Eye Detection and Face Recognition using Line Edge Map

Mihir Jain[1], Suman Mitra[2], Naresh Jotwani[3]
[1] IIIT, Hyderabad, 500032
[2 & 3] DA-IICT, Gandhinagar, 382007

***Abstract*—The task of face recognition has been actively researched in recent years, and many techniques have been proposed. This paper reports on the design, implementation and experimental study of modular implementation of Line Edge Map technique for face recognition, including preprocessing phase of exactly localizing a roughly localized face image using eye detection. We present an innovative algorithm for eye detection on gray intensity face image, which combines a feature-based approach with a Template based approach using proposed eye LEM template. Matching is carried out using line segment Hausdorff distance. Experimental results of eye detection and face recognition using this technique are presented. The proposed modular implementation (or facial feature block-wise implementation) of LEM technique is experimentally compared with normal LEM technique.**

***Index Terms*—Line Edge Map, Face Detection, Eye Detection, Face Recognition, Line Segment Hausdorff Distance.**

I. Introduction

Face recognition – i.e. automatically identifying a person from digital image -- is an important research problem spanning numerous fields and disciplines. Recognition is carried out by comparing selected facial features against a facial database. Over many other biometric systems, it has the benefits of being a passive and non-intrusive system [1]. Face recognition has recently received extensive attention as one of the most significant applications of image understanding [2], [3], [4].

Much research has been carried out in this field and many different techniques have been proposed. Some of them are insensitive to some kinds of variation and others to some other kinds; for example, the Eigenface approach performs well with varying facial expressions, but badly with varying light [1]. Similarly, other existing techniques are also not free from some or other limitations. Further efforts are required to improve the performance of face recognition techniques, especially in the wide range of environments encountered in the real world.

The following technical challenges are to be tackled by any face recognition technique [1]:

- Varying lighting conditions – Due to strong reflection on the face skin caused by illumination, the shape information on faces are suppressed or lost, which could result in the increase of the error rate of classification.
- Varying facial expressions – Varying expressions produce physical variations from the neutral expression, and consequently most facial features get distorted.
- