3D Cascade of Classifiers for Open and Closed Eye Detection in Driver Distraction Monitoring

Mahdi Rezaei and Reinhard Klette

The *.enpeda..* Project, The University of Auckland
Tamaki Innovation Campus, Auckland, New Zealand
mrez010@aucklanduni.ac.nz , r.klette@auckland.ac.nz

Abstract. Eye status detection and localization is a fundamental step for driver awareness detection. The efficiency of any learning-based object detection method highly depends on the training dataset as well as learning parameters. The research develops optimum values of Haar-training parameters to create a nested cascade of classifiers for real-time eye status detection. The detectors can detect eye-status of open, closed, or diverted not only from frontal faces but also for rotated or tilted head poses. We discuss the unique features of our robust training database that significantly influenced the detection performance. The system has been practically implemented and tested in real-world and real-time processing with satisfactory results on determining driver's level of vigilance.

1 Introduction

The automotive industries implements active safety systems into their top-end cars for lane departure warning, safe distance driving, stop and speed sign recognition, and currently also first systems for driver monitoring [Wardlaw 2011]. Stereo vision or pedestrian detection are further examples of components of a driver assistant system (DAS).

Any sort of driver distraction and drowsiness can lead to catastrophic cases of traffic crashes not only for the driver and passengers in the *ego-vehicle* (i.e. the car the DAS is operating in) but also for surrounding traffic participants. Face pose and eye status are two main features for evaluating a driver's level of fatigue, drowsiness, distraction or drunkenness. Successful methods for face detection emerged in the 2000s. Research is now focusing on real time eye detection. Concerns in eye detection still exist for non-forward looking face positions, tilted heads, occlusion by eye-glasses, or restricted lightening conditions.

According to [Zhang and Zhang 2010], research on eye localization can be classified into four categories. *Knowledge-based methods* include some predefined rules for eye detection. *Template-matching methods* generally judge the presence or absence of an eye based on a generic eye shape as a reference; a search for eyes can be in the whole image or in pre-selected windows. Since eye models vary for different people, the locating results are heavily affected by eye model initialization and image contrast. High computational cost also prevents a wide application for this method. *Feature-based approaches* are based on fundamental eye-structures; typically a method starts here with determining properties such

A. Berciano et al. (Eds.): CAIP 2011, LNCS 6855, pp. 171–179, 2011.

as edges, intensity of the iris and sclera, plus colour distributions of the skin around eyes to identify 'main features' of eyes [Niu et al. 2006]. This approach is relatively robust to lightning but fails in case of face rotation or eye occlusion (e.g. by hair or eye-glasses). *Appearance-based methods* learn different types of eyes from a large dataset and are different to template matching. The learning process is on the basis of common photometric features of human eye from a collective set of eye images with different head poses. The paper develops the last one-appearance-based method.

2 Cascade Classifiers Using Haar-Like Masks

Such a system was developed by [Viola and Jones 2001] as a face detector. The detector combines three techniques: the use of a comprehensive set of Haar-like *masks* (also called 'features' by Viola and Jones) that are in analogy to base functions of the Haar transform, the application of a boosted algorithm to select a set of masks for classifier training, and forming a cascade of strong classifiers by merging week classifiers. Haar-like masks are defined by adjacent dark and light rectangular regions; see Fig. 1.

Selection process of the object is based on the value distributions in dark or light regions of a mask that models expected intensity distributions. For example, the mask in Fig. 2, left, relates to the idea that in a face there are darker regions of eyes compared to the bridge of the nose. similarly, the mask in Fig. 2, right, models that the central part of an eye (the iris) is darker than the sclera area.

Computing Mask Values. Mean values in rectangular mask regions are calculated by applying the integral image as proposed in [Viola and Jones 2001]; see Fig. 3. For a given $M \times N$ picture P, at first the *integral image*

$$I(x,y) = \sum_{0 \le i \le x \wedge 0 \le j \le y} P(i,j) \tag{1}$$

is calculated. The sum $P(R_1)$ of all P-values in rectangle region R_1 (see Fig. 3) is then given by $I(D) + I(A) - I(B) - I(C)$. Analogously we calculate sums $P(R_2)$ and $P(R_3)$ from corner values in the integral image I. Values of contributing regions are weighted by reals ω_i that create *regional mask values* in form of

