A Learning-based Eye Detector Coupled with Eye Candidate Filtering and PCA Features

Bruno de Brito Leite,
Eanes Torres Pereira, and
Herman Martins Gomes
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
58109-970 Campina Grande, PB, Brazil
{brunob, eanes, hmg}@dsc.ufcg.edu.br

Luciana Ribeiro Veloso,
Cícero Einstein do Nascimento Santos, and
João Marques de Carvalho
Departamento de Engenharia Elétrica
Universidade Federal de Campina Grande
58109-970 Campina Grande, PB, Brazil
{veloso, cicero.einstein, carvalho}@dee.ufcg.edu.br

Abstract

In this work, we present a system based on a Neural Network classifier for eye detection in human face images. This classifer works on eye candidate regions extracted from a face image and represented by a reduced number of features, selected by Principal Component Analysis. The regions are determined considering that in an image window containing the eye, the grey level distribution will generally assume a pattern of adjacent light-dark-light horizontal and vertical stripes, corresponding to the eyelid, pupil and eyelid, respectively. For training, validation and testing, a database was built with a total of 4,400 images. Experimental results have shown that the proposed approach correctly detects more eyes than any of two existing systems (Rowley-Baluja-Kanade and Machine Perception Toolbox), for eye location error tolerances from 0 to 5 pixels. Considering an error tolerance of 9 *pixels, the correct detection rate achieved was above 90%.*

1 Introduction

Eye detection is a fundamental step of any automatic face analysis system. Not only analysis of the eyes (open, semi-open, closed, etc.) provides valuable information, but also the position of the eyes is generally used as reference to locate other significant face features, like eyebrow, mouth and nose, used for face identification and for facial expression recognition.

This paper is organized as follows. Next section presents a comprehensive literature review of recent work on passive eye detection, which are mostly related to the present work. Section 3 details the proposed approach. Section 4 presents the experimental evaluation and results. Finally, Section 5 draws some final considerations about the eye detection approach presented in this paper.

2 Related work

In a broad way, methods for eye detection fall into one of two categories or approaches: active and passive.

The active approach consists on focusing an infrared light beam into the eyes. The cornea ocular region reflects the beam back, producing what is known as "the red eye effect", very common in pictures taken using flash. This phenomenon produces high bright pupils in grayscale images, indicating the eyes position. The requirements on hardware (infrared light-emitting diode and a camera sensitive to this kind of illumination), scene environment (no infrared reflective surfaces) and on image sequences, make practically impossible the use of this approach on previously taken still images [17].

The passive approach can be further decomposed into three strategies: appearance based, model based and learning based. Since the passive approach is strongly related to the method proposed in this paper, the following subsections give more details about recent works using the passive approach for eye detection. Our choice for this approach is justified by the fact that we are interested in processing previously acquired still images or video frames, not considering the use of any active apparatus (such as infrared beam lighting) to facilitate eye location.

2.1 Appearance based approaches

In appearance based detection, features such as color, texture and shape can be analyzed. Through statistical modelling of human skin color distribution, clusters of skin regions can be specified in some color space and be used for image segmentation and connected component analysis. Some knowledge rules are applied on skin surrounded components to perform eye detection. Generally, appearance based detection requires few computational resources due to space search reduction and simple features, invariant to rotation and scaling, such as color and shape. The major drawbacks of this approach are related with the difficulty of obtaining a satisfactory skin region definition, covering wide ethnic variation and illumination changes. The following paragraphs present a review of works that uses the appearance based approach.

Welmin et al [7] propose a combined approach to solve the problem of robust face and eyes detection from images. They applied Gaussian and steerable filters to detect eye or eyebrow, mouth, and nose in the face image. After that, a structure model is created to group the features focusing points to face candidates using geometric relationships. In the detected face region, the eye/eyebrow area is determined using the face structure. Briefly, for open eyes, the local minimum of the image is taken as the eye center; for closed eyes, the black line or curve is examined and the middle point of the line is taken as the eye center.

