International Conference on
Information, Communications and Signal Processing
ICICS '97
Singapore, 9-12 September 1997

Automatic Segmentation of Training Set for Facial Feature Detection

H. Demirel, T.J.W. Clarke and P.Y.K. Cheung

Department of Electrical and Electronic Engineering
Imperial College of Science, Technology and Medicine
Exhibition Road, London SW7 2BT, UK

Abstract

In conventional image-based feature detection time consuming pre-processing step is required to manually segment the training features from unsegmented face images. In this paper we present a novel method of using automatically segmented facial image data for facial feature detection. A quality measure is defined to identify those image data from a large training set that are better to describe the feature. The best quality subset is then extracted and used to train the feature detector. The detection performance obtained by automatically segmented data set after refinement is almost as high as that obtained by feature detector trained by a manually segmented set.

1. Introduction

Automatic detection of facial features is an important task in facial image processing. Over the years, various strategies have been proposed. Vincent, Waite and Myers [5] used multilayered perceptrons and multi-resolution idea to detect facial features. Deformable templates by Yuille, Cohen and Hallinan [7], as well as, eigenfeatures, derived from Principal Component Analysis (PCA) technique by Pentland, Moghaddam and Starner [3], were also proposed.

There are two main approaches for facial feature detection. One method is to use geometrical information of facial features. Each feature is described as geometrical shapes: for example eye pupil is an exact circle. This method can produce good results, but it takes complete redesign for different features. The second method is image-based feature detection, where a typical set of manually segmented images are used to describe the facial features. Image-based feature detection method requires careful selection of the typical facial features as the training set. The selection requires human supervision and takes a lot of time. In this paper, we propose a novel method of automatically segmenting facial image patches to be used as the training set for the feature detection. We defined a quality measure to identify the image data from a large automatically segmented set that are best to describe the feature. Some images in this data set are *low quality* noisy data (i.e. major displacement of the intended facial feature within the image), and some of the images are *high quality* data (i.e. the center of the image is also the center of the intended feature). The automatically segmented image set is called the *initial training set*. We introduced a refinement method based on the quality measure of the features of the initial training set. This method helps us to find a subset of the initial training set, that contains high quality data. This subset produces high detection performance.

Feature detection performance depends on the quality of the segmented facial features. In manual segmentation, we use human judgment of a typical feature. Depending on the judgment of different individuals, different training sets giving different detection performances are obtained. In our method we by-pass the manual segmentation and employ a method for automatic segmentation of features. We only use the average face obtained by adding M frontal faces divided by M. The average face contains enough information for us to decide which common visual features exist among the contributing faces, and which locations are the most likely locations of these features. Average face is a blurred face-like image. This is the result of varying expressions (open/closed eyes, smiling/not smiling), facial details (glasses/no glasses) as well as small side movements of the individuals. The average face has blurred facial features (eyes, nose, mouth etc.). We call these visual features on the average face as *average features*. The existence of these features at a particular location gives us an idea of the shape and the possible location of that feature. For example, an average feature at location (x,y) suggests that, this location is the most likely location of that feature in all the faces contributing to the average face.

Once we get an average face we decide which visual features we will use and where their locations are. We automatically segment images at those locations. Some of the segmented images may be nowhere close to a feature which was targeted. For example, even if we are trying to make an eye detector, some of the images segmented can be the middle of the nose in some extreme cases. This is mostly because of the side movement of the individual face. But majority of the segmented data are very close to be an eye feature.

When we automatically segment the set of images contributing to the average feature, we apply a refinement technique to obtain a subset which is composed of higher quality facial features. The rest of the images are disregarded. So, a subset of a initial training set is obtained for better detection. In order to refine the initial training set we simply use *average template* obtained by enhancing the gray level pixel intensity of average feature. This enhancement process transforms the noisy average feature to a template which can be used as a model to represent the shape information of the of the intended feature. We use

the average template to calculate matching distance between the template and each image in the initial training set. Because the obtained template looks like the intended feature (i.e. eye), it is very useful for rough refinement of the initial training set. For each image in the initial training set we calculate the matching distance between the image and the average template of that feature. The images with smallest distance values are the ones the most similar to the average template. Hence they are higher quality features. By taking first M ($M \leq N$ where N is the number of automatically segmented images) images with smallest matching distances we get the best quality subset of the initial training set. The larger the initial training set is, the more likely that higher quality images are included in this set. So, larger initial training set produces better performance.

