Facial Feature Extraction and Determination of Pose

Athanasios Nikolaidis
Department of Informatics, Aristotle University of Thessaloniki
Thessaloniki, GREECE

Ioannis Pitas
Department of Informatics, Aristotle University of Thessaloniki
Thessaloniki, GREECE

Abstract

The present paper describes a method for the extraction of facial features. The ultimate goal of the approach followed is to define a sufficient set of feature distances so that a unique description of the structure of a face is produced. Eyebrows, eyes, nostrils, mouth, cheeks and chin are considered as interesting features. The candidates for eyes, nostrils and mouth are determined by searching for minima and maxima in the $x-$ and $y-$ projections of the greylevel relief. The candidates for cheek borders and chin are determined by performing an adaptive Hough transform on a relevant subimage defined according to the position of an ellipse containing the main face region of the image. A technique based on dynamic programming is applied that exploits this ellipse in order to acquire a more accurate model of the face. The candidates for eyebrows are determined by adapting a proper greylevel mask to an area defined by the eye position. Finally, the orientation of face is determined using the symmetric properties of certain facial features. The algorithms presented were tested on the M2VTS multimodal face database.

1 Introduction

A basic topic in building automated systems that perform face recognition tasks is face detection and facial feature extraction. Many algorithms have been proposed for face detection in still images that are based on texture, depth, shape and colour information. For example, face localization can be based on the observation that human faces are characterized by their oval shape and skin-colour [1]. Another very attractive approach for face detection relies on multiresolution images (also known as mosaic images) attempting to detect a facial region at a coarse resolution and, subsequently, to validate the outcome by detecting facial features at the next resolution [2, 3].

The major difficulties encountered in face recognition are due to variations in luminance, facial expressions, visual angles and other features such as glasses, beard, etc. This leads to a need for imposing several constraints on the set of images on which the specific algorithm is tested.

Experimental research through years proved that a fusion of techniques based on both grey level template matching and certain geometrical features usually provides more stable results than using each technique separately. A survey on different techniques can be found in [4]. Following a similar reasoning, we propose in this paper a mixture of complementary techniques that are based either on geometrical shape parameterization or on template matching or on dynamic deformation of active contours. The success of a specific technique depends on the nature of the extracted features. In the present paper, the features under consideration are the eyebrows, the eyes, the nostrils, the mouth, the cheeks and the chin. The eyes, the nostrils and the mouth are determined by searching for minima and maxima in the $x-$ and $y-$ projections of the greylevel relief [1]. In this paper, the following methods are proposed: i) the extraction of cheeks and chin by performing an Adaptive Hough Transform (AHT) on a relevant subimage defined according to biggest best-fit ellipse of the image, ii) the extraction of upper eyebrow edges using a template-based technique that adapts a greylevel mask to an area restricted by the position of the eyes, iii) the accurate extraction of the face contour by means of dynamic deformation of active contours (e.g. snakes), and iv) the determination of the face orientation by exploiting the face symmetry properties.

2 Face detection

As described in [1], skin-like regions in an image can be discriminated in the best way by representing the image in the HSV color space (or other similar color spaces like HSI or HLS) and by choosing proper thresholds for the values of the hue and saturation parameters. Considering the oval-like shape of a face in general, it is convenient to search for the connected components of the skin-like regions using a region growing algorithm and to fit an ellipse to every connected component of nearly elliptical shape. This restricts the search area for the algorithms that detect the various facial features, afterwards.

2.1 Parameteric feature extraction using AHT

In order to select an efficient set of distances, a parameterized representation of some facial features appears as a good approach towards the solution of the problem under study. Such an assumption holds for certain features that can be described in a realistic way by means of a geometric curve. The cheeks can reasonably be approximated by, for example, nearly vertical lines, and the chin by an upward parabola. A straightforward method to detect geometric curves in an image is the Hough transform or one of its variations [5]. The use of an adaptive Hough Transform is proposed in [6]. It is a variant of the Hough transform proposed in [7], which utilizes a small fixed size accumulator and follows a "coarse to fine" accumulation until a desired precision of the values for the parameters is reached. This variation requires a reduced storage space compared to the standard transform due to the small accumulator size. Accordingly, it provides a better performance than the standard transform due to the reduced number of computations. The search for cheek borders is based on the fact that they can be approximated by a straight line of the form:

$$y = mx + c \tag{1}$$

where m is the slope and c is the intercept of the line.