To overcome limitations imposed by illumination and face pose on eye detection, Xingming and Huangyuan [22] propose an algorithm based on a new illumination normalization method and on Adaboost face detection. In the detected face, skin color probability is used to eliminate face shadows and the skin area is binarized. In order to remove non-eye candidates, basic rules, including arc length, width, length to width ratio, position, center of gravity, angle and distance, are applied to filter non-eye feature points. Finally, remaining eye-pairs are verified using SVM. The achieved detection rates are between 94.7% and 98.2%.

Fathi and Manzuri [4] detect skin using clusters defined on normalized RG color space. The RG color space reduces computational costs from that of a 3D to that of a 2D model, also reducing sensitivity to lighting changes. The search for eye candidates is done on regions delimited by detected skin and with low intensity pixels. Eye candidates located in the lower half area of the skin region and in a very little or a very large area are discarded, based on location and size. Remaining candidates are validated by an eye variance filter. In favorable experimental conditions, their system performance goes up to 98% of correct eye detection.

Han et al [5] do not detect skin. They perform a morphological closing operation on the grayscale original image. A valley map is detected subtracting the resulting image from the intensity input image. Cheeks, mouth and eyes are detected on the valley map using an iterative elliptical region growing algorithm. This algorithm compares dark or light pixels between elliptical regions with adjacent radius and segments the image in regions, according to pixels brightness. Combination of high bright circular areas and low bright elliptical areas satisfying a given set of conditions indicates cheeks, mouth and eyes location.

Kumar et al [11] propose a three step algorithm: (1) localization of possible eye areas with skin color model defined in HSV and RGB color spaces; (2) elimination of some candidates using spatial quantization with connected component analysis (aspect ratio above 0.75) and anthropometric knowledge rules (limiting angle and distance between candidates); and (3) the mean and the variance projection functions are employed to validate the presence of eyes.

An automatic face recognition (AFR) system for eye detection under variable lighting and reflection conditions is proposed by Samad et al [15]. Windows are constructed by calculating the average value of the grey level. Rules are applied on these windows, generating eye candidates. In the second step, a contour map is applied to the candidates, showing that the eyes are the regions with the highest number of surrounding contours. Thresholding contours with the highest value are used to indicate the location near the eyes. The final step is to detect the eyes by successively narrowing the set of candidates for eye locations.

A knowledge based eye detection algorithm was presented by Zhang and Lenders [23]. Initially face location is estimated using histogram thresholding. Segmentation is simplified by assuming only one face in the image. Eye regions are located using rules based on hair information. After selecting the eye regions, a knowledge-based edge detector is applied to locate iris and eyelids, thus determining the four corners of a rectangle surrounding the eye. The Yale Face Database was used to test the system.

Zhou and Geng [24] defined a generalized projection function (GPF) which combines the Integral Projection Function (IPF) and the Variance Projection Function (VPF). Another special case of GPF, i.e. the Hybrid Projection Function (HPF), which inherits both the robustness of IPF and the sensitiveness of VPF, is empirically developed. Experiments on face databases show that all the special cases of GPF are effective in eye detection. Moreover, it is found that IPF is more effective on the occidental face databases than on the oriental face databases, and VPF is more effective on the oriental face databases than on the occidental face database. Analysis of the detections reveals that this may be due to the differences in the shadows caused by noses and eyeholes of different ethnies.

Table 1 presents a summary of features used in the appearance based papers described. The first column presents the authors of papers and second column, the strategy applied to extract the appearance based features used to find eyes.

Author	Strategy
Welmin et al [7]	Steerable filter
Xingming and Huangyuan [22]	Mathematical morphology
Fathi and Manzuri [4]	Skin filter and intensity
Han et al [5]	Mathematical morphology
Kumar et al [11]	Skin filter and projection functions
Samad et al [15]	Contour map
Zhang and Lenders [23]	Binary image
Zhou and Geng [24]	Projection functions

Table 1. Summary of appearance-based approaches

2.2 Model based approaches

Model based detection seeks to minimize an error energy function, obtained by applying template matching on the image. The template is normally composed by a circle and two parabolic curves, thus modeling the eye region. Methods that use this approach basically differ on the way of finding the initial template placement. On searching for the eyes, the model is deformed according to the energy function minimization. This allows for good detection even with rotated eyes and is tolerant to eye aperture variations. Some disadvantages of using this approach are the difficulty of finding the points for model placement and the high computational cost. Two reviews of model based approaches are presented below.

