



PII: S0031-3203(97)00048-4

FACIAL FEATURE DETECTION USING GEOMETRICAL FACE MODEL: AN EFFICIENT APPROACH

SHI-HONG JENG,* HONG YUAN MARK LIAO,^{††} CHIN CHUAN HAN,[†]
 MING YANG CHERN* and YAO TSORNG LIU[‡]

*Institute of Electrical Engineering, National Chung Cheng University, Chia Yi, Taiwan

[†] Institute of Information Science, Academia Sinica, 128 Sinica Road, Section 2,
 Nankang 11529 Taipei, Taiwan

[‡]Department of Statistics, University of Illinois, Urbana-Champaign, Illinois, U.S.A.

(Received 25 April 1996; accepted 11 February 1997)

Abstract—A useful geometrical face model and an efficient facial feature detection approach are proposed. Based on the fact that human faces are constructed in the same geometrical configuration, the proposed approach can accurately detect facial features, especially the eyes, even when the images have complex backgrounds. The average computation time for one image of size 512×340 is less than 5 s by a SUN-Sparc 20 workstation. Experimental results demonstrate that the proposed approach can efficiently detect human facial features and satisfactorily deal with the problems caused by bad lighting condition, skew face orientation, and even facial expression. © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Geometrical face model

Facial feature detection

Face recognition

Image processing

1. INTRODUCTION

When we talk to a person, we usually look at his/her face, especially his/her eyes. This observation is supported not only by our experience but also by a number of psychological experiments. The expression of a person's face plays a very important role when communicating with other people. Due to the uniqueness, face is also the most important and effective characteristic for recognizing a person. Compared with fingerprints or retinas, taking a face image from a person is very easy. Therefore, face recognition has become one of the most popular applications in the field of computer vision.

Face recognition requires two major steps. In the first step, faces have to be detected from a scene, either simple or cluttered. Then, the detected faces have to be normalized and recognized by a specially designed classifier. There were a good number of works related to face recognition.^(1–4) Among them, Turk and Pentland [4] have successfully employed the eigenface approach to recognize human face. However, an accurate and efficient method for human face detection is still deficient. Some previous methods relied for their detection on the correlation techniques.^(5, 6) Although this technique can reach better accuracy, the tradeoff is that it is a very time-consuming task. Song, Lee and Tsuji⁽⁷⁾ used a partial feature template and a global facial model to extract facial features, but their approach is heavily affected by a thresholding process and still time-consuming. Govindaraju, Sher and

Srihari⁽⁸⁾ and Govindaraju, Srihari and Sher⁽⁹⁾ presented a system which could locate human faces in photographs of newspapers, but the approximate size of faces and the expected number must be known in advance. Sirohey⁽¹⁰⁾ utilized an elliptical structure to segment the human heads from the gray-scale cluttered images. An ellipse was fitted the human head from the edge map image. His approach is heavily affected by the result of edge detection. Yang and Huang⁽⁶⁾ utilized a three-level hierarchical knowledge-based method to locate the human faces in the complex background. Two sets of rules were applied to the mosaic images at various scales to locate all possible candidates of faces in the two higher levels. At level three, the eyes and mouth regions can be found and precisely located via an improved edge detection scheme. Nguyen and Huang⁽¹¹⁾ collected a set of image operators to analyze the face image. These operators with some specific criteria were extensively tested and some superior results were obtained. Kosagi⁽¹²⁾ was another one who proposed to detect the human face region from the multi-scale mosaic images. His algorithm first found the head regions by matching the input mosaic image and six clustered mosaic face patterns. Then, the precise face location was located via the gray-scale histograms of eye and nose regions. Burel and Carel⁽¹³⁾ used a multi-layer perceptron (MLP) classifier to decide if a sliding window contains a face or not. Sung and Poggio⁽¹⁴⁾ used two distance metrics to measure the distance between the input window image and cluster center. Twelve clusters including six face and six non-face clusters were trained by a modified *k-mean* clustering

^{††}Author to whom correspondence should be addressed.

technique. A feature vector consisting of 12 values (the distances between the window image and the 12 cluster centers) was inputted a multi-layer perceptron network for the desired verification task. Rowley Baluja and Pentland⁽¹⁵⁾ presented another neural network-based detection system to find the location of frontal face. Pentland, Moghaddam and Starner,⁽¹⁶⁾ Moghaddam and Starner^(17, 18) and Darrel, Moghaddam and Pentland⁽¹⁹⁾ used PCA to describe the face patterns with lower-dimensional features. They designed a distance function called *distance from feature space* (dffs) as a metric to evaluate the difference between the input face image and the reconstructed face image. In reference⁽²⁰⁾ Chow and Li used morphological filtering and 8-connected blob coloring to find the intensity valley regions for the matching of face template in low resolution. The fine features such as the locations and shapes of eyes or mouth were analyzed in high resolution based on the Hough transform and the deformable template methods.

Face detection can also be achieved by detecting geometrical relationships among facial organs such as nose, eyes, and mouth.^(21–23) Juell and Marsh⁽²¹⁾ proposed a hierarchical neural network to detect the human face in gray-scale images. An edge enhancing pre-processor and four backpropagation neural networks arranged in a hierarchical structure were devised to find multiple faces in the scene. Burl and Perona⁽²²⁾ coupled a set of local feature detectors via a statistical model to find the facial components for face locating. Besides, Jeng, Mark Liao, Liu and Chen⁽²³⁾ also proposed a fast face detection approach based on a geometrical face model. Matsuno, Kimura and Tsuji⁽²⁴⁾ presented the facial expression as the deformation of a 2-D net called *potential net* to detect the human faces. They first coarsely located the positions of faces from the mosaic images via the potential net. Then, the precise locations of faces were determined by analyzing the horizontal and vertical histograms of edges. Instead of the gray-scale images, face detection can also be achieved by locating facial features or facial organs from color images.^(25–26)

In this paper, we propose an efficient approach for facial feature detection. The proposed approach detects facial features by using a geometrical face model constructed according to the relative geometrical relation among facial organs on a face. Due to the merit of this face model, some essential problems which have to be solved in facial feature detection process, such as facial expression and face orientation, can be overcome easily. The proposed approach has been shown to be very efficient and accurate by experimental results. The rest of this paper is organized as follows. In Section 2, an overview of the proposed human face detection system and several commonly encountered problems will be introduced. The corresponding strategies or constraints for each problem will also be presented in this section. The details of the facial feature detection system will be described in

Section 3. Some experimental results will be demonstrated in Section 4 to corroborate the proposed approach. Conclusions and future work will be given in Section 5.