Once a subset of higher quality images are obtained we use eigenfeatures technique [3] for feature detection. Eigenfeatures technique is a feasible way of PCA in feature detection and explained in section 2. Section 3 describes average face and the generation of the initial training set. Section 4 explains the refinement of the initial training set. Section 5 and 6 gives experimental results and discussion & conclusions respectively.

2. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a technique of mapping high dimensional data onto lower dimensional space. The mapping depends on the statistics of the training set. PCA is widely used in signal and image processing. The technique is very popular for image recognition [4], image coding [6] and feature detection [3]. The use of principal components for synthesizing and recognizing images was first suggested by Kirby and Sirovich [2]. They used principal components to represent picture of faces. Turk and Pentland [4] developed a feasible PCA technique to recognize faces (*eigenfaces technique)*.

2.1 Calculation of Principal Components (eigenfeatures)

Let the training set of facial feature images be $I_1, \ldots, I_M$ where M is the number of the manually segmented features in the training set. The average image is calculated and a set of difference images are derived by subtracting the average image from each image in the training set. Then the difference images are converted into single-column vectors by concatenating consecutive rows of pixel values. The average feature is given by

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} (I_i) \tag{1}$$

and each difference feature is given by

$$\Phi_i = I_i - \Psi, \qquad i = 1, \ldots, M \tag{2}$$

A set of principal component vectors, u_k , which maximize, λ_k are sought where

$$\lambda_k = \frac{1}{M} \sum_{i=1}^{M} (u_k{}^T \Phi_i)^2 \tag{3}$$

subject to $u_j{}^T u_k = 0$, for $j<k$. The vectors u_k and scalars λ_k are the eigenvectors and eigenvalues, respectively, of the covariance matrix

$$C = \frac{1}{M} \sum_{i=1}^{M} (\Phi_i \Phi_i{}^T) \tag{4}$$

Since the number of images in the training set is less than the dimension of the space M $(M<N_h x N_v)$, where N_h is the number of pixels on a line and N_v the number of lines, and the average image is subtracted from each image, there will only be $M-1$ nonzero eigenvectors [2]. The eigenvectors can be calculated by constructing the matrix L, where $L_{ij} = \Phi_i{}^T \Phi_j$ and finding its eigenvectors v_k. Then the eigenvectors u_k are given by

$$u_k = \sum_{i=1}^{M} (v_{ki} \Phi_i), \quad k = 1, \ldots, M-1 \tag{5}$$

When the eigenvectors are obtained from the training set, other images outside the set can be represented as follows. The new image I_{new} is transformed into its eigenvector (principal) components by a simple operation: $\omega_k = u_k{}^T (I_{new} - \Psi)$ $k=1, \ldots, M-1$.

The representation vector $\Omega^T = [\omega_1 , \omega_2 , \ldots , \omega_{M-1}]$ describes the contribution of each eigenvector in representing the new image.

In our work we use facial features as the training set. Following the terminology presented by Pentland, Moghaddam and Starner [3] we call the eigenvectors obtained from this set as *eigenfeatures*. When a training set of M features are used we obtain M-1 nonzero eigenfeatures. Not all of these eigenfeatures are needed to represent the facial features [4]. The first M' ($M' \leq M$-1)eigenfeatures with highest associated eigenvalues are used to represent a given image.

2.2 Reconstruction Error for Feature Detection

Reconstruction Error, also known as *residual error* and *Distance From Feature Space*, is a very good indicator for feature detection. The pixels which are positioned at the center of the possible subimages of the face which can contain the feature, are mapped into eigenspace. After each possible pixel is processed, the reconstruction error values are associated to each pixel. The pixel with minimum reconstruction error is the location of the feature that detected.