In the algorithm proposed by Lin and Yang [12], a Dark-Pixel Filter (DPF) is applied on a face region obtained by a Face-Circle Fitting (FCF) method. To detect human face in a color image, a Hue Saturation intensity (HSI) color model was adopted. However, the noise introduced by skin-color pixel extraction procedure may form connected regions. To overcome this drawback, the FCF method was proposed. By the DPF algorithm, dark pixel fitting, eye verification, and eye location are performed.

Shinjiro and Kawato [10] have proposed a new algorithm to extract and track eyes position in real-time video streams. Positions are identified through the eye blinks located from the differences between successive frames in the video stream. An updating template based on the between-the-eyes pattern is used to track the eye positions.

The summary of templates used in the model based papers described in this work is presented on Table 2. The authors of papers are on the first column and the template used on the second column.

Author	Template
Lin and Yang [12]	Face circle and dark pixel
Shinjiro and Kawato [10]	Between-the-eyes

Table 2. Summary of model based approaches

2.3 Learning based approaches

For learning based detection, a set of eye image examples and counter-examples are used for training a given classifier. Systems that use this approach usually differ mainly on the features used for training and on the techniques employed for extracting those features from the images. Classification can be performed either on a multiscale image or on normalized detected face regions. Eyes are detected on the regions that produce the classifier's highest response. This approach presents training invariance to rotation and to changes of both scale and illumination. No specific criteria for counter-examples selection in generic images are usually provided. Learning based eye detection is the approach we adopted in this work.

Wu and Trivedi [21] propose a framework to solve the eye detection and localization problem for face images, using a binary tree representation and statistical structures to find objects in images. Independent Component Analysis (ICA) is performed on eye image samples and utilized to extract patterns to construct the tree which separates the conditionally independent features into different sub-trees. A k-means algorithm is used to group similar patterns, traversing the tree in a bottom-up procedure. Experimental evaluations show that correct detection reaches a rate of 92.43%, while the false detection rate is 7.26%.

A robust eye detection system has been proposed by D'Orazio et al. [2, 1]. The algorithm works on complex images without constraints on background or on skin color segmentation and no limitations imposed on the eye regions. A circle detector based on the directional Circle Hough Transform is applied on the whole image, looking for regions limited by borders with a circular configuration, similar to that normally presented by the iris. Sub-images containing the result of the detection process are input to a neural network trained to classify images as either eye or non-eye instances. Experiments have shown that this classifier reaches true positive rates between 89% and 96%.

Based on the distribution of discriminant features, Wang et al [20] propose training probabilistic classifiers and combining multiple classifiers using AdaBoost, to form a robust and accurate eye detector. The eye localization method follows a hierarchical procedure: first a face is detected, the face detection method being also based on Adaboost; second, geometric constraints are applied to localize eyes, restricting eyes to be only searched for in the face upper half. To assure robustness, the authors have collected training data from various sources. In practice, only a left eye detector is trained, to profit from eyes mirror-image symmetry. Therefore, images have to be flipped over in order to detect the right eye. The non-eye images were randomly collected from background images. To improve eye detection speed, a cascade structure is utilized, validation results showing that an overall 94.5% eye detection rate is obtained.

Another eye detection approach is presented by Lizuo et al [8]. The low dimensional features representing eye patterns are produced by projecting normalized eye images onto a weighted eigenspace and selected via a filter and a wrapper method for training a simplified maximum likelihood (ML) and a SVM classifiers. To reduce false positive rates, a spatial configuration of eye-pair to reject false blobs yielded by the above classifiers is applied. The performance of this eye detector is assessed on several publicly available face databases (Feret, BioID, ORL and Yale).

Wang and Yin [19] derive a terrain map from the original gray scale image using topographic primal sketch analysis. Pit-pixels in the terrain map are marked as candidates for eye pair selection and a Gaussian Mixture Model(GMM) is used as classifier to select the correct eye pair among the candidates. In the experiments d the detection approach was tested on the Japanese Female Facial Expression (JAFFE) database. From 213 facial images, 204 were correctly detected, which means 95.8% of correct detection rate.

