2005 IEEE International Conference on Systems, Man and Cybernetics
Waikoloa, Hawaii October 10-12, 2005

Fully Automatic Facial Feature Point Detection Using Gabor Feature Based Boosted Classifiers

Danijela Vukadinovic and Maja Pantic
Delft University of Technology
Electrical Engineering, Mathematics and Computer Science
Delft, The Netherlands
{D.Vukadinovic,M.Pantic}@ewi.tudelft.nl

Abstract – *Locating facial feature points in images of faces is an important stage for numerous facial image interpretation tasks. In this paper we present a method for fully automatic detection of 20 facial feature points in images of expressionless faces using Gabor feature based boosted classifiers. The method adopts fast and robust face detection algorithm, which represents an adapted version of the original Viola-Jones face detector. The detected face region is then divided into 20 relevant regions of interest, each of which is examined further to predict the location of the facial feature points. The proposed facial feature point detection method uses individual feature patch templates to detect points in the relevant region of interest. These feature models are GentleBoost templates built from both gray level intensities and Gabor wavelet features. When tested on the Cohn-Kanade database, the method has achieved average recognition rates of 93%.*

1 Introduction

Facial feature points are generally referred to as facial salient points such as the corners of the eyes, corners of the eyebrows, corners and outer mid points of the lips, corners of the nostrils, tip of the nose, and the tip of the chin (see Fig. 1(e)). Detection of facial feature points is often the first step in computer vision applications such as face identification, facial expression recognition, face tracking and lip reading. For example, localization of facial points is the initial step of Active Shape and Active Appearance Models algorithms (e.g. [1]) that are nowadays widely used for face alignment and tracking. Currently, however, this step is usually carried out by manually labeling the required set of points. The localization of stable facial points such as the inner corners of the eyes and the inner corners of the nostrils is also usually used to register each frame of an input image sequence with the first frame of it. In turn, the robustness of the facial feature point detection algorithm highly affects the overall system performance.

Previous methods for facial feature point detection could be classified in two categories: texture-based and shape-based methods. Texture-based methods model local texture around a given feature point, for example the pixel values in a small region around a mouth corner. Shape-based methods regard all facial feature points as a shape, which is learned from a set of labeled faces, and try to find the proper shape for any unknown face. Typical texture-based methods include gray-value-, eye-configuration- and neural-network-based eye-feature detection [2], log Gabor wavelet based facial point detection [3], and two-stage facial point detection using a hierarchy of Gabor wavelet networks [4]. Typical shape-based methods include active appearance model based facial feature detectors [5], [6]. A number of approaches combining texture- and shape-based methods have been proposed as well. Wiskott et al. [7] used Gabor jet detectors and modeled the distribution of facial features with a graph structure. Cristinacce and Cootes used Haar feature based AdaBoost classifier combined with the statistical shape models [8]. Chen et al. proposed a method that applies a boosting algorithm to determine facial feature point candidates for each pixel in an input image and then uses a shape model as a filter to select the most possible position of feature points [9]. In general, although some of these detectors seem to perform quite well when localizing a small number of facial feature points such as the corners of the eyes and the mouth, none of them detects all 20 facial feature points illustrated in Fig. 1(e) and, more importantly, none performs the detection with high accuracy. To wit, the current approaches usually regard as SUCCESS if the bias of automatic labeling result to the manual labeling result is less than 30% of the true (annotated manually) inter-ocular distance (the distance between the eyes). However, 30% of the true inter-ocular value is at least 30 pixels in the case of the Cohn-Kanade database samples [10], which we used to test our method. This means that a bias of 30 pixels for an eye corner would be regarded as SUCCESS even though the width of the whole eye is approximately 50 pixels. This is unacceptable in the case of facial expression analysis, which represents the main focus of our research, since subtle changes in the facial feature appearance will be missed due to the errors in point localization and tracking [11].