The accumulator array is thus two-dimensional, with each cell corresponding to a certain (m, c) pair of discrete values of the slope and of the intercept of the line. A small accumulator size of 9×9 is used. A line segment is assumed to be interesting, if its length is at least 10 pixels in the edge image obtained by applying a Sobel edge operator on the original image of size approximately 350×286 [8]. The cheek border corresponds to the dominant vertical line in the relevant subimage. Accordingly, the algorithm should search only for one line in the subimage. The main idea of the method is to focus on the parameter space on areas of large accumulation by performing iterations of the basic Hough algorithm and redefining the limits of the parameter ranges accordingly. If we let $a_i, 1 \leq i \leq n$ be the parameters representing the curve under consideration (in this case $n = 2, a_1 = m, a_2 = c$), then the search area in iteration i is:

$$S^{(i)} = L_1^{(i)} \times L_2^{(i)} \times \ldots \times L_n^{(i)} \tag{2}$$

where $L_j^{(i)}$ is an interval (U_j, V_j) of possible values of the jth parameter a_j.

In our case, the relevant subimage is defined as a certain region of the largest best-fit ellipse found in the image. This ellipse is assumed to contain the main face region. The success of a search inside the ellipse is ensured by the fact that the ellipse possesses the same vertical symmetry as the face region that it contains. Let a and b be the magnitudes of the major and minor semiaxes and (x_0, y_0) be the center of the best-fit ellipse. A suitable subimage that gives satisfactory results is defined as a portion of the lower half of the ellipse, i.e. between:

$$x_{min} = x_0 + 0.16 \cdot a \tag{3}$$

$$x_{max} = x_0 + 0.5 \cdot a. \tag{4}$$

A rectangular subimage is defined between x_{min} and x_{max} and between y_{min} and y_{max} given by:

$$y_{min} = y_0 - 0.86 \cdot b \tag{5}$$

$$y_{max} = y_0 - 0.55 \cdot b. \tag{6}$$

The left cheek is searched for inside the above described region. Similarly the right cheek is searched for inside a rectangular subimage defined between x_{min} and x_{max} and between y_{min} and y_{max} given by:

$$y_{min} = y_0 + 0.55 \cdot b \tag{7}$$

$$y_{max} = y_0 + 0.86 \cdot b, \tag{8}$$

respectively.

The selection of the above described areas of the ellipse ensures that some edge found in these subimages contains part of the cheek and does not contain other significant features, like ears.

In the case of not nearly frontal face images, the above mentioned symmetry does not hold and correcting moves of the relevant subimage along the horizontal direction can be useful in order to compensate for rotations along the x-axis.

The chin can be described by the equation of an upward parabola:

$$x - x_v = g(y - y_v)^2 \tag{9}$$

where (x_v, y_v) is the vertex of the parabola and g is the parameter which controls how fast the parabola opens outwards. In this case, the accumulator array is three-dimensional and thus $n = 3$ and $a_1 = g, a_2 = x_v, a_3 = y_v$ are the three different parameters of the parameter space. The size of the accumulator array chosen in our implementation is $6 \times 6 \times 6$. Again, a minimum segment length of 10 pixels is chosen in order to avoid erroneous results. The relevant subimage used for searching is defined in a similar way as in the case of cheeks. The limits of this search area are as follows:

$$x_{min} = x_0 + 0.62 \cdot a \tag{10}$$

$$x_{max} = x_0 + 0.85 \cdot a \tag{11}$$

$$y_{min} = y_0 - 0.53 \cdot b \tag{12}$$

$$y_{max} = y_0 + 0.53 \cdot b. \tag{13}$$

A compensation for rotations along the y-axis is provided by performing correcting moves along the vertical (major) axis of the ellipse.

2.2 Eyebrows extraction using template matching

Unlike the cheeks and the chin, the eyebrows do not possess a simple description by means of a geometric curve, because of their variations between different people. One aspect concerning eyebrows that is irrelevant to their specific shape is the uppermost point of the eyebrow. This implies that we have to detect the upper arch of the eyebrow. This edge constitutes a transition from a completely smooth area (the forehead) to an elongated approximately horizontal edge, as one is scanning the ellipse along the vertical axis. A template matching technique is more appropriate in this case. A prototype block has to be defined explicitly, simulating the form of the upper edge of an eyebrow, because no a priori information exists concerning the approximate position of the correct block. Let P be the predefined block (acting as a mask) and B_i be the real block of the image under consideration, where $i \in S$ with S being the set of all blocks centered on the search area. The block that should be chosen as optimal is a block B_j of maximal correlation to the prototype block.