2. FACE DETECTION SYSTEM

In this section, we will overview the face detection system and address some problems which might be encountered. These problems include face size, face orientation, lighting condition, covering, and facial expression. For each problem, a strategy or a reasonable constraint will be introduced to solve it.

2.1. Overview

Before building a face detection system, it is very useful if we know how eyes function. In general, eyes can detect a wide variety of objects under various environment. However, if an object appears at a particular pose, e.g., upside down, we may not recognize it at once. The reason is the orientation of this object is different from that of its counterpart in memory. Therefore, when one wants to check what exists in an image, usually the whole geometrical configuration is checked instead of checking object's details. Besides, since a face contains four main organs, i.e., eyebrows, eyes, nose, and mouth, it is very important to detect these facial features so that the whole geometrical configuration of the face can be identified. In general, eyebrows, eyes, nostrils and mouth always look darker than the rest of a face due to illumination. Among them, the darkness of eyebrows also depends on their density and color, and the nostrils may not always be seen. Hence, in our system, eyes and mouth are considered as primary features, while eyebrows and nostrils are considered as secondary features. Based on the aforementioned observation, the procedure for developing our face detection system can be described as follows.

When a face image is fed into the system, a pre-processing step will be applied to remove small light details and to enhance the contrast. Then, the processed image will be thresholded to produce a binary image. A processed binary image is a sketch-like image that preserves most of the important information including facial features. Then, a labeling step and a grouping algorithm will be used to group detected features block by block. These feature blocks are regarded as *facial feature candidates*. Finally, we will choose a set of blocks from those previously detected feature blocks and use these blocks as candidates to check with a geometrical face model. The geometrical face model is constructed based on the real proportion between each facial organ. Since the feature blocks used are all potential candidates of a face, our face detection system is much more efficient than other techniques. Fig. (1) shows the main steps covered in the proposed system.

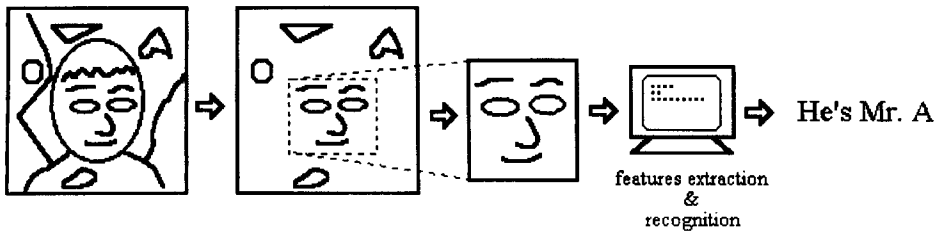


Fig. 1. Illustration of the main steps of the proposed face recognition system.

2.2. Problems and Strategies

Before the real implementation of our face detection system, some problems must be considered and solved. In what follows, a complete discussion on some basic problems in face detection will be given. The solution for each problem will be provided as well.

2.2.1. Size and orientation. Due to the imaging distance and direction, face size and orientation are varying from one image to another. Figure (2) illustrates this concept. Since our detection scheme is based on the relative geometrical relation between each facial feature, small variations of face size and orientation can be handled by our scheme effectively. In addition, a detected face is generally used for recognizing its identity. If a detected face is too small or in a too skew orientation, it is very difficult to recognize it by a machine. Hence, some reasonable constraints on face size and orientation are required. For an image of size 512×340 , the size of a detected face is expected to be not smaller than 80×80 . As to the constraints of face orientation, since our geometrical face model is a frontal view model, a face in a too skew orientation will not be well-fitted with this model. Therefore, the face orientations with respect to the x -axis and the z -axis are constrained, respectively, to the range from -30° to 30° and -45° to 45° (see Fig. 3). The face orientation with respect to the y -axis is confined to the range such that both eyes can be seen and neither appears on the face contour. However, the rotation constraint with respect to the z axis can be removed if the consideration of the image rotation is included.

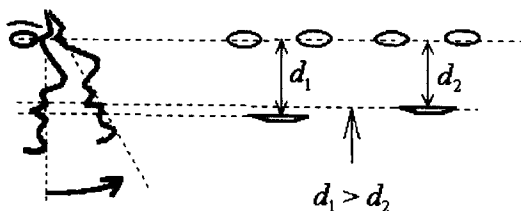


Fig. 2. Variations of face size and orientation.

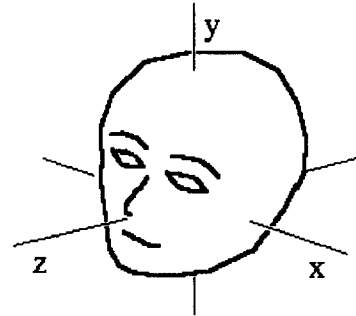


Fig. 3. Definitions of the three principal axes.

2.2.2. Lighting condition. The brightness and clearness of an image depend heavily on the lighting condition. An inadequate illumination possibly makes the grouping process fail to generate correct and necessary feature blocks. Thus, the success rate of the proposed system will be reduced. In our detection scheme, the preprocessing step, including an opening operation and a high-boost filtering, can effectively reduce the effect of this problem.