We propose here a robust, highly accurate method for detecting 20 facial points in images of expressionless faces with possible in-plane rigid head rotations, recorded under various illumination conditions. The method consists of 4 steps (Fig. 1): Face Detection, Region Of Interest (ROI) Detection, Feature Extraction, and Feature Classification. To detect the face region in an input image, we adopted fast and robust face detector based on a cascade scheme consisting of a set of Haar feature based GentleBoost

classifiers [12]. The detected face region is then divided in 20 relevant ROIs, each one corresponding to one facial point to be detected. A combination of heuristic techniques based upon the analysis of the vertical and horizontal image histograms achieves this. The proposed facial feature point detection method uses individual feature patch templates to detect points in the relevant ROI. These feature models are 13×13 pixels GentleBoost templates built from both gray level intensities and Gabor wavelet features. In the training phase, the feature models are learned using a representative set of positive and negative examples, where the positive examples are image patches centered on a particular facial feature point and the negative examples are image patches randomly displaced a small distance from the same facial feature. In the testing phase, each ROI is filtered first by the same set of Gabor filters used in the training phase (in total, 48 Gabor filters are used). Then, for a certain facial point, 13×13 pixels window (*sliding window*) is slid pixel by pixel across 49 representations of the relevant ROI (grayscale plus 48 Gabor filter representations; see Fig. 1(c)). For each position of the sliding window, GentleBoost classifier outputs a response depicting the similarity between the 49-dimensional representation of the sliding window compared to the learned feature point model. After scanning the entire ROI, the position with the highest response reveals the feature point in question.

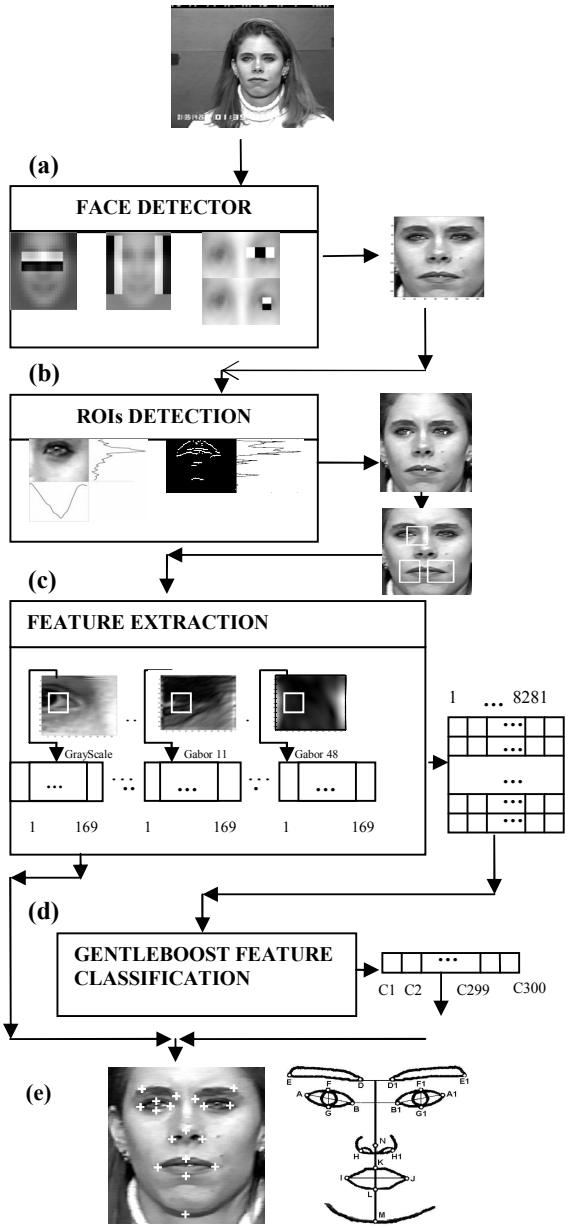


Fig. 1. Outline of the method. (a) Face detection using Haar feature based GentleBoost classifier; (b) ROI extraction, (c) feature extraction based on Gabor filtering, (d) feature selection and classification using GentleBoost classifier, (e) output of the system compared to the face drawing with facial landmark points we aim to detect.

The remainder of this paper is organized as follows. In Section 2 we describe the four steps of our method. Section 3 describes the experimental results achieved by the proposed method. Section 4 concludes the paper.