Given a new image I_{new} the reconstruction error ε_r is defined:

$$\varepsilon_r = (I_{new} - \Psi) - \sum_{i=1}^{M} \omega_i u_i \tag{6}$$

3. Average Face and the Generation of the Training Set

Let F_i (i=1, … ,M) to be a frontal face where M is the number of the faces, then the *average face* Ψ_F is defined by

$$\Psi_F = \frac{1}{M} \sum_{i=1}^{M} F_i \qquad (7)$$

Average face provides us valuable information about the dominant features of the faces contributing to it. For example, if most of the faces are dark and few of the faces are light, the average face is bound to be dark. Similar argument applies to the common visual features of the faces in the contributing to the average face. For example the left eyes of the frontal faces are bound to be on the upper left of the face images. Although the location of a particular feature varies from face to face, most of them clusters around a particular location. This location corresponds to the mean of the locations of that particular feature of the faces in the given training set. There are variations in expressions (open/closed eyes, smiling/not smiling), facial details (glasses/no glasses), feature sizes and small side movement changes on the faces. As a result of these variations the average face is a blurred face-like image. Because it looks like a face, we can identify the visual features of the average face. These are average features. Figure 1 shows samples of frontal faces as well as average face obtained by using 100 frontal faces.

The average features give us some idea about the shape of the common facial features of the frontal faces contributing to the average face. In addition, the location of this average feature gives us some idea about the statistical mean location of the corresponding feature contributing to the average feature.

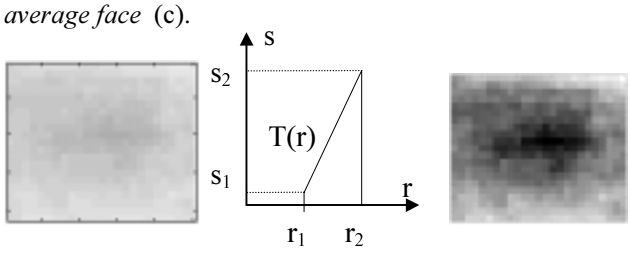
Initial training set is segmented automatically based on the location obtained from the average face. Because there is only one average image we need to find the location of the average features only once. When this location is obtained we automatically segment features from all the faces contributing to the average face at that location. By this way we get low quality noisy features as well as high quality features. Our approach is to automatically segment as many facial features as possible. In the next section we introduce a method of isolating high quality images from low quality images.

4. Refinement of the Automatically Segmented Training Set

The features obtained from automatic segmentation contains high and low quality images. The average of these features can hardly be recognized as a feature. But as we can identify it on the average face as a feature, that means the majority of the contributing images are high quality features dominating the low quality features. Because we use large number of faces the noise introduced by the low quality images cancels each other up to a certain degree. But, still the average feature is very blurred (Figure 2(a)).

(a) (b) (c)

Figure 1: The sample faces (i.e. a, b) are used to obtain the *average face* (c).



(a) (b) (c)

Figure 2: Contrast stretching transformation function (b) is used to enhance *average feature* (a) to obtain *average template* (c).

We propose to use the average feature to obtain average template. The average template is defined to be enhanced version of the average feature that contains much clearer shape information than the average feature. The quality of a given image is calculated as the matching distance between this template and the image. The quality measure is used to identify the best quality features from the initial training set. Average feature does not contain enough shape information. This is because the dynamic range of the gray level intensity pixels of the average image is very small. If we increase the range of this dynamic range, we can obtain a new image - *average template* - containing enough shape information describing the feature. This is a standard image enhancement method, also known as *contrast stretching*.

Let r and s be the gray level intensity of pixels before and after contrast stretching. Figure 2(b) shows the transformation function we have used for contrast stretching of the average feature. r_1 is the minimum and r_2 is the maximum intensity level in the average feature. s_1 and s_2 are the average minimum and average maximum gray level intensity levels of the images contributing to the average feature respectively. The values of s_1 and s_2 are calculated by

$$s_1 = \frac{1}{M} \sum_{i=1}^{M} \min\,(\textit{Intensity of } I_i)\,, \qquad (8)$$

$$s_2 = \frac{1}{M} \sum_{i=1}^{M} \max\,(\textit{Intensity of } I_i) \qquad (9)$$

where I_i is the i'th automatically segmented feature of the initial training set and M is the number of automatically segmented features.