2.2.3. Covering. Sometimes facial features may be covered by other things, such as a hat, a glasses, or hairs. Most of the existing face detection systems cannot handle this problem well. Because the final matching scheme of our detection system is based on the two eyes of a face (see Section 3.3 for details), another necessary constraint is required, i.e., neither of the two eyes can be covered.

2.2.4. Facial expression. Human faces convey a variety of emotions by many different expressions. Although facial expressions may cause variations of facial features, the relative geometrical relation between each facial organ is still unchanged. Our detection scheme is based on the fixed relative geometrical relation among different facial organs, therefore, the problem of facial expression can be handled satisfactorily.

3. THE PROPOSED APPROACH

In this section, we shall describe in detail about the detection of facial features. Our approach can be

divided into three main steps. The first step is preprocessing. The goal of this step is to get rid of high intensity noises and to transform the input image into a binary one. The second step includes a labeling process and a grouping process. This step tries to generate facial feature candidates block by block. Finally, a geometrical face model is used to locate the actual position of a face.

3.1. Preprocessing

As mentioned in Section 2.1, eyebrows, eyes, nose, and mouth are the most important facial features on a human face. Replacing nose by nostrils, those facial organs, in general, appear as dark features in a face image. In order to prevent facial features from breaking into pieces, sharp peaks and bright noises, such as reflections on eyes or spectacles, must be eliminated in advance. An *opening operation* is employed on the input image to achieve this purpose. The results after morphological opening are shown in Fig. 4. The right image is the opening of the left image in each image pair. These results show that reflections on eyes and spectacles will be removed and all dark facial features will be preserved after performing the opening operations.

In order to properly generate facial feature candidates, the unnecessary information in a face image must be removed in advance. For this purpose, a *boost-filtering*⁽²⁷⁾ is used to enhance the contrast between facial features and the rest of a face image. The mask used for high-boost filtering is shown in Fig. 5. To prevent the side-effect of over-enhancing background noises, a gradient operator is introduced for computing the center weight of the high-boost mask. The center weight w at location (x, y) can be calculated by the following functions:

$$s = \frac{\text{Sobel}(x, y)}{255},$$

$$g = \frac{\text{Gray}(x, y)}{255},$$

$$z = \begin{cases} 0.4 & \text{if } s > 0.5, \\ 0.8 & \text{if } s \leq 0.5, \end{cases}$$

$$w = [z(1 - s) + (1 - z)(1 - g)] \times (f - 8) + 8,$$

$$\frac{1}{9} \times \begin{bmatrix} -1 & -1 & -1 \\ -1 & w & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Fig. 5. A high-boost mask.

where s is the normalized gradient value of the *Sobel operator*⁽²⁷⁾ at (x, y) , g is the normalized *gray level* at (x, y) , z adjusts the weights of w to prevent the over-enhancing of background noises according to the gradient value, and f is a boosting factor and usually set to be 11 for most lighting conditions. The center weight w depends on both the gradient value and the gray level. Its value is always greater than 8.

Finally, an image after high-boost operation will be binarized into a sketch-like image by a thresholding process. The images in Fig. 6 demonstrate the results after performing these two processes. Fig. 6(b) is a high-boosted image of the original image shown in Fig. 6(a). Figure 6(c) shows the sketch-like image resulted from binarizing the high-boosted image. Compared with the similar result shown in Fig. 6(d), which is the image obtained after performing the Sobel operation, it is obvious that the image produced by our method can preserve all necessary facial features, especially the eyes.

The dark regions in a sketch-like binary image consist of all active pixels which include facial features. The so-called *run-length local table method*⁽²⁸⁾ is then employed to group the active pixels into maximal connected blocks. This labeling algorithm makes only two passes, a top-down pass and a bottom-up pass, through the binary image. After the labeling process, for each feature block \mathcal{B} containing pixels, its center of mass (\bar{x}, \bar{y}) and orientation θ can be obtained as follows:⁽²⁹⁾

$$\bar{x} = \frac{1}{N} \sum_{(x,y) \in \mathcal{B}} x,$$

$$\bar{y} = \frac{1}{N} \sum_{(x,y) \in \mathcal{B}} y,$$

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \right),$$

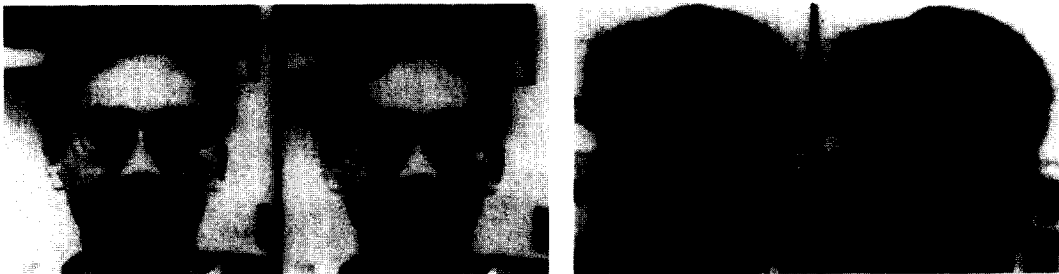


Fig. 4. Results of the opening operation.

