2D Cascaded AdaBoost for Eye Localization

Zhiheng Niu[1], Shiguang Shan[2], Shengye Yan[2], Xilin Chen[1, 2] and Wen Gao[1, 2]
[1] *School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China*
[2] *ICT-ISVISION FRJDL, Institute of Computing Technology, CAS, Beijing, China*
{zhniu, sgshan, syyan, xlchen, wgao}@jdl.ac.cn

Abstract

In this paper, 2D cascaded AdaBoost, a novel classifier designing framework, is presented and applied to eye localization. By the term "2D", we mean that in our method there are two cascade classifiers in two directions: The first one is a cascade designed by bootstrapping the positive samples, and the second one, as the component classifiers of the first one, is cascaded by bootstrapping the negative samples (please refer to Fig.1). The advantages of the 2D structure include: (1) It greatly facilitates the classifier designing on huge-scale training set; (2) It can easily deal with the significant variations within the positive (or negative) samples; (3) Both the training and testing procedures are more efficient. The proposed structure is applied to eye localization and evaluated on four public face databases, extensive experimental results verified the effectiveness, efficiency, and robustness of the proposed method.

1. Introduction

In most pattern recognition and computer vision tasks, the localization and alignment of target object in one static image is a task of great importance. Particularly, it is believed that the performance of face-related vision systems (e.g. face recognition, gaze tracking, facial expression analysis and human-computer interaction) may be greatly degraded due to the imprecise localization of the eyes [1].

There are many researchers focusing on tackling the eye detection tasks. Compared with the active IR based approaches [2], the image based passive approaches are widely used for no extra equipment is needed. Generally, approaches for eye localization can be classified into three categories: template based methods, feature based methods and appearance based methods. In the template based methods [3], a generic eye model, based on the eye shape, is designed first. Template matching is then used to search for the eyes in the image. But the locating results are heavily affected by the eye model initialization and the image contrast. The expensive time cost also prevents its wide application. Feature based methods explore the characteristics (such as edge and intensity of iris, the color distributions of the sclera and the flesh) of the eyes to identify some distinctive features around the eyes. But if the eye is closed or partially occluded by hair or due to face orientation, it will fail. Zhou et al. [4] use Generalized Projection Function (GPF) to detect the eyes. The appearance based methods detect eyes based on their photometric appearance. To well represent the eyes of different subjects, these methods usually need a large amount of training data under different face orientations and different illumination conditions. ASM [5] and AAM [6] are widely used for landmarks localization. And some other works are published for eye localization. Jesorsky et al. use Haussdorf distance on edge images to complete the task [7]. Chen et al. use boosted detectors to detect eyes, nose and mouth corners [8].

In this paper, we propose a real-time robust method for eye localization across pose and lighting variations, and partial occlusions, based on 2D cascaded framework. In the general AdaBoost framework, the positive and negative samples are always considered as a whole which results in many problems: first, it can not deal with large-scale training set containing a great number of positive and negative samples due to its time-space requirements; second, the convergence cannot be guaranteed because of the significant variations within not only the negative samples but also the positive ones due to various intrinsic or extrinsic factors; third, the error rate of the classifier may be very high; Finally, the speed of both the training and testing procedure is low. Aiming at these problems, a 2D cascaded AdaBoost framework is proposed.

0-7695-2521-0/06/$20.00 (c) 2006 IEEE

2. 2D cascaded AdaBoost

AdaBoost classifier using bootstrap on the negative examples is described by Viola and Jones [9]. Our method bootstraps both the negative and positive examples in a similar way, as illustrated in Figure 1.