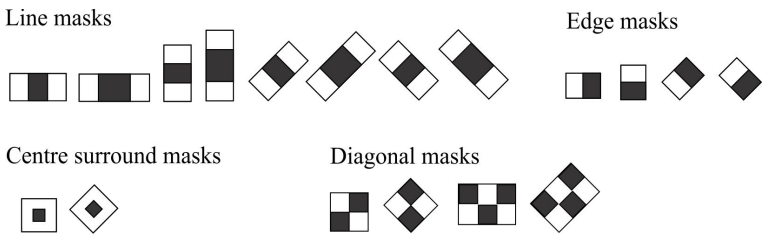

Fig. 1. Four different sets of masks for calculating Haar-like masks

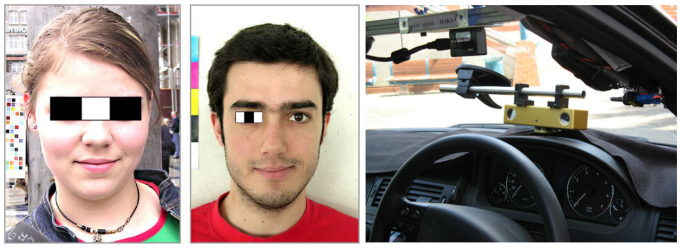

Fig. 2. *Left*: Application of two triple masks for collecting mean intensities in bright or dark regions. *Right*: Camera assembly in HAKA1 for driver distraction detection.

$v_i = \omega_i \cdot P(R_i)$, and then a *total mask value*; for the shown example this is $V_i = \omega_1 \cdot P(R_1) + \omega_2 \cdot P(R_2) + \omega_3 \cdot P(R_3)$. Signs of ω_i's are opposite for light and dark regions. In generalizing this approach, we also allow for arbitrary rotations. R_i is now defined by five parameters x, y, w, h, and φ, where x and y are coordinates of the lower-right corner, w and h are width and height, and φ is the rotation angle [Zhang and Zhang 2010]. For example, $P_\varphi(R_1) = I_\varphi(B) + I_\varphi(C) - I_\varphi(A) - I_\varphi(D)$ and for $\varphi = 45°$ we have

$$I_{45°}(x, y) = \sum_{|x-i| \le y-j \ \wedge \ 0 \le j \le y} P(i, j) \tag{2}$$

For any angle φ, the calculation of all $M \times N$ integral values I_φ takes time $\mathcal{O}(M \times N)$. This allows for real-time calculation of features on Haar-like masks.

Cascaded Classifiers via Boosted Learning. In a search window of 24×24 pixel there are more than 180,000 different rectangular masks of different shape, size, or rotation. However, only a small number of masks (usually less than 100) is sufficient to detect a desired object in an image (e.g. eye). In addition to defining regional mask weight w_i, using a boosting algorithm, the classifier can learn to sort out the prominent masks μ_i based on their overall wight W_i. Such wights determine the importance of each mask in an object detection process so we arrange all the masks in cascaded nodes as Fig. 4.

Each node (weak classifier) tries to determine whether the object (e.g. an eye) is inside the search window or not. The first classifier simply reject non-objects if

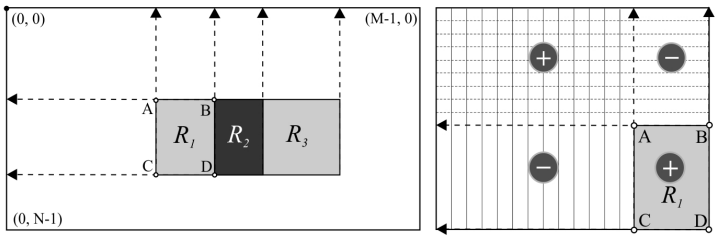

Fig. 3. Illustration for calculating a mask value using integral images. The coordinate origin is in the upper left corner.

the main masks (such as in Fig. 2) do not exist. If they exist then more detailed masks will be evaluated in next classifiers and the process continues. Actually each node represents a boosted classifier adjusted not to miss any object while it is rejecting non-objects if not matching the desired masks. Although each node is a weak classifier but all of them are considered a strong classifier and reaching the final node means that all non-objects have already been rejected and we have only one object (here: an eye). The function μ_i returns +1 if the mask value V_i is greater or equal to a trained threshold, and -1 if not:

$$\mu_i = \begin{cases} +1 & \text{if } V_i \geq T_i \\ -1 & \text{if } V_i < T_i \end{cases} \tag{3}$$

$\mu_i = +1$ means that the current weak classifier matches the object and we can proceed to the next classifier. Statistically about 75% of non-objects are rejected by the first two classifiers; the remaining 25% are for a more detailed analysis. This speeds up the process of object detection. In order to train the the algorithm we need a database of positive images (e.g. eyes)and on the first pass through the positive image database, we learn threshold T_1 for μ_1 such that it best classifies the input. Then boosting uses the resulting errors to calculate the overall weight W_1. Once the first node is trained then boosting continues for other nodes but with some other masks that are more sophisticated than previous ones [Freund et al. 1996].