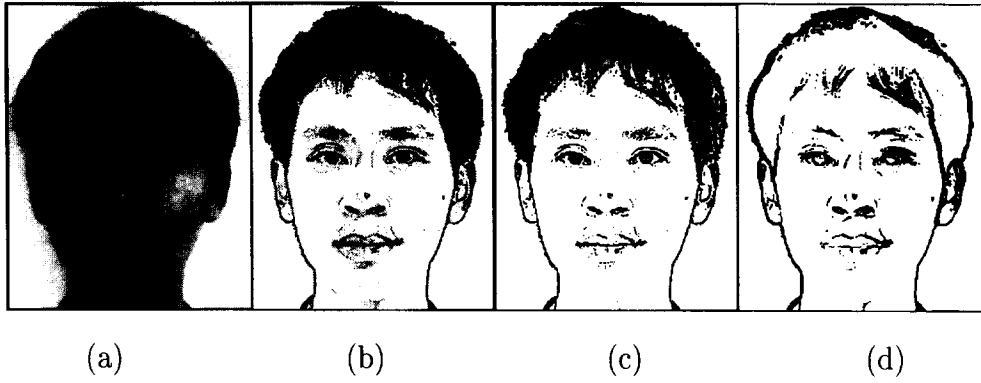


Fig. 6. Results after performing the preprocessing step.

where $\mu_{p,q}$ is the (p,q) order central moment and can be calculated by $\mu_{p,q} = \sum_{(x,y)} \sum_{\mathcal{B}} (x - \bar{x})^p (y - \bar{y})^q$. Once the orientation is known, the *bounding rectangle* and the length of semimajor axis for each feature block can also be obtained.⁽²⁹⁾

A potential problem required to be handled is that some feature blocks belonging to the same facial feature are still separate after the labeling process. If a facial feature is split into different feature blocks, its size (total number of pixels), center, and orientation will be changed. This outcome will certainly affect the accuracy of the matching process. Therefore, a grouping algorithm is proposed to merge these split feature blocks. This algorithm contains two characteristics: (a) the merging radius depends on the size of a feature block under consideration (the maximal block size, N_{\max} , considered for merging is preset to be 30), in other words, the larger the block size is, the smaller the merging radius is; (b) two feature blocks which are to be merged must lie on the same axis if either of the block sizes is larger than N_{\max} . In what follows, the grouping algorithm is given. $N(i)$ represents the total number of pixels of the i th block \mathcal{B}_i , and N_b denotes the total number of feature blocks in the image.

Procedure Grouping

For $N := 1$ **to** N_{\max} **do**

$$r := 8 - 6 \times \frac{N - 1}{N_{\max} - 1}$$

for $i := 1$ **to** N_b **do**

if $N(i) = N$ **then**

“Find all blocks $\{\mathcal{B}_j\}$ which are within the circle range of \mathcal{B}_i with radius r .”

if $\{\mathcal{B}_j\} \neq \emptyset$ **then**

“Find the nearest block \mathcal{B}_k from $\{\mathcal{B}_j\}$.”

if $(N(k) < N_{\max} \text{ or } (\mathcal{B}_k \text{ lies on the same axis of } \mathcal{B}_i))$ **then**

“Merge \mathcal{B}_i with \mathcal{B}_k ”

end for

end for

end Grouping

Figure (7) illustrates an example showing how the grouping algorithm works. In the beginning, \mathcal{B}_1 is the

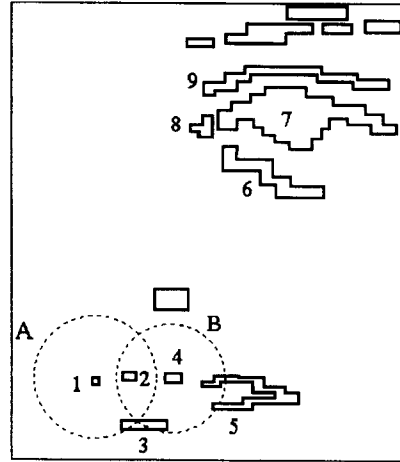


Fig. 7. A grouping example.

smallest block, hence it is merged with its nearest block, \mathcal{B}_2 . The size, center, orientation, and bounding rectangle of the merged block are recomputed. Then, the smallest block, \mathcal{B}_4 , is merged with its nearest block, \mathcal{B}_5 . The basic information is also recomputed. This grouping process proceeds until no blocks can be merged or the size of a block under consideration is larger than a preset value, N_{\max} (which is set according to the image size). Note that the merging radius decreases as the block size increases. Before any two feature blocks can be merged, the requirement of characteristic (b) must be satisfied. Therefore, \mathcal{B}_8 can be merged with \mathcal{B}_7 , whereas \mathcal{B}_6 and \mathcal{B}_9 cannot.

3.2. Matching

After the labeling and grouping processes, all feature blocks are regarded as facial feature candidates. Under the constraints of face pose mentioned in Section 2.2, all faces in the input images generally have the same geometrical configuration. According to the average proportion between each facial organ obtained by estimating several real faces, a geometrical face model can be constructed to evaluate

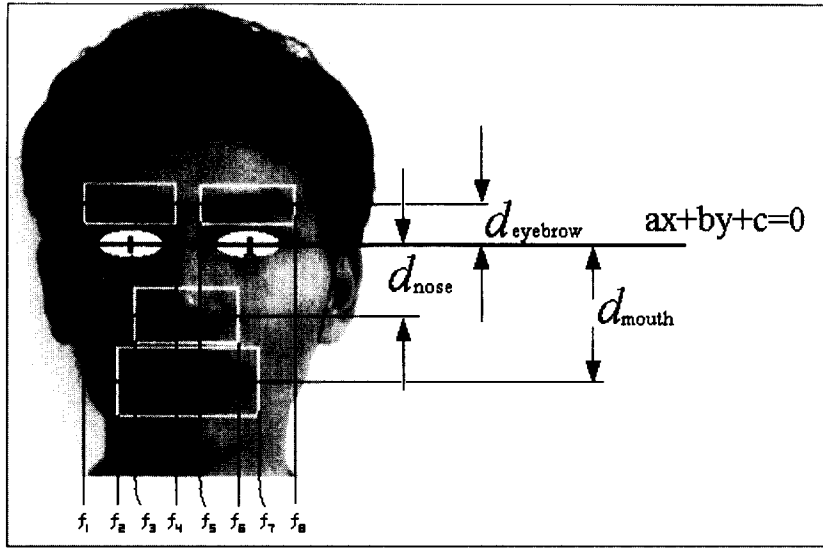


Fig. 8. The geometrical face model.

which combination of feature blocks is a face (see Fig. 8).