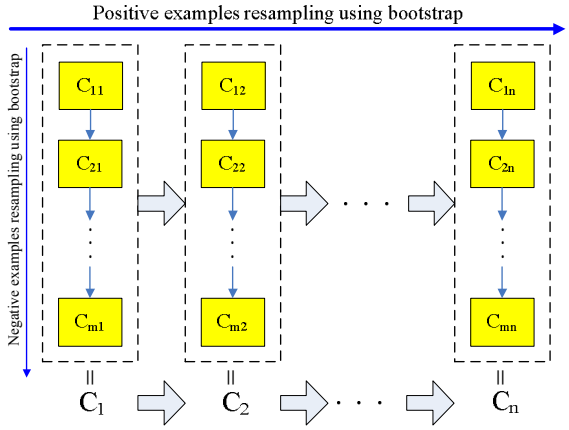

Figure 1. Framework of 2D cascaded AdaBoost

2.1. Training Procedure

The training set contains five databases: FERET, BANCA, FRGC, and two face data bases build by our lab. There are nearly 30,000 face images under almost all of the usual circumstances. We use only the left eye for training. The positive samples (cropped eye images) are expanded by scaling and rotation, so the total training set is nearly 150,000. The negative samples are the windows cropped some distance away from the eye region. Therefore both of the positive and negative samples are incompact and large sized, and can not be manageable by AdaBoost classifier.

In our method, we first randomly down sample the positive samples to 10,000, using the 10,000 positive samples and all the negative samples to train a cascaded AdaBoost classifier C_1 with bootstrap on the negative samples, but the detection rate can be set to much lower, 50% in this paper. Then we bootstrap the 150,000 positive samples to obtain about 75,000 which can not be correctly classified by C_1, and down sampled to 10,000 to train classifier C_2. After n times the training procedure terminates. Table 1 shows the performance changes of component classifier C_{ij} with the increase of i, j. ↑ indicates increasing and ↓ indicates decreasing. From top left to bottom right the task becomes more and more tough, and the procedure becomes more and more time-consuming. And, C_1 is completed using only dozens of Haar features, while C_n may need thousands of them.

Table 1. Characteristics of the 2D cascades

	Complexity of samples	Volume of features	Detection rate	False alarm	Time needed
Horizontal ($C_{i1} \sim C_{in}$)	↑ (positive)	↑	↑	↑	↑
Vertical ($C_{1j} \sim C_{mj}$)	↑ (negative)	↑	↓	↓	↑

2.2. Localization Procedure

Our localization procedure depends on a face detector who gives the approximate eye region. The eye detector C_i searches in this region (the right eye region is mirrored). Then the location of eye L_i is determined by C_i, and we finally fuse the locations to an accurate one.

2.2.1. Localization by single classifier C_i. We denote the total number of detected windows to be N_i, the center of p^{th} detected window to be $C(p)$. And the location of eye is calculated by:

$$L_i = \arg\max_{1\le p\le N_i} \sum_{q=1}^{N_i} \varphi(p,q) \tag{1}$$

$$\text{where } \varphi(p,q) = \begin{cases} 1 & if \ \ \|C(p)-C(q)\| < \varepsilon \\ 0 & else \end{cases} \tag{2}$$

2.2.2. Localization by weighting all classifiers. Using all the classifiers can achieve precise localization result. But the procedure is somewhat time-consuming, mainly because the last classifiers are much more complex and using much more features than the first ones. The final localization L is calculated by:

$$L = \frac{1}{\|W\|} \sum_{i=1}^{n} W_i L_i \tag{3}$$

$$\text{where } W_i = \begin{cases} 1 & if \ \ N_i > 0 \\ 0 & else \end{cases} \tag{4}$$

2.2.3. Localization by cascading all classifiers. Landmarks localization is different from object detection. The later is to determine whether there are any objects within a given image, and return the location and extent of each object in the image if one or more ones are present. However, the former is to localize the landmarks in the denoted region, which not only are present but also appear only once. Thus from C_1 to C_n, once the eye is detected the procedure stops. Send not for a hatchet with which to break open an egg. There is no need to localize a good conditioned image with the last complex classifiers. Therefore the easier image is, the faster the speed is. The algorithm is simply formulated as:

$$L = L_i \left(N_i > 0 \text{ and for all } 0<j<i,\ N_j = 0 \right) \tag{5}$$

3. Experiments

To evaluate our method fairly and comprehensively, we chose the same databases and evaluation protocol which is also used in previous works [7, 10]. We also compare and analyze several different solutions based on multi classifier fusion. The robustness analysis to different image conditions is given finally.