2 Methodology

2.1 Face Detection

To build a system capable of automatically labeling facial feature points in a face image, it is first necessary to localize the face in the image. We make use of a real-time face detection scheme proposed in [12], which represents an adapted version of the original Viola-Jones face detector [13]. The Viola-Jones face detector consists of a cascade of classifiers trained by AdaBoost. Each classifier employs integral image filters, which remind of Haar Basis functions and can be computed very fast at any location and scale (Fig. 1(a)). This is essential to the speed of the detector. For each stage in the cascade, a subset of features is chosen using a feature selection procedure based on AdaBoost.

The adapted version of the Viola-Jones face detector that we employ uses GentleBoost instead of AdaBoost. It also refines the originally proposed feature selection by finding the best performing single-feature classifier from a new set of filters generated by shifting and scaling the chosen filter by two pixels in each direction, as well as composite filters made by reflecting each shifted and scaled feature horizontally about the center and superimposing it on the original. Finally the employed version of the face detector uses a smart training procedure in which, after each single feature, the system can decide whether to test another feature or to make a decision. By this the system retains information about the continuous outputs of each feature detector rather than converting to binary decisions at each stage of the cascade. The employed face detector was trained on 5000 faces and millions of non-face patches from about 8000 images collected from the web by Compaq Research Laboratories [12]. On the test set of 422 images from the Cohn-Kanade database [10], the detection rate was 100%.

2.2 Detecting Regions of Interest

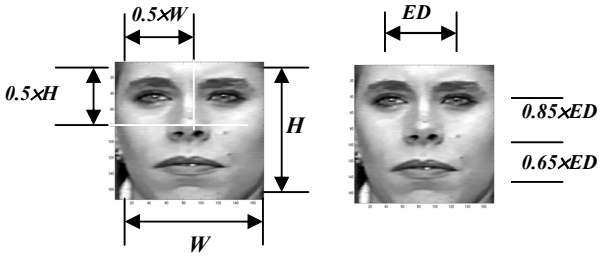
The next step in the automatic facial point detection is to determine Region Of Interest (ROI) for each point, that is, to define more or less a large region which contains the point that we want to detect. To achieve this we apply a fully automated method for detecting the irises and the medial point of the mouth. When those three positions are known we can easily determine other ROIs within the face region.

The iris and mouth detection is achieved as follows. First, we divide the face region horizontally into two parts: the upper face region containing the eyes and the lower face region containing mouth (Fig. 2). Since the face detector described above is highly accurate and the detected face region is always extracted in the same way regarding the relative size and position of the face box, it is sufficient, for the first step, to roughly divide the face region horizontally in two halves (Fig. 2(a)). The upper face region is again divided into two halves in a vertical direction (Fig. 2 (a)) so that each eye can be analyzed separately.

The positions of the irises are located in the segmented eye regions by sequentially applying the analysis of the vertical histogram (showing the intensity differences between the successive rows, pixel-wise) and then the horizontal histogram (showing the intensity differences between the successive columns, pixel-wise). The peak of the vertical histogram of the eye-region box corresponds with the y-coordinate of the iris, and the peak of the horizontal histogram of the eye-region box corresponds with the x-coordinate of the iris. By knowing y and x coordinates of both irises, we are able to calculate the angle that they make with the horizontal plane and, if necessary, to rotate the image for that angle. In this way, possible in-plane rotations of the face can be eliminated. With this method we achieve a detection rate of 100% (i.e., all the segmented ROIs were correctly identified) on the test set of 422 Cohn-Kanade database images.

To locate the medial point of the mouth we first define a ROI of the mouth. Since we know the distance between the irises (ED), we define the mouth region to be the horizontal strip whose top is at 0.85×ED from the eyes horizontal position and has a height equal to 0.65×ED (Fig. 2). In that region horizontal and vertical thresholded edges will give us the shape of the mouth. Analysis of the vertical histogram of such a thresholded mouth region, obtains the graph similar to the one illustrated in Fig. 1(b). The center of the widest peak will define vertical position of the medial point of the mouth. By choosing the widest peak, the possibility of detecting the nose instead of the mouth is avoided. The horizontal position of the point in question is defined as the middle point between the eyes. Fig 1(b) shows typically detected positions of the eyes and mouth.

