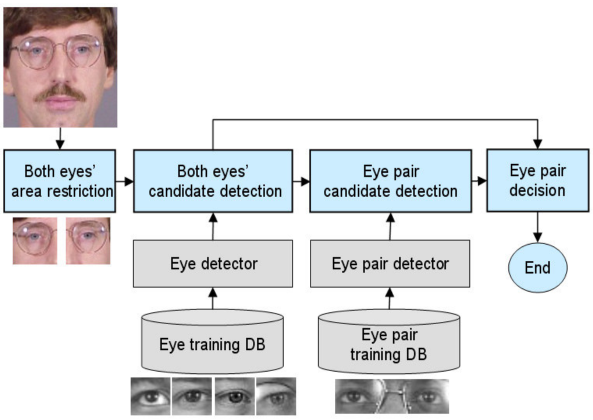


Fig. 4. The structure of eye detector.

In both eyes' area restriction step, both eyes' areas are clipped from the input image according to the face view, and are normalize into 50×50 size.

In both eyes' candidate detection step, a classifier trained by AdaBoost detects both eyes. Training DB is composed of 6,000 eye samples as shown in Fig. 4. Their width:height ratio is 1:1. It means that they include eye and eye neighbors. We apply only simple-to-complex search on eye detector. We select sub-windows, which reach the latest stage, as the eye candidates. For example, the eye detector has 15 stages and any sub-window cannot reach the last stage. But, 4 sub-windows reach 13 stages. Then, these 4 sub-windows become eye candidates.

If the eye candidates occlude with over 80% area, they are merged into an average size and position. When *k* eye candidates are merged, we say that neighbor number is *k*. Neighbor number is used as a feature in eye pair decision step.

In eye pair candidate detection step, we extract a feature used in eye pair decision step. When there are 3 left eye candidates and 2 right eye candidates, the number of possible eye pairs is 6. The 6 eye pairs become inputs in this step. We detect the eye pairs using a trained eye pair detector. For this, we made a training DB including both eyes' image and between the eyes' image.

If the left eye and right eye do not have same height, we should rotate the image to have an upright image. The procedure needs much computation. To solve it, we propose a method as shown in Fig. 5. To get an eye pair image, we clip both eyes' image and between the eyes' image separately, and connect them into one image. This method has a little computation and same performance.

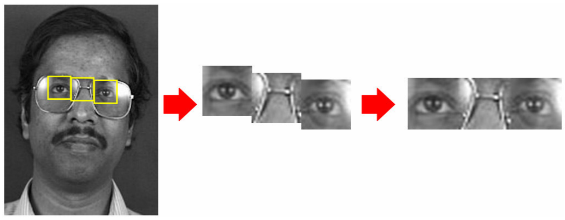


Fig. 5. The method to rendering an eye pair image.

In eye pair decision step, we decide an eye pair based on a confidence level, which is the sum of three features as follows:

Feature 1: the number of final stage proceeded in the eye pair candidate detection step.

Feature 2: the sum of left eye's neighbor number and right eye's neighbor number.

Feature 3: $-dx-dy$

where $dx = \left|\frac{L_x^n + R_x^n}{2} - 25\right|$, $dy = \left|L_y^n - R_y^n\right|$, left eye position on 50×50 image: (L_x^n, L_y^n), and right eye position on 50×50 image: (R_x^n, R_y^n).

One eye pair having a maximum confidence level is decided as the eye pair.

5 Experimental Results

5.1 Test DB and Performance Measures

Reducing the dependency of test DB, we use 7 different face DBs 17,018 images. 5 DBs 9,458 images are all frontal faces and 2 DBs 7,560 images are all non-frontal faces.

When a detected face includes ground truth eye position, we call it as "success in face detection." When the distance between the detected eye and ground truth eye is less than 1/10 of distance between both ground truth eyes, we call it as "success in eye detection." Based on these concepts, we define three detection ratio, Y, Y_1, and Y_2 as follows:

Y = eye detection ratio with respect to total image.

Y_1= (number of images succeeding in face detection) / (total number of images)

Y_2= (number of images succeeding in eye detection) / (number of images succeeding in face detection)

$Y = Y_1 \times Y_2$

5.2 The Performance of the paper

We use a somewhat restricted view detector to have a maximum performance on test DB. Specifically, all faces rotated with respect to y-axis could be detected. However, the view is restricted in x and z-axis rotation. The detection results are shown in Fig. 6.

The detection ratio of the paper is shown in Table II and the detectable rotation angle is shown in Table III. The detection speed on 320×240 images is 15 frames/sec with a multi-view face detector and 30 frames/sec with a frontal face detector.

Fig. 6. Examples of experimental results for multi-view face and eye detection.

Table II. The face and eye detection ratio.

Frontal DB (9,458 images)			Non-frontal DB (7,560 images)			Total DB (17,018 images)		
Y_1	Y_2	Y	Y_1	Y_2	Y	Y_1	Y_2	Y
99.9	94.2	94.1	99.0	80.7	79.9	99.5	88.3	87.8

Table III. The detectable rotation angle (%)

x-axis rotation	y-axis rotation	z-axis rotation
[-22°, 40°]	[-90°, 90°]	[-20°, 20°]

6 Conclusion

So far, we dealt with a multi-view face and an eye detection used as core technologies to face recognition and image retrieval. In both detections, we used cascaded classifiers trained by gentle AdaBoost algorithm, one of the appearance-based pattern learning method having fast and good classification performance. In order to detect multi-view face, we propose a special cascaded classifier including coarse-to-fine search, simple-to-complex search, and parallel-to-separated search. In order to detect eye, we propose a four-step eye detection method. Based on these proposed methods, we got high performance on face and eye detection.

However, to get higher face recognition rate, more precise eye detection is needed. Also, more robust face detection and eye detection are needed even though there are some occlusions. Therefore, further researches should be concentrated on these areas.

References

[1] H. Rowley, S. Baluja and T. Kanade, "Neural Network-Based Face Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23-28, 1998.

[2] H. Schneiderman and T. Kanade, "Object Detection Using the Statistics of Parts," *Int'l Journal of Computer Vision*, vol. 56, no. 3, pp. 151-177, 2004.

[3] S.Z. Li and Z.Q. Zhang, "FloatBoost Learning and Statistical Face Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1112-1123, 2004.

[4] M. Jones and P. Viola, "Fast Multi-view Face Detection," Proc. of Computer Vision and Pattern Recognition. 2003.

[5] B. Wu, H. Ai, C. Huang and S. Lao, "Fast Rotation Invariant Multi-View Face Detection Based on Real Adaboost," *Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pp. 409-414, 2004.

[6] T. Kawaguchi, D. Hikada and M. Rizon, "Detection of the eyes from human faces by hough transform and separability filter," *Proc. of IEEE Int'l Conf. on Image Processing*, pp. 49-52, 2000.

[7] S. Baskan, M.M. Bulut, V. Atalay, "Projection based method for segmentation of human face and its evaluation," *Pattern Recognition Letters*, vol. 23, no. 14, pp. 1623-1629, 2002.

[8] S. Lucey, S. Sridharan and V. Chandran, "Improved facial feature detection for AVSP via unsupervised clustering and discriminant analysis," *EURASIP Journal on Applied Signal Processing*, vol. 3, pp. 264-275, 2003.

[9] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," *Proc. of Int'l Conf. on Machine Learning*, pp. 148-156, 1996.

[10] P. Viola and M. Jones, "Robust Real-time Object Detection," *Proc. of IEEE Int'l Conf. on Computer Vision 2001 Workshop on Statistical and Computation Theories of Vision*, 2001.