Given average feature with gray level intensity, r, the contrast stretched average template with gray level intensity, s, can be computed by applying the piece-wise transformation function shown in Figure 2(b) to every pixel. The function is defined by

$$s = T(r) = \frac{(s_2 - s_1)}{(r_2 - r_1)}(r - r_1) + s_1 \quad (r_1 \le r \le r_2), \qquad s = T(r) = 0 \quad (r < r_1 \ , \ r > r_2) \tag{10}$$

After obtaining the average template (Figure 2(c)) we compute the matching distance between average template and each feature in the training set. The *matching distance* is defined to be the Euclidean distance between the given image and the average template. The matching distance is the key quality measure we use to identify the best quality images from the initial training set. The images with the lowest matching distance is the best quality image, where the image with the highest distance is the worst quality image. Because the average template contains some shape information of the feature, the matching distance shows us how similar the average template and the other image features are. Our approach is to obtain a subset with the lowest matching distances of the initial training set. The average template may not be the best model for the feature it represents. But, because it contains average information obtained from the complete set, the matching distance gives us some idea about the similarity of the feature and the average template. We are not claiming that the use of average template is the best method for refinement. The advantage of this average template is that it does not require to be manually designed. A simple image enhancement method is enough to get average template from the average feature.

5. Experimental Results

In our experiments we have used gray scale face images. The images have varied lighting, facial expressions and facial details. All the images were taken against a dark homogeneous background with the subjects in upright, frontal position (with tolerance for some side movement). The size of each face image is 112x92 pixels with 256 gray levels per pixel. The size of the left and the right eye features, which are used in out experiments, is 21x25.

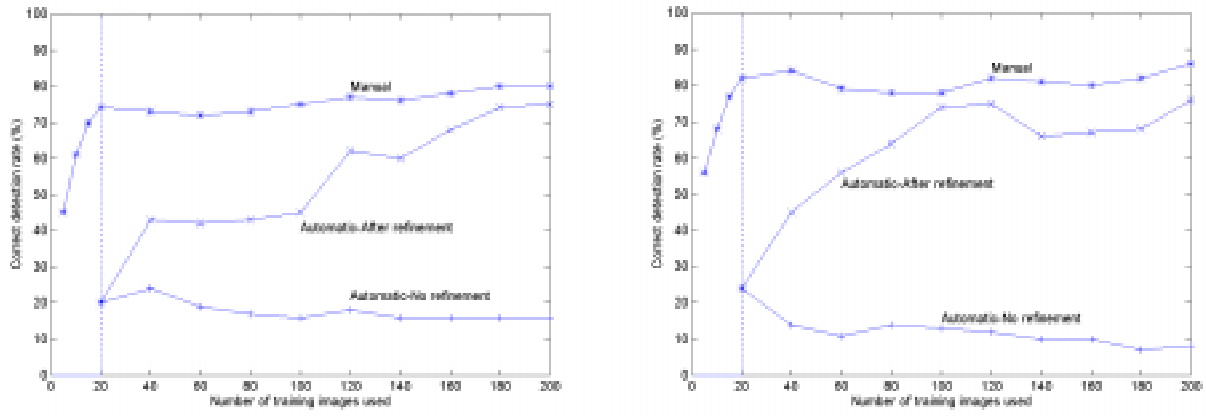
In the experiments we have used 200 faces for training and 100 faces for the testing of the detection performance of the feature detectors. The performance of the feature detectors have been calculated by using *correct detection rate* [1]. The correct detection rate is the percentage of the fraction of the test set images correctly detected with a tolerance of 5 pixels.

In the first experiment manually segmented left eye features are used as training set and then the correct detection rate was calculated for different number of training sets. We have observed rapid increase in the detection performance when we use manually segmented features from 5 to 20. The performance of the feature detector changes slightly with more training feature. It was observed that 20 features gives %74 and 200 features give %80. This is because more or less 20 features is enough to represent a left eye feature. More added features improve the performance slightly. By making use of this observation we decided to have 20 images in our subset obtained from the automatically segmented features after refinement(explained in Section 4). Starting from 20 images to 200 images 10 automatically segmented initial training sets were used by adding 20 new images to the previous set.