In Fig. 8, the line passing through the centers of both eyes is called the *base line*. Let (x_1, y_1) and (x_2, y_2) be, respectively, the centers of right eye **A** and left eye **B**, then the coefficients of the base line $ax + by + c = 0$ and the angle θ between the base line and the x-axis can be calculated as follows:

$$a = y_2 - y_1,$$

$$b = x_1 - x_2,$$

$$c = x_2 y_1 - x_1 y_2,$$

$$\theta = \tan^{-1}\left(-\frac{a}{b}\right), \quad -\frac{\pi}{2} \leq \theta < \frac{\pi}{2}.$$

The four white rectangles indicate the regions, in which other facial features possibly appear, with respect to the known locations of two eyes. The notations D_{nose} , $D_{eyebrow}$, and D_{mouth} represent the distances from the centers of nose, eyebrows, and mouth to the base line, respectively. Let $D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ be the distance between two centers of the eyes, and the distances D_{nose} , $D_{eyebrow}$, and D_{mouth} in the geometrical face model be, respectively, estimated with $0.6D$, $0.3D$, and $1.0D$. The constants c_i , $i = 1, 2, \dots, 8$ are the constant terms of the line equations $bx - ay - c_i = 0$, which are the boundary lines of those white rectangles that are perpendicular to the base line.

The matching process starts by randomly selecting two facial features as the eyes. Their corresponding base line as well as the angle θ can be obtained by previous formula. Due to the constraints of face orientation, the slope angle must be checked to see whether it is within the range of -45° to 45° . If not,

the current pair will be abandoned and another pair will be considered. In this step, most of the combinations of facial feature candidates will be rejected. For those survived pairs, their orientations, θ_1 and θ_2 , and “normalized” lengths of semimajor axes, l_1 and l_2 , are calculated. The normalized length of semimajor axis is obtained by dividing the original length by the distance between the two blocks, i.e., D . Then, the following evaluation function will be employed to evaluate whether the current block pair is the eyes or not.

$$E_{eye} = \exp[-1.2 \times ((l_1 - l_2)^2 + (l_1 + l_2 - 1)^2 + (\theta_1 - \theta)^2 + (\theta_2 - \theta)^2)].$$

In the exponent of this evaluation function: the first term $(l_1 - l_2)^2$ accounts for the length between two eyes should be kept similar; the second term $(l_1 + l_2 - 1)^2$ enforces the distance between two eyes should be about twice the length of each eye; the last two terms $(\theta_1 - \theta)^2$ and $(\theta_2 - \theta)^2$ enforce that both eyes should align with the base line. This evaluation function ranges from 0 to 1. The larger value the evaluation function gets, the higher probability these two feature blocks are the eyes. If the evaluation value is larger than a threshold, ex., 0.7, these two feature blocks are accepted as *eye candidates* and the matching process proceeds by evaluating other facial features. This process continues until all potential pairs are tested.

For each pair of eye candidate, the relative regions in the geometrical face model will be searched for other facial features. In order to achieve this goal, every feature block located within these relative regions must be evaluated to decide whether it is a real facial feature or not. In order to make things clear, we use mouth feature as an example. Let the center of a feature block k be located at (\bar{x}_k, \bar{y}_k) . The

distance from the center to the base line can thus be calculated as

$$d_{mouth} = \frac{|a\bar{x}_k + b\bar{y}_k + c|}{\sqrt{a^2 + b^2}}.$$

Because the center of the mouth feature must lie in the area between the boundary lines $bx - ay + c_2 = 0$ and $bx - ay + c_7 = 0$, the condition $(b\bar{x}_k - a\bar{y}_k + c_2)(b\bar{x}_k - a\bar{y}_k + c_7) < 0$ is required. Then, the following evaluation function can be employed to evaluate the probability of the feature block k being the mouth feature.

$$E_{mouth} = \exp \left[-4.8 \times \left(\frac{d_{mouth} - D_{mouth}}{D} \right)^2 \right].$$

In the exponent of this evaluation function, the term $(d_{mouth} - D_{mouth}/D)^2$ accounts for the distance from the mouth center to the base line should be appropriately constrained according to the geometrical face model. The divisor D and the outer multiplier -4.8 make the evaluation value have an equal influence as that of the eye feature. This evaluation function also ranges from 0 to 1. After the evaluation value for each feature block is obtained, the feature block having the largest value will be regarded as the mouth feature corresponding to the current pair of eye candidates. The corresponding facial features of right eyebrow, left eyebrow, and nose can be obtained accordingly. Their evaluation functions are given as follows:

$$E_{Reb} = \exp \left[-4.8 \times \left(\frac{d_{Reb} - D_{eyebrow}}{D} \right)^2 \right],$$

$$E_{Leb} = \exp \left[-4.8 \times \left(\frac{d_{Leb} - D_{eyebrow}}{D} \right)^2 \right],$$

$$E_{nose} = \exp \left[-4.8 \times \left(\frac{d_{nose} - D_{nose}}{D} \right)^2 \right].$$

After the above evaluation processes have been finished for each pair of eye candidates, an overall evaluation function is employed to decide which combination of feature blocks is most likely to be a face. The overall evaluation function is defined as the weighted sum of the evaluation values of each facial feature:

$$E = 0.5E_{eye} + 0.2E_{mouth} + 0.1E_{Reb} + 0.1E_{Leb} + 0.1E_{nose}.$$

Since every evaluation value ranges from 0 to 1, they all have equal influences. Since the eyes are considered the most important facial features, the weight for eye feature is set larger than others. Mouth in our system is considered the second important facial feature, it is assigned a second largest weight. The weights for other facial features, eyebrows and nose, are all set to be 0.1.