Peng et al [13] proposed a hybrid neural method for human eyes detection. Image preprocessing includes two phases: resolution reduction and gray level normalization. According to the authors, a Radial Basis Function (RBF) neural network was used due to its fast training, good-generality, and simplicity as compared to the multi-layer perceptron (MLP). The RBF network serves as a filter between the input facial image and the map image where the intensity peaks are referred to as possible eye regions. Knowledge-based rules are used to validate the eye candidate regions. In the experiments reported an image database consisting of more than 300 facial images captured from 18 Chinese students was used, producing 97.41% of accuracy.

The learning based papers described in this work are summarized on Table 3. The first column contains the authors of papers, the second shows the features used as input to the classification process and the third, the type of classifier employed.

2.4 Databases and performance

When considering performances of learning-based classifiers, it is important to analyze the database used for training and evaluation, as it may strongly influence the results obtained. However, many of the databases utilized in the papers reviewed are not publicly available, which makes analysis impossible. Others, like JAFFE and Peng‘s are strongly polarized towards one specific ethny or gender. In some cases, the procedure for obtaining the base is quite complicated and time consuming, discouraging its use.

The databases and performance found in the reviewed work are summarized on Table 4. The first column enumerates the authors, while test databases and performances are showed in second and third columns, respectively.

Author	**Strategy**	**Classifier**
Wu and Trivedi [21]	Independent component analysis	Bayesian
D'Orazio et al [2, 1]	Wavelet transform	Neural network
Wang et al [20]	Recursive non-parametric discriminant feature	AdaBoost
Lizuo et al [8]	Normalized image projection onto weighted eigenspace	Support vector machine
Wang and Yin [19]	Terrain Features	Gaussian mixture model
Peng et al [13]	Raw image	Radial basis neural network

Table 3. Summary of learning approaches

Author	**Test database**	**Performance**
Welmin [7]	Proprietary	97.9%
Xingming and Huangyuan [22]	SCUT	96.2%
Fathi and Manzuri [4]	Proprietary	98%
Han et al [5]	AR	98.8%
Kumar et al [11]	Not informed	Not informed
Samad et al [15]	Olivetti, Yale	94%
Zhang and Lenders [23]	Yale	Not informed
Zhou and Geng [24]	BioID, JAFFE and NJUFace	98.4%
Lin and Yang [12]	HHI	74.0%
Shinjiro and Kawato [10]	Proprietary	Not informed
Wu and Trivedi [21]	Feret	92.43%
D'Orazio et al [2, 1]	Proprietary	89.0%
Wang et al [20]	FRGC 1.0	94.5%
Lizuo et al [8]	Feret	99.76%
Wang and Yin [19]	JAFFE	95.8%
Peng et al [13]	Proprietary	97.41%

Table 4. Databases and performance

3 Proposed approach

Our approach, described in this section falls into the learning-based category. The block diagram of the eye detection system is presented in Figure 1.

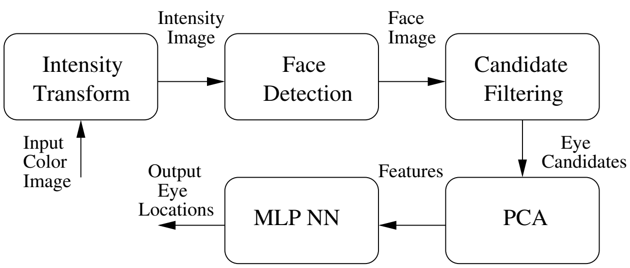

Figure 1. Block diagram of the proposed eye detection approach

Initially, the original three channel (RGB) color image is converted to single channel using the intensity transformation function given by:

$$\begin{aligned} I(x,y) &= 0.299 \times R(x,y) \\ &\quad + 0.587 \times G(x,y) \\ &\quad + 0.114 \times B(x,y), \end{aligned} \tag{1}$$

where $R(x,y)$, $G(x,y)$ and $B(x,y)$ are the red, green and blue channel values, respectively, and $I(x,y)$ is the intensity of the transformed image.

Next, the intensity image $I(x,y)$ is submitted to face detection. The face detector (provided by Hewlett-Packard) returns a list of coordinates with the upper left and bottom right corners of delimiting rectangles for the detected faces. Using these coordinates, face images are cropped and normalized to 100×100 pixels size by a linear interpolation algorithm.