We regard the detection scheme described above as successful if the eye position was detected within the iris. For the test set of 422 images from the Cohn-Kanade database, the detection rate for the irises was 100%. For the medial point of the mouth, the detection rate was 99%, i.e., we had 2 false detections for our test set.



(a) (b)

Fig. 2. (a) Dividing the face horizontally in half and dividing the upper face region vertically in half. (b) Finding the mouth region within the face region by means of Eye Distance (ED)

Subsequently, we use the detected positions of the irises and the medial point of the mouth to divide the face into 20 regions so that each of the points to be localized is within one ROI. An example of ROIs extracted from the face region for points B, I, and J, is depicted in Fig. 1(b).

2.3 Feature Extraction

The proposed facial feature point detection method uses individual feature patch templates to detect points in the relevant ROI. These feature models are 13×13 pixels GentleBoost templates built from both gray level intensities and Gabor wavelet features.

Recent work [14] has shown that a Gabor approach for local feature extraction outperformed PCA (Principal Component Analysis), FLD (Fisher's Linear Discriminant) and LFA (Local Feature Analysis). The essence of the success of Gabor filters is that they remove most of the variability in image due to variation in lighting and contrast, at the same time being robust against small shift and deformation [15]. Gabor wavelets seem to be a good approximation to the sensitivity profiles of neurons found in visual cortex of higher vertebrates [16]. There is evidence that those cells tend to come in pairs with even and odd symmetry [17], [18] similar to the real and imaginary part of Gabor-based wavelets.

A 2D Gabor filter $\psi(x,y)$ can be defined as:

$$\psi(x,y) = \frac{\alpha\beta}{\pi} e^{-(\alpha^2 x'^2 + \beta^2 y'^2)} e^{j2\pi f_0 x'} \quad (1)$$

$$x' = x\cos\theta + y\sin\theta,$$

$$y' = -x\sin\theta + y\cos\theta,$$

where f_0 is the central frequency of a sinusoidal plane wave, θ is the anti-clock wise rotation of the Gaussian and the plane wave, and α and β are the parameters for scaling two axis of the elliptic Gaussian envelope. Here we consider that the orientation of the Gaussian envelope and the orientation of the sinusoidal function are the same (which is one the characteristics of complex cells of the mammals' visual cortex). The Gabor function is actually Gaussian shaped function (first part of the equation (1))

which modulates sinusoidal plane wave carrier (second part of the equation (1)). Its 2D Fourier transform is:

$$\psi(u,v) = e^{-\pi^2\left(\frac{(u\cos\theta+v\sin\theta-f_0)^2}{\alpha^2}+\frac{(u\sin\theta-v\cos\theta)^2}{\beta^2}\right)} \quad (2)$$

By fixing the ratio of the frequency of the wave and the sharpness of the Gaussian we get that the spatial filter (1) includes a constant number of waves. The ratios which are known to hold for the cells in human visual cortex are [16]:

$$\gamma = \frac{f_0}{\alpha} = \frac{1}{\sqrt{\pi \cdot 0.9025}} \quad (3)$$

$$\eta = \frac{f_0}{\beta} = \frac{1}{\sqrt{\pi \cdot 0.58695}} \quad (4)$$

Thus, a normalized filter can be presented in spatial domain as:

$$\psi(x,y) = \frac{{f_0}^2}{\pi\gamma\eta} e^{-\left(\frac{f_0^2}{\gamma^2}x'^2+\frac{f_0^2}{\eta^2}y'^2\right)} e^{j2\pi f_0 x'}, \quad (5)$$

and in frequency domain as:

$$\Psi(u,v) = e^{-\frac{\pi^2}{{f_0}^2}(\gamma(u'-f_0)^2+\eta^2 v'^2)}, \quad (6)$$

$$u' = u\cos\theta + v\sin\theta,$$
$$v' = -u\sin\theta + v\cos\theta.$$

Thus, in the frequency domain, the filter is an oriented Gaussian with orientation θ centered at frequency f_0. Gabor filter formulated in this way has the response at zero frequency (DC-response) which has a value close to 0 and is the same for all central frequencies. This ensures that the method is insensitive to illumination variations.