Assume that each node (a weak classifier) is trained to correctly match and detect objects of interest with the true rate of $p = 99.9\%$ (true positive, TP). Since each stage alone is a weak classifier it is expected to be many false detections of non-objects, say $f = 50\%$ (false positive, FP)in each stage. This is still acceptable because, due to the serial nature of cascade classifiers, the overall detection ratios remains high (near 1) but it leads to a logarithmic decrease in the false positive rate (approaches to 0).

3 Scenarios and 3D Cascaded Classifiers

Most of eye detection algorithms such as [Wang et al. 2010] just look for the eyes in an already localized face. Therefore, eye detection simply fails if there is

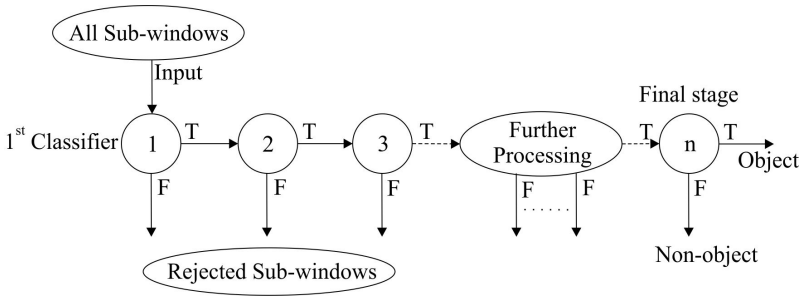

Fig. 4. Structure of cascaded classifiers for object detection

no full frontal view of a face, or some parts of face be occluded, or if parts of a face are outside of the camera viewing angle .

Our method follows a dynamic approaches, if the initial result of face detection is positive then we just look through the face region. Detection of an eye in a previously detected face region supports a double confirmation, and more confidence for the validity of eye detection. But if the face is not detected our 3D cascade looks for eye in the whole image. In our particular context we consider driver fatigue, drowsiness, distraction, or drunkenness when the driver misses to look forward on the road, or when the eyes are closed for some long uninterrupted period of time (say 1 sec. or more). As an example, when driving with a speed of 100 km/h, just one second eye closure means passing of 28 meters without paying attention. This can easily cause lane drift and a fatal crash. In our method we assume two status of *Looking Forward* and *Open Eyes* as important properties for judging driver's vigilance. for the face detection we follow the classifier in [Lienhart et al. 2003] for face detection and for the eye status detection we design our own classifiers. the proposed 3D designed classifier is able to detect and define 5 different scenarios while driving as below (see Fig. 5 from left to right):

Scenario 1: Obviously eyes are in the upper half of face region. By assessing 200 different faces from different races we derived that human eyes are geometrically located in segment A between 0.55 to 0.75 of the face's height. Applying this rough estimation in eye localization we already increased the search speed by factor 5 compared to a blind search, as we are only looking into 20% of the face's region. An eye pair is findable in segment A while the driver is looking forward.

Scenario 2: Some rare times happens that only one eye is detectable in segment A when the driver tilts his face. In that case we need to look for the second eye in segment B in the opposite half of the face region. Segment B is considered to be between 0.35 to 0.95 of the face's height; this covers more than ± 30 degrees of face tilt. The size of the search window in segment B is 30% of the face region. In that case of a tilted face we search both sections A and B (in total, 50% of the face's region). In Scenarios 1 and 2, the driver is looking forward to the roadway. So if we detect two open eyes then we decide that the driver is in the *Aware* state.

Scenario 3: If a frontal face is not detectable and just one of the eyes is detected, then this can be due to more than 45° of face rotation. The driver is looking

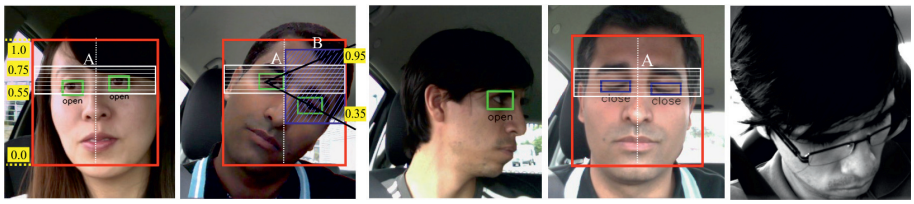

Fig. 5. Left to right: Scenarios 1 to 5 for driver's face and eye poses; see text for details

towards the right or left such that the second eye is occluded by the nose. The system immediately measures the period of time that the driver is looking to other sides instead of forward. This scenario also happens when the driver looks to side mirrors (but this takes normally less than second). Depending on the ego-vehicles speed, any occurrence of this scenario that takes more than 1 sec is considered as a sign of *Distraction* and the system will raise an alarm.