In the first initial training set of 20 images we used all set as the refined set. The second set contained 40 images. After refinement 20 refined images were selected for the training of the feature detection. The third set contained 60 images, and again 20 images were selected. This was repeated until 200 automatically segmented set was used. The performance of the first set is %20, where performance obtained by the refined subset of 200 training images is %75. The results suggest that as the number of the training set increases, provided the refined set stays at the fixed number, the performance increases. Meanwhile we have also calculated the performance of the automatically segmented training set without refinement. The result is dramatic. The training set which is not refined shows very low performance, which is between %16 and %20. Figure 3(a) shows the experimental results of the first experiment.

In the second experiment same approach have been applied to the right eye. Again the subset of the refined set is fixed to 20. The performance of the refined set from the automatically segmented training set starts from %24 and increases to %76, as we increase the number of the training set from 20 to 200. The performance without refinement stays between %8 to %24. Figure 3(b) shows the experimental results of the second experiment.

The experimental results suggest that as we increase the number of automatically segmented training images the performance obtained by using the refined subset increases. Performance obtained by our method is very close to the performance of the manual segmentation. Fluctuations on the performance are because of the refinement process. The use of noisy average template obtained by a simple image enhancement operation is not accurate enough. But it proves that by using refinement approach we can employ automatic segmentation of facial features to obtain high detection performance almost as high as manual segmentation.



(a) (b)

Figure 3: Performance of the feature detectors for the left (a) and the right (b) eyes for manual, automatic with refinement, and automatic without refinement.

6. Discussion and Conclusions

In this paper we have introduced a novel method for using automatically segmented images for the facial feature detection. The automatically segmented image set contains high and low quality feature images. Our approach is to obtain a subset with high quality images. We perform this by introducing a refinement method. The refinement method is based on the matching distances associated to every image, showing how similar they are to the average template which is a rough model of the intended feature. We use fixed number of images for the refined subset. As we increase the number of automatically segmented images, we increase the possibility of including higher quality facial features to be included in the initial training set. That is why, as we increase the number of automatically segmented features, the detection performance obtained from the refined subset increases.

The refinement method basically depends on the template obtained from the average feature. The template is noisy, because it depends on the average feature. But, it contains average information contributed from all of the automatically segmented images, and the extent of the dissimilarity of the automatically segmented features is very high. So, the matching distance between the template and each image is good enough for us to select a reliable high quality subset. We do not claim that the matching distance between the average template and the given image is the best indicator for the quality. Alternative refinement methods can easily be employed. The importance lies on the idea of the refinement. We claim that the idea of refinement of facial features segmented automatically at a location where it is most likely to exist is very useful.

The detection performance obtained after the refinement of initial training set is high. It is almost equal to the performance of the feature detectors obtained by manually segmented training set. This method can be improved by employing a fine tuning type of refinement in addition to the simple primary refinement technique we introduced. Such a refinement method is currently being developed.

References

[1] H. Demirel, T. J. W. Clarke, and P. Y. K. Cheung, "Adaptive Automatic Facial Feature Segmentation," Second International Conference on Face and Gesture Recognition, pp. 277-282, Vermont, USA, Oct 1996.

[2] M. Kirby, and L. Sirovich, "Application of Karhunen-Loeve Procedure for Characterisation of Human Faces," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 1, 1990.

[3] A. Pentland, B. Moghaddam, and T. Starner, "View-based and Modular Eigenspaces for Face Recognition," IEEE Conference on Computer Vision and Pattern Recognition, 1994.

[4] M. Turk, and A. Pentland, "Eigenfaces for Recognition," Journal of Cognitive Neuroscience, 3(1):71-86, 1991.

[5] J. M. Vincent J. B. Waite, and D. J. Myers, "Automatic Location of Visual Features by a System of Multilayered Perceptrons," IEE Proceedings, vol. 139, no. 6, Dec. 1992.

[6] W. J. Welsh, D. Shah, "Facial-feature Image Coding Using Principal Components," IEE Electronic Letters, vol. 28, no. 22, Oct 1992

[7] A. L. Yuille, D. S. Cohen, and P. W. Hallinan, "Feature Extraction from Faces Using Deformable Templates," Proceedings of CVPR, San Diego, CA, June 1989.