Several Gabor filters are combined to form a filter bank. The filter bank is usually composed of filters in several orientations and frequencies, with equal orientation spacing and octave frequency spacing, while the relative widths of Gaussian envelope γ and η stay constant. In the frequency domain Gabor filter must obey Nyquist rule, which means that

$$f_0 \leq 0.5, \text{ for each } \theta. \quad (7)$$

Feature vector for each facial point is extracted from the 13×13 pixels image patch centered on that point. This feature vector is used to learn the pertinent point's patch template and, in the testing stage, to predict whether the current point represents a certain facial point or not. This 13×13 pixels image patch is extracted from the gray scale image of the ROI and from 48 representations of the ROI obtained by filtering the ROI with a bank of 48 Gabor filters at 8 orientations and 6 spatial frequencies (2:12 pixels/cycle at ½ octave steps). Thus, 169×49=8281 features are used to represent one point. Each feature contains the following information: (i) the position of the pixel inside the 13×13 pixels image patch, (ii) whether the pixel originates from a grayscale or from a Gabor filtered representation of the ROI, and (iii) if appropriate, which Gabor filter has been used (See Fig. 1(c)).

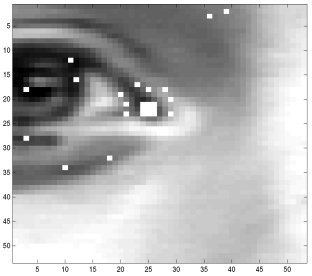


Fig. 3. Positive and negative examples for training point B. The big white square on the inner corner of the eye represents 9 positive examples. Around that square are 8 negative examples randomly chosen near the positive examples. Another 8 negative examples are randomly chosen from the rest of the region.

2.4 Feature Classification

In the training phase, GentleBoost feature templates are learned using a representative set of positive and negative examples. As positive examples for a facial point, we used 9 image patches centered on the true point and on 8 positions surrounding the true (manually labeled) facial point in a training image. For each facial point we used two sets of negative examples. The first set contains 8 image patches randomly displaced 2-pixels distance from the true facial point. The second set contains 8 image patches randomly displaced in the relevant ROI (Fig. 3). Thus, for each ROI, we have 9 positive and 16 negative examples, meaning that there is a 25×8281 size matrix representing training data for each ROI for each training image. Even though each feature can be computed very efficiently, computing the complete set is computationally expensive. Adding the fact that such a representation of features is highly redundant, we used GentleBoost technique to reduce the dimensionality.

In contrast to AdaBoost, GentleBoost [19] uses real valued features. GentleBoost seams to converge faster than AdaBoost, and performs better for object detection problems [20]. It is simple to implement, it is numerically robust and it has been shown experimentally to outperform (with respect to detection accuracy) other boosting variants for the face detection task [21]. The performance of boosting methods on data which are generated by classes that have a significant overlap, in other words, classification problem where even the Bayes optimal prediction rule has a significant error is discussed in [22]. For this case, GentleBoost performs better than AdaBoost since AdaBoost over-emphasizes the atypical examples which eventually results in inferior rules. As explained in [22], the reason for this might be that GentleBoost gives less emphasis to misclassified examples since the increase in the weight of the example is quadratic in the negative margin, rather than exponential.