4. EXPERIMENTAL RESULTS

In order to prove the efficiency and accuracy of our system, we used a number of face images taken under roughly controlled imaging conditions as test images. 114 test images were selected to test the validity of our system. All test images were of size 512×340 . In these test images, all human faces were oriented into various directions and positioned arbitrarily in cluttered background. As we mentioned before, the minimum size of a face which could be detected was larger than

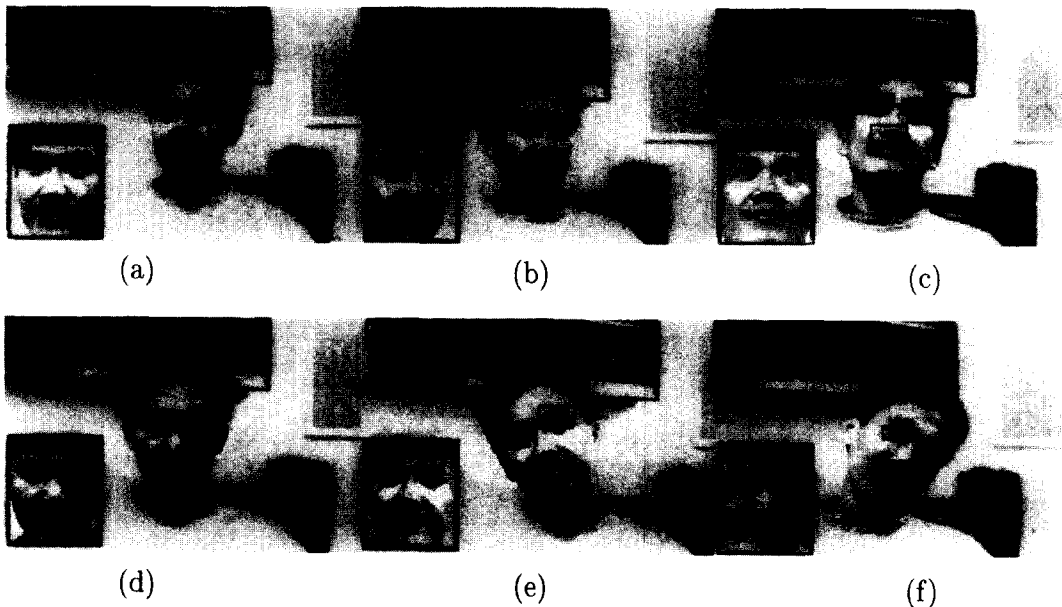


Fig. 9. Experimental example I: (a) turning right, (b) turning left, (c) turning up, (d) turning down, (e) tilting right, and (f) tilting left.

80×80. The computer we adopted to measure the performance of the proposed approach is a SUN-Sparc 20 workstation.

Some experimental results in which faces were with or without spectacles at various poses and orientations were shown in Figs 9–12. In these figures, all detected facial features were drawn with small black rectangles. In Figs 9 and 10, faces were turning right,

turning left, turning up, turning down, tilting right, and tilting left. Besides, the detected faces in the small windows of Fig. 9 were already normalized to the standard size and orientation by a scaling and a rotation. In addition to the orientation and position problems, our proposed algorithm can also solve the scaling and spectacled problems as shown in Figs 11 and 12. For an overall evaluation, 98 faces were detected

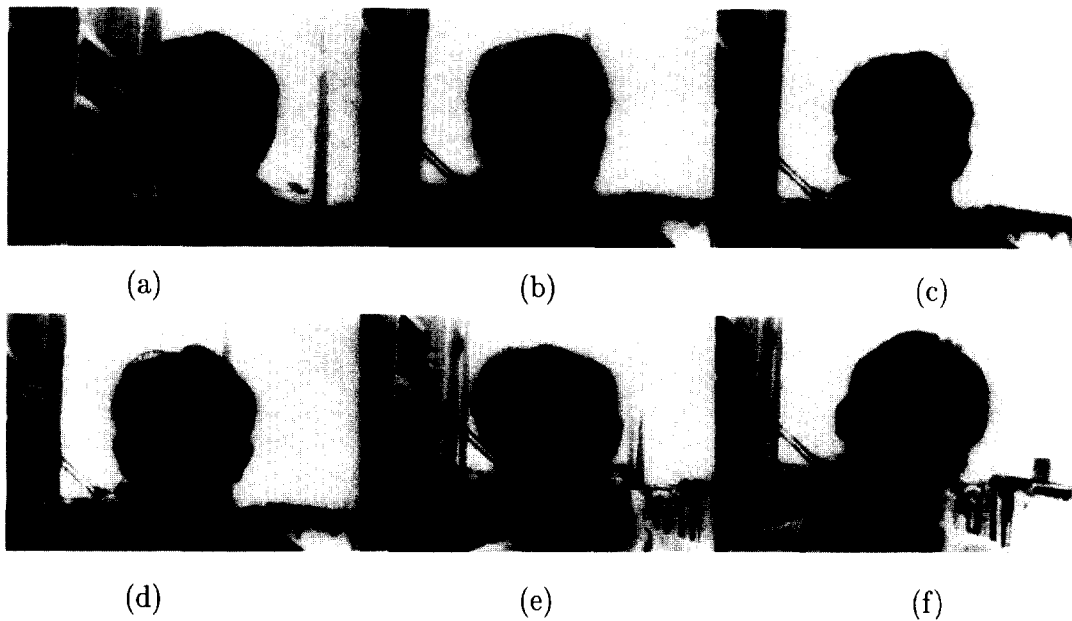


Fig. 10. Experimental example II: (a) turning right, (b) turning left, (c) turning up, (d) turning down, (e) tilting right, and (f) tilting left.