The size of the block to be searched for should be normalized according to the scaling factor of the face in the image. A straightforward way to normalize the search process is to define the dimensions of the block based on the distance between the eyes. More specifically, a realistic consideration is to adapt a block of height equal to 0.125 of the eyes distance and of width equal to 0.5 of the eyes distance. The optimal block is expected to contain the upper edge of the eyebrow and can be used to provide a feature-point when defining the discriminating set of Euclidean distances.

2.3 Snake deformation for face contour definition

In many cases, a problem arises when there is no sufficient edge information in the edge image so that the AHT algorithm can find the straight lines representing the cheeks or the parabola representing the chin. This is usually due to the poor lighting conditions. A helpful technique to overcome this problem is to consider the best-fit ellipse surrounding the face as well as the ellipse of half-size axes and search for an active contour whose elements are constrained to lie on certain lines joining points between the two ellipses. Adjacent points on each ellipse have approximately equal Euclidean distances. In general, the energy of a snake with snake elements (snaxels) $v_i = (x_i, y_i)$, $i = 0, ..., M-1$, can be given by the equation:

$$E_{snake} = \sum_{i=0}^{M-1} E_{int}(v_i) + E_{ext}(v_i) \tag{14}$$

where E_{int} and E_{ext} denotes the internal and external energy terms of snaxels v_i.

The internal energy represents the forces acting on the contour and constraining it to be smooth. This is due to the continuity and curvature constraints that it imposes. The first and second derivative of the function representing the contour are considered to comprise this term. Using finite differences, this energy term can be approximated by:

$$\begin{aligned} E_{int}(v_i) &= w_1 \cdot (|v_i - v_{i-1}|^2 + |v_i - v_{i+1}|^2) \\ &+ w_2 \cdot |v_{i-1} - 2v_i + v_{i+1}|^2 \end{aligned} \tag{15}$$

The first order term ensures the string-like behaviour of the snake, and the second order term ensures the rod-like behaviour. The extent to which a snake is allowed to be stretched or bent is controled by the weighting functions w_1 and w_2. In our case, experiments proved that there was no significant improvement of the quality of the final face contour when an adaptive approach was used, so the weight factors were finally chosen to be constantly at the value one.

The exterior energy refers to the forces that pull the snake to features on the image that are considered in each case to be significant. In our case, the exterior energy consists of two terms. The first one corresponds to the magnitude of the output of a gradient operator (e.g. Sobel operator) and helps attract the snake to edges on the image. This is the main external energy term over regions where a transition from skin colour to background colour is dominant. The other external energy term refers to the degree of change of the Euclidean distance between adjacent pairs of candidate snaxels. This is to ensure that the chosen snaxels lie on a contour that is smooth enough to be used as the input for the next stage of the procedure. The exterior energy component of the snake energy can be written as:

$$E_{ext}(v_i) = E_{image} + E_{dist} \tag{16}$$

with the two terms being

$$E_{image} = -\left|\nabla I(x,y)\right| \tag{17}$$

where $I(x,y)$ denotes the grey level function of the original image and

$$E_{dist} = k \cdot \left|1 - \frac{d(v_{i-1}, v_i)}{d(v_i, v_{i+1})}\right| \tag{18}$$

where k is a normalizing factor and $d(v_i, v_{i+1})$ denotes the Euclidean distance between adjacent (candidate) snaxels.

A dynamic programming algorithm can be used for minimizing the energy of the snake. This method overcomes the problem of contour initialisation and thus is very useful in our case since no sufficient approximation exists as an input to the algorithm. The energy of each snake point v_i is given by:

$$\begin{aligned} E_{i-1}(v_{i-1}, v_i, v_{i+1}) &= \lambda_i E_{int}(v_{i-1}, v_i, v_{i+1}) \\ &+ (1-\lambda_i) E_{ext}(v_i) \end{aligned} \tag{19}$$

where $\lambda_i \in [0,1]$ is a regularisation parameter. A local evolutionary search over a window of small size is also introduced afterwards to provide us with a smoothened version of the estimated contour.