Inspired by the work of Viola and Jones [18], eye candidates are filtered using an intermediate representation of the normalized image, called integral image, defined as:

$$\begin{aligned} IN(x,y) &= I(x,y) \\ &\quad + IN(x-1,y) \\ &\quad + IN(x,y-1) \\ &\quad - IN(x-1,y-1), \end{aligned} \tag{2}$$

where $I(x,y)$ is the intensity image and $IN(x-1,y)$, $IN(x,y-1)$ and $IN(x-1,y-1)$ are recursive calls of the integral image function.

Integral image representation allows fast mean intensity calculation for a region starting at x, y coordinates with width w and height h, as given by:

$$\begin{aligned} M(x,y,w,h) &= IN(x+w,y+h) \\ &\quad - IN(x+w,y) \\ &\quad - IN(x,y+h) \\ &\quad + IN(x,y). \end{aligned} \tag{3}$$

To search for eye candidates, a 24×15 pixels grid was empirically defined and organized as a matrix of 9 cells, each of size 8 $(w) \times 5$ (h), as illustrated in Figure 2.

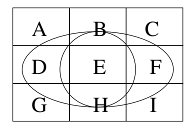

Figure 2. Integral image grid

The grid slides over the integral image, allowing cell mean value calculation at each position. We search for regions satisfying location and intensity restrictions. Location restriction is given by:

$$(20 < distance(x_g, y_g, x_f, y_f) < 45) \wedge y_g < 50, \tag{4}$$

where $distance(x_f, y_f, x_f, y_f)$ is the Euclidean distance from the grid center (x_g, y_g) to the center of the cropped face image (x_f, y_f), and 50 is half of the height of the cropped face image (so that the search is constrained to the upper half of the face). The values 20 and 45 were experimentally defined by measuring the distance from the eyes's center to the center of normalized detected face images.

Intensity restrictions are defined as:

$$\begin{aligned} (M(ABC) &> M(DEF)) \\ \wedge\, (M(GHI) &> M(DEF)) \\ \wedge\, ((M(E) < M(D) &\vee M(E) < M(F)) \\ \wedge\, ((M(E) < M(G) &\vee M(E) < M(I)) \\ \wedge\, (M(E) &< M(H)), \end{aligned} \tag{5}$$

where A,...,I represent grid cells and $M(X)$ is the mean intensity value of cell (or group of cells) X calculated using the integral image. The application of this restriction allows reduction of search space to light regions with a black area at the center (appearance of an eye). The central coordinates of regions satisfying the conditions above define eye candidates.

The 360 (24×15) features (pixels) from eye candidate regions are reduced to 98 features using Principal Component Analysis (PCA). PCA [6] is a useful statistical technique for finding patterns in high dimensional data. Essentially, a set of correlated variables is transformed into a set of uncorrelated variables (by linear combination) and ordered by decreasing variability. The main goal of PCA is to reduce the dimensionality of a data set while retaining as much information as possible. The resulting set of PCA components account for 90% of the total variance.

The output of PCA is applied to the input of a Multilayer Perceptron Neural Network (MLP NN) [6], for eye detection. The determination of the number of components of PCA and the neural network training were performed using a database formed by images acquired from the BioID database (http://www.bioid.com/downloads/facedb/) and collected from the web. The procedure for building the eye and non-eye image database, for each image, followed the steps: (1) faces were detected, cropped and normalized; (2) eye candidate filtering was applied on each detected face; (3) candidates satisfying the filtering conditions were stored, others were discarded; and (4) stored candidates were labeled by humans as eyes (only pupil centered eye images were considered) and non-eyes.

At the end of this process, eyes were extracted from BioID (85%) , and from web crawled images (15%). All non-eye images were extracted from web crawled images. Table 5 contains the partition of the image database into training, validation and test sets, used for determination of the number of components of PCA and training the MLP Neural Network (NN). The first column indicates the phase of the training process, the second column shows the quantity of images used (of which 50% are eyes and 50% are non-eyes in all cases), and the third column gives the relative partition size.