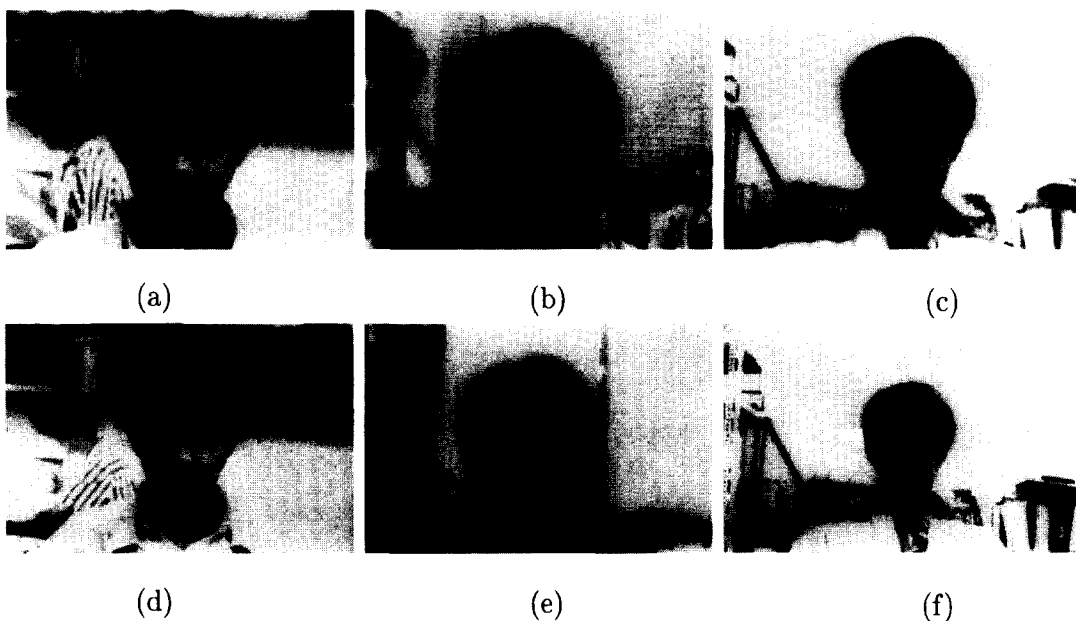


Fig. 11. Experimental example III: (a)–(c) Scaling up, (d)–(f) scaling down.

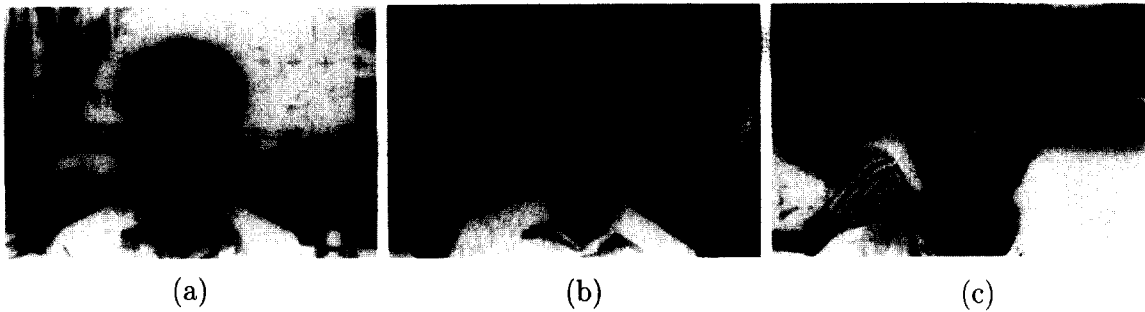


Fig. 12. Experimental example IV : spectacled faces.



Fig.13. Experimental example V : mis-detected faces.

successfully out of the total of 114 faces. Therefore, the success rate for detecting face was roughly about 86%. On the other hand, Fig. 13 shows two unsuccessful detections. In these examples, the face size was smaller than 80×80 in Fig. 13(a) and multiple faces existed in Fig. 13(b).

As to the execution time problem, the average computation time for one image was less than 5 s. Most part of the computation time, about 70%, were consumed by the preprocessing step, i.e., the morphological opening and the high boosting. It is obvious that our approach can accurately and efficiently detect facial features, especially the eyes, even when the images have complex backgrounds.

5. CONCLUSIONS

In this paper, we have proposed an efficient face detection approach based on a geometrical face model. First of all, a morphological opening and a high-boost filtering operations are performed to prevent facial features from breaking into pieces and to enhance the contrast between facial features. Next, all feature blocks are regarded as facial feature candidates after the labeling and grouping processes. Based on the configuration of facial organs, the most possible set of facial blocks is chosen to be the detected face. The average execution time needed is less than 5 s for an image of size 512×340 . At current stage, our detection approach simply chooses the most possible

set of feature blocks to be the detected face and assumes only one face in the image. If there are multiple faces or no face in the image, the mis-detected faces or a wrong face will be detected. In the future, we will improve the detection approach to detect all faces from the cluttered images.

REFERENCES

1. M. A. Shackleton and W. J. Welsh, Classification of facial features for recognition, *Proc. Computer Vision and Pattern Recognition*, pp. 573–579. Lahaina, Maui, Hawaii (1991).
2. B. S. Manjunath, A feature based approach to face recognition, *Proc. Computer Vision and Pattern Recognition*, pp. 373–378 Champaign, Illinois (1992).
3. R. Brunelli and T. Poggio, Face recognition: Features versus templates, *IEEE Trans. Pattern Analysis Mach. Intell.* **15**, 1042–1052 (1993).
4. M. Turk and A. Pentland, Eigenfaces for recognition, *J. Cognitive Neuroscience* **3**, 71–86 (1991).
5. A. Tsukamoto, C. W. Lee, and S. Tsuji, Detection and tracking of human face with synthesized templates, *Proc. Asian Conf. on Computer Vision*, pp. 183–186 (1993).
6. G. Yang and T. S. Huang, Human face detection in a complex background, *Pattern Recognition* **27**, 53–63 (1994).
7. X. Song, C. W. Lee, and S. Tsuji, Extraction facial features with partial feature template, *Proc. Asian Conf. on Computer Vision*, pp. 751–754 (1993).
8. V. Govindaraju, D. B. Sher, and R. K. Srihari, Locating human face in newspaper photographs, *Proc. Computer Vision and Pattern Recognition*. San Diego, California (1989).