The outline of the GentleBoost algorithm is as follows. At each boosting round, a regression function is fitted (by weighted least-squared error) to each feature in

the training set. The fitting of the regression function is done for one feature for all training examples by minimizing the weighted error

$$errorW = \frac{\sum_i (w_i | y_i - (a_i (x_i > th_i) + b_i |^2)}{\sum_i w_i}, \quad (9)$$

where i is the i-th training example. By minimizing weighted error through all features, we get the feature with the smallest error and with the adequate parameters which minimize this error (a, b and th). Next step is estimation of the fitting function fm for each training example with this parameters:

$$fm_i = (a \cdot (x_i(FeatureIndex) > th) + b), \quad (10)$$

where $FeatureIndex$ is the feature which is chosen in the round m. The next step is to update the classifier output and the weights for each training example:

$$F(x_i) = F(x_i) + fm_i \quad (11)$$

$$w_i = w_i e^{-y_i f_{mi}} \quad (12)$$

Finally, the weights should be renormalized and for each testing example x_i the output of the classifier should be calculated as:

$$sign[F(x_i)] = sign[\sum_{m=1}^{M} f_m(x_i)] \quad (13)$$

where M is the number of the most relevant features the classifier has chosen for the classification.

Eventually, in the testing phase, each ROI is filtered first by the same set of Gabor filters used in the training phase (in total, 48 Gabor filters are used). Then, for a certain facial point an input 13×13 pixels sliding window is slid pixel by pixel across 49 representations of the ROI (grayscale plus 48 Gabor filter representations; see Fig. 1(c)). For each position of the sliding window, GentleBoost classifier outputs a response depicting the similarity between the 49-dimensional representation of the sliding window compared to the learned feature point model. After scanning the entire ROI, the position with the highest response (i.e., with the largest positive sum $F(x_i)$ in Equation (11)) reveals the feature point in question.

3 Results

3.1 Training Set

The facial feature detection method was trained and tested on the Cohn-Kanade database [10], which consists of approximately 2000 gray-scale image sequences in nearly frontal view from over 200 subjects, male and female, being 18 to 50 years old. From those some 480 samples were made publicly available. Each video pictures a single facial expression and ends at the apex of that expression while the first frame of every video sequence shows an expressionless face. For our study, we used only the first frames of 300 Cohn-Kanade database samples. No further registration of the images was performed. Note, however, that for all first frames from the Cohn-Kanade database, the variation in the inter-ocular distance (the distance between the eyes) is max 30%, i.e., minimum distance measured was ±80 pixels and maximum distance measured was ±120 pixels. Thus, in the case that the inter-ocular distance measured in the input image is way below ±80 pixels or way above ±120 pixels, we cannot guarantee that the method's performance reported below will remain the same. The actual influence of such occurrences on the performance of the method is, however, the matter of future experimental studies.

3.2 Experimental Results

The 300 images of the data set were divided into 3 subsets containing 100 images each. The proposed method has been trained and tested using a leave-one-subset-out cross validation. To wit, training and testing procedure was repeated 3 times. Each time one of the 3 subsets was used as a test set and the other 2 subsets were used as a training set. We applied this method for 19 facial feature points depicted in Fig. 1(e), while point N (the tip of the nose) was defined as the middle point between points H and H1.

To evaluate the performance of the method, each of the automatically located facial points was compared to the true (manually annotated) point. As explained above, we used as positive examples the true location of the point and 8 positions surrounding the true facial point in a training image. Hence, automatically detected points displaced 1-pixel distance from relevant true facial points are regarded as SUCCESS. Additionally, we define errors with respect to the inter-ocular distance measured in the test image. An automatically detected point displaced in any direction, horizontal or vertical, less than 10% of inter-ocular distance from the true facial point is regarded as SUCCESS.

However, for some facial points, variations in vertical direction are considered more cumbersome than variations in horizontal direction. This is mostly due to the fact that the main focus of our research is to develop an automatic facial point tracker, the output of which could be used for automatic facial expression analysis. For 2D tracking, an initial template for each facial point is sampled from the first frame of the input image sequence. This template is updated throughout the sequence during tracking. Up to now, the location of the initial sample templates was manually selected. Thus for automation, the tracker requires automatic detection of the facial points in question. Ideally, this automatic detection will resemble manual annotation of the facial points.

This means that points G, G1, F, and F1 (outer mid points of the eyes, Fig. 1(e)) will be detected on the border between the sclera and the eyelash. In turn, to favor such detecting of the outer mid points of the eyes, we applied the above mentioned rule only in horizontal direction. In vertical direction, an automatically detected point displaced 3-pixels distance from the true point is regarded as SUCCESS, given that the radius of the iris is ±25 pixels. In addition, the horizontal position of point L (the bottom of the lips) is specified by the horizontal position of point K (the top of the lips) and we search for the correct point by varying only the vertical coordinate.