2.4 Determining face orientation

A relevant problem concerning face detection is the determination of face orientation. Once certain features of the face region, namely the eyes and the mouth, are correctly detected, and provided that no tilt about the $z-$ axis is present, a good estimation of an angle uniquely defining the face orientation can be obtained by exploiting face symmetry properties.

The three facial features determine an isosceles triangle. If we consider one side of the triangle to lie on the image plane, it is easy to compute a good estimate of the angle between the triangle plain and the image plain, by means of trigonometric functions of the distances between the three facial features under consideration.

In particular, let α be the side of the triangle that is considered to lie on the image plane and that corresponds to the distance between one of the eyes and the center of the mouth, and β be the orthogonal projection of the other side of equal length, on the image plane. This is explained in Figure 1 where the projection of the isosceles triangle is represented by continuous lines, whereas the isosceles triangle is represented by dotted lines. The dashed lines denote a triangle that lies on a plane that is perpendicular to both the image plane and the plane of the isosceles triangle. Let θ be the angle formed by line segments α and β, and ϕ be the angle between the image plane and the isosceles triangle plane, the latter can be computed using the following formula:

$$cos\phi = \frac{\beta sin\theta}{\sqrt{\alpha^2 - \beta^2 cos^2\theta}} \tag{20}$$

A rule-of-thumb to determine the orientation of the face is to look at the isosceles triangle as it is projected on the image plane: the biggest of the eye-to-mouth distances reveals the direction at which the person is staring. The success of the method is heavily dependent, as in the case of eyebrows detection, on the correctness of the eye detection.

3 Experimental Results

The methods proposed above where tested on a set of 37 images of the M2VTS project multimodal face database. The results were rather satisfactory, considering the fact that the face images have not been really frontal in some cases.

Figures 2(a) and 2(b) show examples of correct feature constellation results. The best-fit ellipse as well as the corresponding blocks where the cheeks and the chin were detected are indicated.

Table 1 shows the results of extraction of cheek borders, chin and eyebrows for the set of images mentioned above.

Problems have been encountered in cheek-extraction when the false symmetry of the ellipse leads to bad definition of the relevant subimage, and thus to an erroneous extraction of some other feature considered as predominant. Similar problems cause the AHT to fail in the case of chin detection. The inability of the edge operator to detect weak edges caused by bad luminance, is more obvious here.

The correct extraction of the eyebrows is heavily dependent on the position of the eyes, as one would expect. If at least one eye is detected at a wrong position, and especially when the distance between the eyes is bigger than the actual one, the size of the block to be matched is incorrect. This leads to the false extraction of another feature. Even when eyes are correctly extracted, a problem may arise if hair covers the forehead and the eyebrows, as is shown in Figure 2(c). Another reason for a reduced detection rate is the false definition of the search area, resulting from a bad definition of the best-fit ellipse. The rate of correct eyebrows detection increases to about 95% if we consider the set of images where correct eyebrows detection has been performed.

In cases of images that suffer from a lack of edge content, the dynamic programming algorithm succeeds in providing additional information for the later definition of feature distances. This is illustrated in Figure 3. Both the inner and the best-fit ellipses are shown in these figures. The method fails only in extreme cases, when the main connected component of the image contains features that deform the actual face region dramatically (e.g., when no clear separating line between the hair and the forehead exists).

Finally, the face orientation determination technique led to correct results in all cases where eyes and mouth were extracted correctly. Figure 4(a) shows an example of a person looking to the left at an angle of 18.467^0 compared to a frontal picture, Figure 4(b) shows a person looking to the left at an angle of 8.062^0, and Figure 4(c) shows a person looking to the right at an angle of 19.115^0. The white triangle in these pictures corresponds to the projection of the isosceles triangle on the image plane.

4 Conclusions

A combination of methods based on adaptive Hough transform, block matching and active contour deformation variations are proposed in order to provide sufficient feature information to be used in the definition of a set of geometric distances for face recognition purposes. A geometric method for computing the face orientation based on the symmetric properties of certain facial features is also proposed. Examples demonstrating the success of the different methods, as well as criticism when they fail have been included.

References

[1] K. Sobottka and I. Pitas, "A novel method for automatic face segmentation, facial feature extraction and tracking", *Image Communication, Elsevier*, accepted for publication, 1997.

[2] G. Yang and T.S. Huang, "Human face detection in a complex background", *Pattern Recognition*, 1994, **27**(1), pp. 53-63.