9. V. Govindaraju, S. N. Srihari, and D. B. Sher, A computational model for face location, *Proc. Computer Vision and Pattern Recognition*, pp. 718–721 (1990).
10. S. A. Sirohey, Human face segmentation and identification, Master's thesis, University of Maryland (1993).
11. T. Nguyen and T. Huang, New robust image operations and applications in automatic facial feature analysis, *Proc. Int. Conf. on Image Processing*, pp. 524–527. Washington, District of Columbia (1995).
12. M. Kosugi, Human-face search and location in a scene by multi-pyramid architecture for personal identification, *Systems Comput. in Japan* **26**, 27–38 (1995).
13. G. Burel and D. Carel, Detection and localization of faces on digital images, *Pattern Recognition Lett.* **15**, 963–967 (1994).
14. K. K. Sung and T. Poggio, Example-based learning for view-based human face detection, *Proc. Image Understanding Workshop*, pp. 843–850. Monterey, California, (1994).
15. H. A. Rowley, S. Baluja, and T. Kanade, Neural network-based face detection, *Proc. Computer Vision and Pattern Recognition*, pp. 203–208. San Francisco, California (1996).
16. A. Pentland, B. Moghaddam, and T. Starner, View-based and modular eigenspaces for face recognition, *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 84–91. Seattle, Washington (1994).
17. B. Moghaddam and A. Pentland, Probabilistic visual learning for object detection, *Proc. 5th Int. Conf. on Computer Vision*, pp. 786–793. Cambridge, Massachusetts (1995).
18. B. Moghaddam and A. Pentland, An automatic system for model-based coding of faces, *Proc. IEEE Data Compression Conf.* Snowbird, Utah (1995).
19. T. Darrel, B. Moghaddam, and A. P. Pentland, Active face tracking and pose estimation in an interactive room, *Proc. Computer Vision and Pattern Recognition*, pp. 67–72. San Francisco, California (1996).
20. G. Chow and X. Li, Towards a system for automatic facial feature detection, *Pattern Recognition* **26**, 1739–1755 (1993).
21. P. Juell and R. Marsh, A hierarchical neural network for human face detection, *Pattern Recognition* **29**, 781–787 (1996).
22. M. C. Burl and P. Perona, Recognition of planar object classes, *Proc. Computer Vision and Pattern Recognition*, pp. 223–230. San Francisco, California (1996).
23. S. H. Jeng, H. Y. Mark Liao, Y. T. Liu, and M. Y. Chern, An efficient approach for facial feature detection using geometrical face model, *Proc. 13th Int. Conf. on Pattern Recognition*, pp. 426–430. Vienna, Austria (1996).
24. K. Matsuno, C. W. Lee, S. Kimura, and S. Tsuji, Automatic recognition of human facial expressions, *Proc. Int. Conf. on Computer Vision*, pp. 352–359. Cambridge, Massachusetts (1995).
25. K. Sobottka and I. Pitas, Extraction of facial regions and features using color and shape information, *Proc. 13th Int. Conf. on Pattern Recognition*, pp. 421–425. Vienna, Austria (1996).
26. Q. Chen, H. Wu, and M. Yachida, Face detection by fuzzy pattern matching, *Proc. Computer Vision and Pattern Recognition*, pp. 591–595. Cambridge, Massachusetts (1995).
27. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Addison-Wesley, Reading, Massachusetts (1992).
28. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Addison-Wesley, Reading, Massachusetts (1993).
29. A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice-Hall, Englewood cliffs, New Jersey (1989).

About the Author—SHI-HONG JENG received a B.S. degree in electrical engineering from Chung-Yuan Christian University in 1991 and an M.S. degree in electrical engineering from National Chung-Cheng University in 1995. He is currently an Assistant Research Engineer at the Telecommunication Laboratories of Taiwan. His major research interest includes image processing, pattern recognition, computer vision, and neural networks.

About the Author—HONG-YUAN MARK LIAO received a B.S. degree in physics from National Tsing-Hua University in 1981, and an M.S. and a Ph.D. degree in electrical engineering from Northwestern University in 1985 and 1990, respectively. He is currently an Associate Research Fellow in the Institute of Information Science, Academia Sinica, Taipei, Taiwan. He is also a member of the IEEE Computer Society and the International Neural Network Society (INNS). His current research interest includes neural networks, computer vision, fuzzy logic, and image analysis.

About the author—CHIN-CHUAN HAN He received the B.S. degree in computer engineering from National Chiao-Tung University in 1989, and an M.S. and a Ph.D. degree in computer science and electronic engineering from National Central University in 1991 and 1994, respectively. He is currently a postdoctoral fellow in the Institute of Information Science, Academia Sinica, Taipei, Taiwan. His research interests are in the areas of 2D image analysis, computer vision, and pattern recognition.

About the author—MING-YANG CHERN received a Ph.D. degree in electrical engineering and computer science from the University of Illinois at Chicago in 1983. From 1983 to 1987, he was an Assistant Professor in the Department of Electrical Engineering and Computer Science, Northwestern University. From 1987 to 1991, he worked as a member of Technical Staff at the AT&T Bell Laboratories. He is currently a Professor of Electrical Engineering at National Chung-Cheng University, Chia-Yi, Taiwan. His research interest includes computer vision and autonomous vehicle.

About the Author—YAO-TSORNG LIU received a B.S. in mathematics and an M.S. in computer science from National Taiwan University in 1991 and 1995, respectively. After 2-year military service in R.O.C. Navy he was a research assistant in the Institute of Information Science, Academia Sinica, Taipei, Taiwan. He is currently a graduate student at the Department of Statistics, University of Illinois at Urbana-Champaign. His major research interest includes computer vision and artificial intelligence.