3.1. Databases

We use the XM2VTS database and the BioID database, which are both used for algorithm testing in Jesorsky [7] and Hamouz's [10] methods. The XM2VTS database contains 1180 color images, each one showing the face of one out of 295 different test persons. The BIOID database consists of 1521 images of 23 different persons and has been recorded during several sessions at different places. Compared with the XM2VTS this set features a larger variety of illuminations, backgrounds and face sizes. We also test our algorithm on the CMU Pose, Illumination, and Expression (PIE) [11] face database (68 images in each subset) and JAFFE [12] database (213 images totally) with mainly expression variations.

Note that all of the databases we test on are strictly excluded from our training set, unlike some other researchers to use half of the database for training and the rest for test. The training set of our algorithm is composed of the five face databases mentioned in the section 2.1.

3.2. Evaluation protocol

In our experiment, we adopt the stringent localization criterion to evaluate the error rate which is proposed in work [7]. It records the maximum point to point error over both eye point predictions, which has been normalized by the known inter-ocular separation. The measure is defined as follows:

$$Err = \frac{\max(d_l, d_r)}{\|C_l - C_r\|} \tag{6}$$

where C_l, C_r are the ground truth eye center coordinates and d_l, d_r are distances between the detected eye centers and the ground truth ones. Figure 2 and Figure 3 are shown as the proportion (y) of testing samples under an error metric (x):

$$y = P(Err \le x) \tag{7}$$

3.3. Results

It should be mentioned that the error rate of face detector is not deducted from the whole. And some previous work took the error level of Err < 0.25 as successful localization, but we think that Err < 0.1 would be more appropriate.

3.3.1. Comparisons with other researchers' work. We compare our method (weighting all classifiers) with that of Jesorsky [7] (AVBPA01), Hamouz [10] (FG04) who reported results for the same databases and experimental protocol, and also Zhou [4] (PR04).

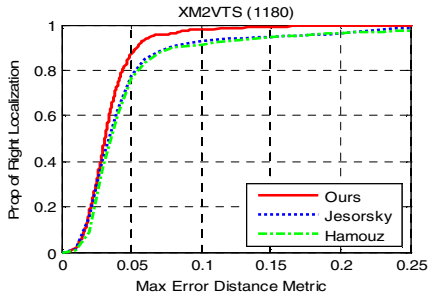

Figure 2. Results on the XM2VTS database

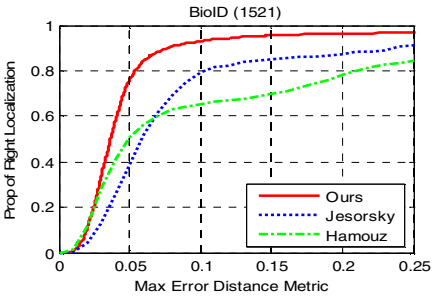

Figure 3. Results on the BioID database

Table 2. Comparisons with GPF [4] (PR04)

Database	BioID	JAFFE
Ours	93.0 (Err<0.1)	100 (Err<0.1)
GPF [4]	94.8 (Err<0.25)	97.2 (Err<0.25)

From these results, it is obvious that our algorithm performs much better than other methods, such as Jesorsky, Hamouz and Zhou's methods even at the error level of Error < 0.1.

3.3.2. Comparisons with different solutions. Our method utilizes all of the samples to train a 2D cascaded AdaBoost classifier. We also design the experiments on training traditional cascaded AdaBoost classifiers with random down sampling and bagging, which can only use part of the positive samples. For the 150,000 positive samples we randomly sample

10,000 each time to train a cascaded AdaBoost classifier, and we do this 3 times for bagging.

Table 3. Comparisons with different solutions

Solutions		2D Cascades (Weighting)	2D Cascades (Cascading)	Bagging	Random sampling
Prop. (%) (Err<0.1)	XM2VTS	98.1	97.0	97.2	96.8
	BioID	93.0	91.4	91.1	91.0
Speed(ms/image) (2.4G CPU)		~1200	~25	~1300	~420

Table 3 shows that the 2D cascaded method with weighting all classifiers (Section **2.2.2**) is most accurate, but somewhat time-consuming. 2D cascaded method with Cascading all classifiers (Section **2.2.3**) is still accurate enough but extreme efficient. The solution of random sampling is hard to converge, therefore it is less efficient, and every time the classifier has similar performance. Thus improvement of bagging is limited on accuracy, and the speed is 3 times lower.