Step	**Images**	**Partition**
Training	2,640	60%
Validation	660	15%
Test	1,100	25%
Total	4,400	100%

Table 5. Image database partition

The architecture of MLP NN used has a topology composed of 98 neurons (same size of PCA output) in the input layer, 50 neurons in the hidden layer and 2 neurons in output layer (one for eye and other for non-eye). Since training success may rely on the weights' initialization, ten networks (randomly initialized) were trained with backpropagation momentum learning function algorithm by 1000 cycles. The network with the best least Sum Square Error (SSE) was integrated in the eye detection system. The precise eye location is defined applying the output of neural network in the following equation:

$$response = \begin{cases} 0.5 \times eye \\ \quad + 0.5 \times \Delta eye, & \text{if } \Delta eye > 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $\Delta eye = (eye - noneye)$, eye and $noneye$ are the values of the two output layer neurons, and $response$ is the outcome of the eye classification. The center of an eye candidate with maximum value in Equation 6 is chosen as the center of eye for each halves of the face.

4 Experiments and results

A comparative experiment with two other eye detectors was performed. This experiment used images from the Cohn-Kanade AU-Coded Facial Expression Database [9] (available at http://vasc.ri.cmu.edu/idb/html/face/facialexpression/).

This database is composed by video frame sequences of 9 distinct subjects, representing a broad range of ethnic, age and gender groups. In each sequence, the subject changes from a neutral face to one of seven facial expressions (anger, surprise, disgust etc), coded according to the Action Units (AU) system. These attributes make the Cohn-Kanade database very attractive to use, besides the fact it can be easily obtained upon request to the owners.

A subset of 907 face images of 97 different subjects was randomly selected from this database and the positions of the eyes in those images were manually marked. Next, the images were fed into three learning based eye detectors: an implementation of our proposed approach, an eye detector from the Rowley-Baluja-Kanade Face (and eye) detection project [14] (available at http://vasc.ri.cmu.edu/NNFaceDetector/) and another eye detector from the Machine Perception Toolbox [3] (available at http://mplab.ucsd.edu). The results produced by the three methods were evaluated, relative to the manual markings (or ground truth).

The main reason for choosing these two methods was the fact that they are freely available in executable code, ready to use. Using the techniques reviewed in Section 2 would require implementation which usually is a very time consuming (and sometimes impossible) task, due to the usual lack of detailed algorithm description.

The histograms of errors presented by all eye detectors revealed a Gaussian shape-like distribution. Table 6 shows accuracy statistics yielded by each detector using images from the Cohn-Kanade AU-Coded Facial Expression Database. The first column indicates the eye detector. The second and third columns show the mean error (μ_{error}) with regards to the correct eye location and the standard error deviation (σ_{error}), respectively, both measured in pixels.

Detector	μ_{error}	σ_{error}
Proposed approach	4.42	5.96
Rowley-Baluja-Kanade	3.68	2.14
Machine Perception Toolbox	7.42	6.66

Table 6. Error Statistics for the Cohn-Kanade Database

A second similar experiment was done to further evaluate the performance of the three eye detectors, using the the IMM Face Database [16], which is composed by 240 annotated images of 40 different human faces. The results obtained are shown in Table 7. Although not as large as varied, the IMM has the advantage of being dissociated from the detectors tested, contrasting to the Cohn-Kanade database which was developed by the same group as the Rowley-Baluja-Kanade detector. Independence between training and testing databases is crucial when evaluation learning based detectors, which motivated the use of the IMM in our experiments, besides the fact that it can be easily obtained at http://www2.imm.dtu.dk/ aam/datasets/datasets.html. It can be seen that the Rowley-Baluja-Kanade detector [14] presented the smallest mean error and standard deviation, followed by the proposed approach, in the two experiments.

Detector	μ_{error}	σ_{error}
Proposed approach	7.57	4.81
Rowley-Baluja-Kanade	6.46	2.48
Machine Perception Toolbox	8.96	8.61