Table 1. Facial Feature Point Detection results for 300 samples from the Cohn-Kanade database

Detected Point	Detect. Rate
A: Outer corner of the left eye	0.92
A1: Outer corner of the right eye	0.96
B: Inner corner of the left eye	0.96
B1: Inner corner of the right eye	0.99
G: Bottom of the left eye	0.95
G1: Bottom of the right eye	0.99
F: Top of the left eye	0.91
F1: Top of the right eye	0.83
D: Inner corner of the left eyebrow	0.96
D1: Inner corner of the right eyebrow	0.95
E- Outer corner of the left eyebrow	0.96
E1- Outer corner of the right eyebrow	0.90
H-Left nose corner	0.98
H1-Right nose corner	0.97
I-Left mouth corner	0.97
J-Right mouth corner	0.91
K-Mouth top	0.93
L-Mouth bottom	0.80
M-Chin	0.90
AVERAGE RATE FOR ALL POINTS	0.93

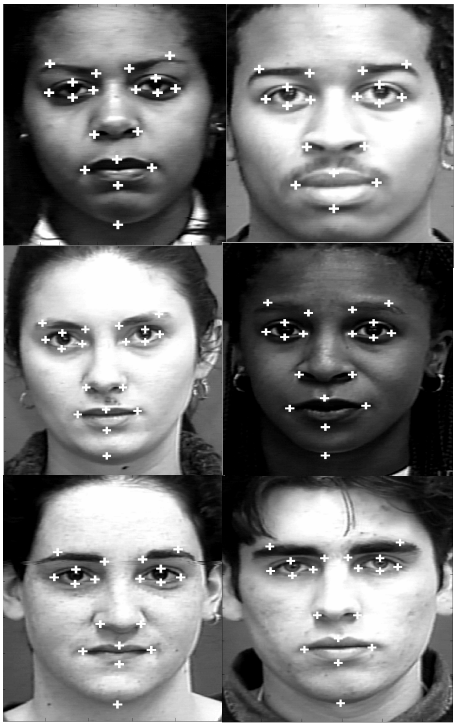


Fig. 4. Accurate detection of all facial points

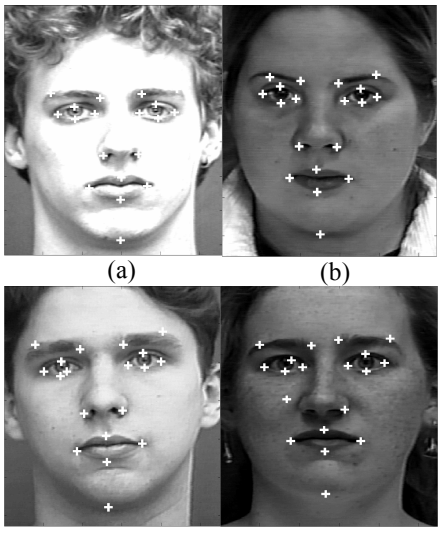


Fig. 5. Inaccurate detection of facial points: (a) point D1, (b) point E, (c) point B, (d) points F and H. For feature point notation see Fig. 1(e).

Overall, we achieved an average recognition rate of 93% for 20 facial feature points using the above described evaluation scheme. The detection rates for each point are shown in Table 1. Typical results are shown in Fig. 4. Most misclassifications (encountered mostly with points F1 and M) can be attributed to the lack of consistent rules for manual annotation of the points. Typical misclassifications are illustrated in Fig. 5.

It is interesting to mention that detailed analysis of the Gentleboost classifiers revealed that the vast majority of features (over 98%) were selected from the Gabor filter components rather than from the grayscale values.

4 Conclusion

In this paper we present a robust, highly accurate method for fully automatic detection of 20 facial feature points in images of expressionless faces using Gabor feature based boosted classifiers. When tested on images from the Cohn-Kanade database, with possible in-plane head rotations and recorded under various illumination conditions, the method has achieved average recognition rates of 93%.