3.3.3. Robustness Analysis. To investigate the robustness of our algorithm to pose and lighting variances, we test our algorithm on the PIE face database and the results are given in table 4 and table 5.

Table 4. Results on PIE database (Pose)

Pose(°)	-45	-22.5	0	+22.5	+45
Proportion (%) (Err<0.1)	77.9	97.1	100	98.5	75

Table 5. Results on PIE database (Lighting)

Lighting position	Full left	Half left	Front	Half right	Full right
Proportion (%) (Err<0.1)	92.6	98.5	100	97.1	94.1

Our method achieves high accuracy at the error level of Err < 0.1 in most of the case on PIE database, except for large pose variance of yaw rotation. However this is mainly because of the failure of near frontal face detector. It proves that our method is very robust under the usual circumstances, even if full side lighting makes the eye almost invisible.

4. Summary

In this paper we introduced a framework of 2D cascaded AdaBoost for eye localization. This framework can efficiently deal with tough training set with vast and incompact positive and negative samples by two-direction bootstrap strategy. And the 2D cascaded classifier with cascading all the sub-classifiers in localization period can also speed up the procedure and achieve high accuracy. Extensive experiments indicate that our method is very accurate, efficient and robust under usual condition.

5. Acknowledgement

This research is partially sponsored by Natural Science Foundation of China under contract No.60332010, "100 Talents Program" of CAS and ISVISION Technologies Co. Ltd, and the Program for New Century Excellent Talents in University (NCET-04-0320).

References

[1] S.Shan, Y.Chang, W.Gao, B.Cao, "Curse of Mis-Alignment in Face Recognition: Problem and A Novel Mis-Alignment Learning Solution", *Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR'04)*, Korea, 2004, pp. 314-320.

[2] Z. Zhu, Q. Ji, "Real-time eye detection and tracking under various light conditions", *Proceedings of ACM SIGCHI Symposium on Eye Tracking Research and Applications*, New Orleans, 2002.

[3] A. Yuille, P. Hallinan, D. Cohen, "Feature extraction from faces using deformable templates", *International Journal of Computer Vision*, 1992, pp. 99-111.

[4] Z. Zhou and X. Geng, "Projection Functions for Eye Detection", *Pattern Recognition*, 2004.

[5] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham, "Active Shape Models - Their Training and Application", *In Computer Vision and Image Understanding*, January 1995, pp. 38-59.

[6] T.F. Cootes, G.J. Edwards, and C.J. Taylor, "Active appearance models", *In 5th European Conference on Computer Vision*, Berlin, 1998, pp. 484-498.

[7] Oliver Jesorsky, Klaus J. Kirchberg, and Robert W. Frischholz, "Robust Face Detection Using The Hausdorff Distance", *The 3rd International Conference on Audio and Video-Based Biometric Person Authentication*, Verlag Berlin Heidelberg, 2001, pp. 91-95.

[8] L. Chen, L. Zhang, L. Zhu, M. Li, H. Zhang, "A Novel Facial Feature Localization Method Using Probabilistic-like Output", *Asian Conference on Computer Vision* , 2004, pp. 1-10.

[9] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", *In Computer Vision and Pattern Recognition Conference* 2001, Kauai, Hawaii, 2001, pp. 511-518.

[10] M. Hamouz, J. Kittler, J.K. Kamarainen, and H. Kalvioinen, "Affine-Invariant Face Detection And Localization Using Gmm-Based Feature Detectors And Enhanced Appearance Model", *FGR'04*, 2004, pp. 67-72.

[11] T. Sim, S. Baker and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) Database", *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FGR'02)*, 2002, pp. 46-51.

[12] M.J. Lyons, S. Akamatsu, M. Kamachi and J. Gyoba, "Coding Facial Expressions with Gabor Wavelets", *Proceedings of 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, pp. 200-205.