[3] C. Kotropoulos and I. Pitas, "Rule-based face detection in frontal views", in *Proc. of the IEEE ICASSP '97*, April 1997, Munich, Germany, pp. 2537-2540.

[4] R. Chellapa, C.L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey", *Proc. of the IEEE*, 1995, **83**(5), pp. 705-740.

[5] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, vol. I., Addison-Wesley, Reading, Massachussets, 1992.

[6] X. Li and N. Roeder, "Face contour extraction from front-view images", *Pattern Recognition*, 1995, **28**(8), pp. 1167-1179.

[7] J. Illingworth and J. Kittler, "The Adaptive Hough Transform", *IEEE Trans. on PAMI*, 1987, **9**(5), pp. 690-698.

[8] I. Pitas, *Digital Image Processing Algorithms*, Prentice Hall, UK, 1993.

features	detected [%]	correctly [%]
left cheek border	100	78
right cheek border	100	86
chin	100	65
left eyebrow	100	73
right eyebrow	100	81

Table 1: Detection rates for facial feature extraction

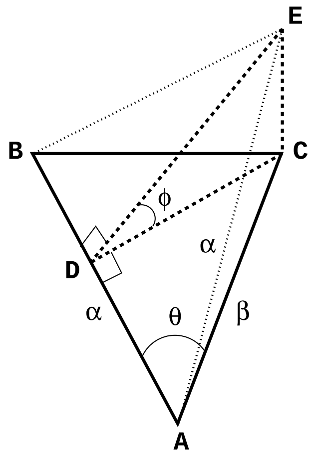
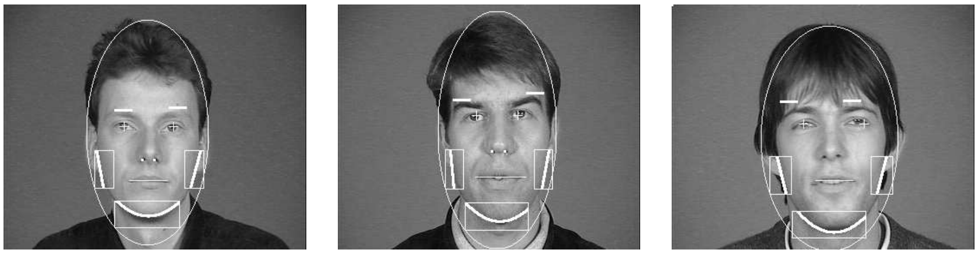


Figure 1: Geometry of projected isosceles triangle.



(a) (b) (c)

Figure 2: Examples of results for facial features.

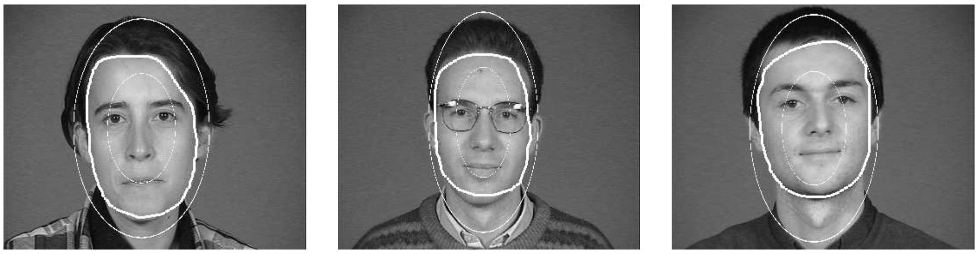
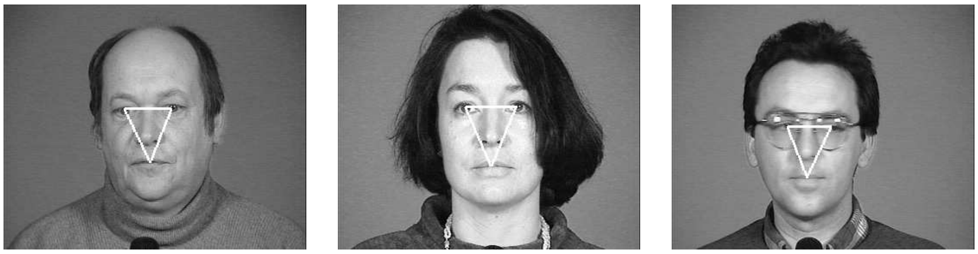


Figure 3: Examples of results for face contour.



(a) (b) (c)

Figure 4: Examples of results for face orientation.