Table 7. Error Statistics for the IMM Database

Tables 7 and 6 show that, overall, the Rowley-Baluja-Kanade approach is more accurate than the proposed approach. However, a third experiment, shown next, demonstrates that under certain conditions our approach performs best. The chart in Figure 3 summarizes this experiment, in which eye detection rates (y axis) were obtained by varying different error tolerances (x axis), expressed in pixels, between the detected eye and the manual eye markings (ground truth). Considering an error tolerance of only 9 pixels for the center of the detected eyes relative to the manually marked center of the pupils, the correct detection rates achieved by both the proposed approach and the Rowley-Baluja-Kanade Face (and eye) Detection approach were above 90%, whereas the MPT approach only reached a bit over 70%. The chart also shows that our approach correctly detects more eyes than any of the other methods, for error tolerances of up to 5 pixels. The Rowley-Baluja-Kanade eye detector only becomes better than our approach for values of error tolerance higher than 5 pixels. The Machine Perception Toolbox presents the worst performance of the three methods, for all tolerance values. Performances converge to about 99% of correct detection when error tolerance approaches 20 pixels, for all three detectors. Our approach takes a cropped face image as input, thus two eye coordinates are allways produced as output. Since all images in the test database contain two eyes, misdetecting an eye will necessarily imply in misclassifying a non-eye as eye, therefore the false acceptance and false rejection errors for our approach were equally balanced in the experiments.

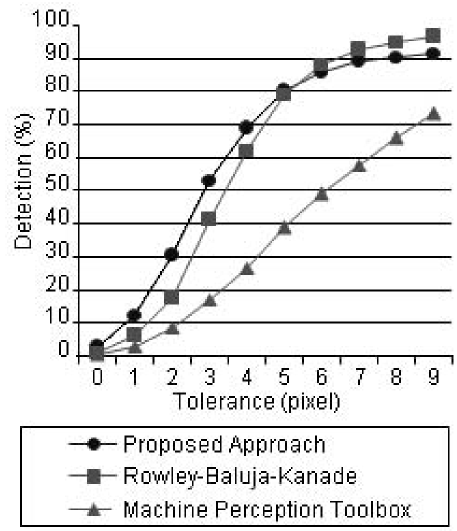

Figure 3. Comparative performance chart

Figure 4 reinforces the accuracy of the proposed approach by illustrating the detected eye positions in test images from the Cohn-Kanade and IMM databases, and comparing them with the Rowley-Baluja-Kanade (RBK) outputs.

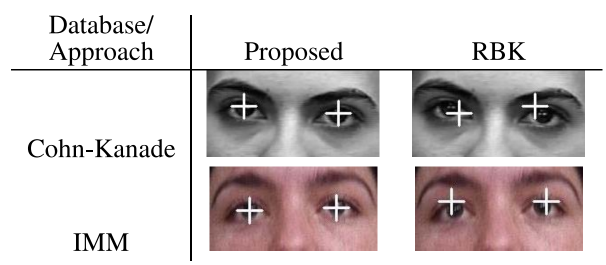

Figure 4. Detected eyes from proposed and Rowley-Baluja-Kanade approaches using images from Cohn-Kanade and IMM databases.

5 Final considerations

Correct recognition and location of eyes in face images is essential for recognition and location of other face features, as well as for recognition of facial expressions, to mention only a few from a vast set of potential applications.

Several systems have been proposed to perform eye detection, as described in Section 2, utilizing distinct types of classifier, training and testing methodology. In this work we described an approach composed by eye filtering (integral image rules) and classification (neural network) modules. The use of integral image in eye filtering allows fast search space reduction, while application of PCA improves the correlation between the features used by the classifier and those on the classification problem. The obtained results were encouraging when compared with those produced by two other well known eye detectors. A point for future investigation is the use of appearance models to help with eye location and to determine the state of the eyes.

Many of the reviewed papers report only detection rates when evaluating performance, not (in general) describing tolerance errors (regarding eyes' correct location) used to generate those ratios. This fact, along with the differences in databases, invalidates a direct comparison of detection rates between the methods in Table 4 and the approaches we tested. To consider the error tolerance is essential for performance evaluation in order to obtain realistic results. In most practical face analysis problems the eyes coordinates are utilized to locate other facial features (e.g. nose and mouth). If a large error tolerance is assumed, very high detection rates can usually be obtained. However, the actual performance in practical situations can be disappointing.

Acknowledgment

This work was developed in collaboration with HP Brazil R&D.

References

[1] T. D'Orazio, M. Leo, G. Cicirelli, and A. Distante. An algorithm for real time eye detection in face images. In *Int. Conf. on Pattern Recognition*, volume 3, pages 278–281, 2004.