Scenario 4: Detection of closed eyes. Here we use an individual classifier for close eye detection. A closed-eye status happens frequently for normal eye blinking, and the eye *closure time* t_c is normally less than 0.3 sec. Any longer eye closures is a strong evidence of fatigue, drowsiness, or drunkenness. The system will raise an alarm for *Drowsiness* status if there is no open eye and at least one closed eye is detected.

Scenario 5: The worst case is when neither face, nor open eyes, nor closed eyes are detectable. This case occurs, for example, when the driver is looking over the shoulder, when the head falls in, or when the driver is performing secondary tasks. The system will raise an alarm for a detected *Risky Driving* status.

Considering all active detectors (face, open-eye, and close-eye detectors), we have cascaded classifiers in three dimensions that work in parallel. Implementing separate detectors for open and closed eye detection is important because at some times the open eye detector may fail to detect open eyes, but this does not necessarily mean that the eyes are closed. Missing eyes may be because of a specific head pose or bad lightening conditions. Having a separate closed-eye detector is a step toward high accuracy in driver distraction detection.

4 Training Image Database

The process of selecting positive and negative images is a very important step that affects the overall performance considerably. After several experiments it is determined that, although a larger number of positive and negative images can improve the detection performance in general, there is also an increase of the risk of mask mismatching during the training process. Thus, a careful consideration for number of positive and negative images and their content is essential. In addition, the multi-dimensionality of training parameters and the complexity of the feature space defines challenges. We propose optimized values of training parameters as well as unique features for our robust database.

In the initial negative image database, we removed all images that contained any objects similar to human eye (e.g. animal eyes). We prepared the training database by manually cropping closed or open eyes from positive images. Important questions needed to be answered: how to crop the eye regions and in what shapes (e.g. circular, isothetic rectangles, squares)? There is a general believe that circles or horizontal rectangles are best for fitting eye regions. However, we obtained the best experimental results by cropping eyes in square form. We fit the square enclosing full eye-width; for the vertical positioning we select balanced portions of skin area below and above the eye region. We cropped 12,000 eyes

from selected positive images of our own database plus six other databases: FERET database sponsored by the DOD Counterdrug Technology Development Program Office [Phillips et al. 1998, Phillips et al. 2000], Radbound face database [Langner et al. 2010], Yale facial database B [Lee et al. 2005], BioID database [Jesorsky et al. 2001], PICS database [PICS], and the "Face of Tomorrow" [FTD]. The positive database includes more than 40 different poses and emotions for different faces, eye types, ages, and races:

- Gender and age: females and males between 6 to 94 years old,
- Emotion: neutral, happy, sad, anger, contempt, disgusted, surprised, feared,
- Looking angle: frontal (0°), ±22.5°, and profile (±45.0°), and
- Race: East-Asians, Caucasians, dark-skinned people, and Latino-Americans.

The generated multifaceted database is unique, statistically robust and competitive compared to other training databases.

We also selected 7,000 negative images (non-eye and non-face images) including a combination of common objects in indoor or outdoor scenes. Considering a search window of 24×24 pixel, we had about 7,680,000 sub-windows in our negative database. An increasing number of positive images in the training process caused a higher rate for true positive cases (TP) which is good, and also increased false positive cases (FP) which is bad. Similarly, when the number of negative training images increased, it lead to a decrease in both FP and TP. Therefore we needed to consider a good trade-off for the ratio of number of negative sub-windows to the number of positive images. For eye classifiers, we got the highest TP and lowest rate for false negative detection when we arranged the ratio of $N_p/N_n = 1.2$ (this may vary for face detection).

5 AdaBoost Learning Parameters and Experiments

We implemented the training algorithm in OpenCV 2.1. With respect to our database we gained a maximum performance by applying the following settings: Size of mask-window: 21×21 pixel. Total number of classifiers (nodes): 15 stages; any smaller number of stages brought a lot of false positive detection, and a larger number of stages reduced the rate of true positive detection. The minimum of acceptable hit rate for each stage: 99.80% and increasing; a rate too close to 100% may cause the training process to take for ever or early failure. The maximum acceptable false alarm for the 1st stage: 40.0% per stage; this error goes to zero exponentially when the number of iterations increases. Weight trimming threshold: 0.95; this is the similarity weight to pass or fail an object in each stage. Boosting algorithm: among four types of boosting (Discrete AdaBoost, Real AdaBoost, Logit AdaBoost, and Gentle AdaBoost), we got about 5% more TP detection rate with Gentle AdaBoost. [Lienhart et al. 2003] also proved that GAB will result into lower FP ratios for face detection.