In future work we will investigate effects of using a reduced number of features for classification. Also, we plan to conduct extensive experimental studies using other publicly available face databases.

Acknowledgement

The authors would like to thank Jeffrey Cohn of the University of Pittsburgh for providing the Cohn-Kanade database. The work of D. Vukadinovic is supported by the Netherlands BSIK-MultimediaN-N2 Interaction project.

The work of M. Pantic is supported by the Netherlands Organization for Scientific Research Grant EW-639.021.202.

References

[1] I. Matthews and S. Baker, "Active Appearance Models Revisited", *Int'l Journal Computer Vision*, vol. 60, no. 2, pp. 135 – 164, 2004.

[2] M.J.T. Reinders, et al., "Locating Facial Features in Image Sequences using Neural Networks", *Proc. IEEE Int'l Conf. Face and Gesture Recognition*, pp. 230-235, 1996.

[3] E. Holden, R. Owens, "Automatic Facial Point Detection", *Proc. Asian Conf. Computer Vision*, 2002.

[4] R.S. Feris, et al., "Hierarchical Wavelet Networks for Facial Feature Localization", *Proc. IEEE Int'l Conf. Face and Gesture Recognition*, pp. 118-123, 2002.

[5] C. Hu, et al., "Real-time view-based face alignment using active wavelet networks", *Proc. IEEE Int'l Workshop Analysis and Modeling of Faces and Gestures*, pp. 215-221, 2003.

[6] S. Yan, et al., "Face Alignment using View-Based Direct Appearance Models", *Int'l J. Imaging Systems and Technology*, vol. 13, no. 1, pp. 106-112, 2003.

[7] L. Wiskott, et al., "Face recognition by elastic bunch graph matching". *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.

[8] D. Cristinacce, T. Cootes, "Facial Feature Detection Using AdaBoost With Shape Constrains", *British Machine Vision Conference*, 2003.

[9] L. Chen, et al., "3D Shape Constraint for Facial Feature Localization using Probabilistic-like Output", *Proc. IEEE Int'l Workshop Analysis and Modeling of Faces and Gestures*, pp. 302-307, 2004.

[10] T. Kanade, et al., "Comprehensive database for facial expression analysis", *Proc. IEEE Conf. FGR*, pp. 46-53, 2000.

[11] M. Pantic and I. Patras, "Detecting Facial Actions and their Temporal Segments in Nearly Frontal-View Face Image Sequences", *Proc. IEEE Conf. Systems, Man and Cybernetics*, 2005.

[12] I.R. Fasel, et al., "GBoost: A generative framework for boosting with applications to realtime eye coding", *Computer Vision and Image Understanding*, in press. (For source code see http://kolmogorov.sourceforge.net.)

[13] P. Viola, M. Jones, "Robust real-time object detection", *ICCV Workshop on Statistical and Computation Theories of Vision*, 2001.

[14] G. Donato, et al., "Classifying Facial Actions", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974-989, 1999.

[15] http://rita.osadchy.net/papers/gabor-3.pdf.

[16] J. Jones, L.Palmer, "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex," *J. Neurophysiology*, pp. 1233-1258, 1987.

[17] D. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *J. Opt. Soc. Amer. A,* vol. 4, no. 12, pp. 2379-2394, 1978.

[18] D. Burr, et al., "Evidence for edge and bar detectors in human vision," *Vision Res.,* vol. 29, no. 4, pp. 419-431, 1989.

[19] J Friedman, et al, „Additive logistic regression: A statistical view of boosting". *Annals of Statistics*, vol. 28, no. 2, pp. 337–374, 2000.

[20] A. Torralba, et al., "Sharing features: efficient boosting procedures for multiclass object detection", *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2004.

[21] R. Lienhart, et al., "Empirical analysis of detection cascades of boosted classifiers for rapid object detection", *Proc. German Pattern Recognition Symposium*, 2003.

[22] Y. Freund, R.E. Schapire, "Discussion of the Paper 'Additive Logistic Regression: A Statistical View of Boosting'", *The Annals of Statistic*, January 28, 2000.