[2] T. D'Orazio, M. Leo, P. Spagnolo, and C. Guaragnella. A neural system for eye detection in a driver vigilance application. In *IEEE Int. Conf. on Intelligent Transportation Systems*, pages 320–325, 2004.

[3] I. Fasel, B. Fortenberry, and J. R. Movellan. A generative framework for real time object detection and classification. *Computer Vision and Image Understanding*, 98:182–210, 2005.

[4] A. Fathi and M. T. Manzuri. Eye detection and tracking in video streams. In *IEEE Int. Symp. on Comm. and Information Technology*, volume 2, pages 1258–1261, 2004.

[5] H. Han, T. Kawaguchi, and R. Nagata. Eye detection based on grayscale morphology. In *IEEE Region 10 Conf. on Computers, Comm., Control and Power Engineering*, volume 1, pages 498–502, 2002.

[6] S. Haykin. *Neural Networks: A Comprehensive Foundation, 2nd Edition*. Prentice Hall, 1998. BAL h 01:1 1.Ex.

[7] W. Huang, Q. Sun, C.-P. Lam, and J.-K. Wu. A robust approach to face and eyes detection from images with cluttered background. In *Int. Conf. on Pattern Recognition*, volume 1, pages 110–113, 1998.

[8] L. Jin, X. Yuan, S. Satoh, J. Li, and L. Xia. A hybrid classifier for precise and robust eye detection. In *Int. Conf. on Pattern Recognition*, volume 4, pages 731–735, 2006.

[9] T. Kanade, J. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *Int. Conf. on Automatic Face and Gesture Recognition*, pages 46–53, 2000.

[10] S. Kawato and N. Tetsutani. Detection and tracking of eyes for gaze-camera control. *Image and Vision Computing*, 22:1031–1038, 2004.

[11] R. T. Kumar, S. K. Raja, and A. G. Ramakrishnan. Eye detection using color cues and projection functions. In *Int. Conf. on Image Processing*, volume 3, pages 337–340, 2002.

[12] D.-T. Lin and C.-M. Yang. Real-time eye detection using face-circle fitting and dark-pixel filtering. In *IEEE Int. Conf. on Multimedia and Expo*, volume 2, pages 1167–1170, 2004.

[13] H. Peng, C. Zhang, and Z. Bian. Human eyes detection using hybrid neural method. In *Int. Conf. on Signal Proc.*, volume 2, pages 1088–1091, 1998.

[14] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE TPAMI*, 20(1):23–38, 1998.

[15] S. A. Samad, A. Hussain, and A. Teoh. Eye detection using hybrid rule based approach and contour mapping. In *Int. Symp. on Signal Processing and its Applications*, volume 2, pages 631–634, 2001.

[16] M. B. Stegmann, B. K. Ersbøll, and R. Larsen. FAME – a flexible appearance modeling environment. *IEEE Trans. on Medical Imaging*, 22(10):1319–1331, 2003.

[17] F. H. C. Tivive and A. Bouzerdoum. A fast neural-based eye detection system. In *Int. Symp. on Intelligent Signal Proc. and Comm. Systems*, pages 641–644, 2005.

[18] P. Viola and M. J. Jones. Robust real-time face detection. *Int. J. of Computer Vision*, 57:137–154, 2004.

[19] J. Wang and L. Yin. Eye detection under unconstrained background by the terrain feature. In *IEEE Int. Conf. on Multimedia and Expo*, pages 1528–1531, 2005.

[20] P. Wang, M. B. Green, Q. Ji, and J. Wayman. Automatic eye detection and its validation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, volume 3, pages 164–164, 2005.

[21] J. Wu and M. M. Trivedi. A binary tree for probability learning in eye detection. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, volume 3, pages 170–170, 2005.

[22] Z. Xingming and Z. Huangyuan. An illumination independent eye detection algorithm. In *Int. Conf. on Pattern Recognition*, volume 1, pages 392–395, 2006.

[23] L. Zhang and P. Lenders. Knowledge-based eye detection for human face recognition. In *Int. Conf. on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, volume 1, pages 117–120, 2000.

[24] Z. Zhi-Hua and X. Geng. Projection functions for eye detection. *Pattern Recognition*, 37(5):1049–1056, 2004.