We performed a performance evaluation test on 2,000 images from the second part of the FERET database plus on 2,000 other image sequences recorded by HAKA1, our research vehicle (see Fig. 2, right). None of the test images were

Table 1. Classifiers accuracy (in %) in terms of true positive and false positive rate

Facial status	Open-eye detection		Closed-eye detection	
	TP	*FP*	*TP*	*FP*
Frontal face	98.6	0.0	97.7	0.20
Tilted face (up to $\pm 30^\circ$)	98.2	0.002	97.1	0.54
Rotated face (up to $\pm 45^\circ$)	96.8	0.0	96.8	0.7

included before in the training process and all the images are recorded in daylight condition. Table 1 shows the final results of open and closed eye detection rate.

6 Conclusions

With the aim of driver distraction detection, we implemented a robust 3D detector based on Haar-like masks and AdaBoost machine learning that is able to inspect for face pose, open eyes and closed eyes at the same time. Despite the similar research that are only able to work on frontal faces, The developed classifier is also able to works for tilted and rotated faces in real-time driving applications. There are no comprehensive data about performance evaluation for eye detection. Comparing results in [Kasinski and Schmidt 2010], [Niu et al. 2006], [Wang et al. 2010] and in [Wilson and Fernandez 2006] with our results (see Table 1), the method appears to be superior in a majority of cases. The method still needs improvement for dark environments. High-dynamic range cameras or some kind of preprocessing might be sufficient to obtain satisfactory detection accuracy also at night or in low-light environments.

References

[Langner et al. 2010] Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D.H.J., Hawk, S.T., Van Knippenberg, A.: Presentation and validation of the Radbound faces database. Cognition Emotion 24, 1377–1388 (2010)

[Freund et al. 1996] Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Machine Learning, pp. 148–156 (1996)

[Jesorsky et al. 2001] Jesorsky, O., Kirchberg, K., Frischholz, R.: Robust face detection using the Hausdorff distance. J. Audio Video-based Person Authentication, 900–995 (2001)

[Kasinski and Schmidt 2010] Kasinski, A., Schmidt, A.: The architecture and performance of the face and eyes detection system based on the Haar cascade classifiers. J. Pattern Analysis Applications 3, 197–211 (2010)

[FTD] Face of tomorrow database (2010), `http://www.faceoftomorrow.com/posters.asp`

[Lee et al. 2005] Lee, K.C., Ho, J., Kriegman, D.: Acquiring linear subspaces for face recognition under variable lighting. IEEE Trans. Pattern Analysis Machine Intelligence 27, 684–698 (2005)

[Lienhart et al. 2003] Lienhart, R., Kuranov, A., Pisarevsky, V.: Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 297–304. Springer, Heidelberg (2003)

[Niu et al. 2006] Niu, Z., Shan, S., Yan, S., Chen, X., Gao, W.: 2D cascaded AdaBoost for eye localization. In: ICPR, vol. 2, pp. 1216–1219 (2006)

[Phillips et al. 1998] Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.: The FERET database and evaluation procedure for face recognition algorithms. J. Image Vision Computing 16, 295–306 (1998)

[Phillips et al. 2000] Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET evaluation methodology for face recognition algorithms. IEEE Trans. Pattern Analysis Machine Intelligence 22, 1090–1104 (2000)

[PICS] PICS image database: University of Stirling, Psychology Department (2011), http://pics.psych.stir.ac.uk/

[Viola and Jones 2001] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR, pp. 511–518 (2001)

[Wang et al. 2010] Wang, H., Zhou, L.B., Ying, Y.: A novel approach for real time eye state detection in fatigue awareness system. IEEE Robotics Automation Mechatronics, 528–532 (2010)

[Wardlaw 2011] Wardlaw, C.: 2012 Mercedes-Benz C-Class preview (2011), http://www.vehix.com:80/articles/auto-previewstrends/2012-mercedes-benz-c-class-preview

[Wilson and Fernandez 2006] Wilson, P.I., Fernandez, J.: Facial feature detection using Haar classifiers. J. Computing Science 21, 127–133 (2006)

[Zhang and Zhang 2010] Zhang, C., Zhang, Z.: A survey of recent advances in face detection. MSR-TR-2010-66, Microsoft Research (2010)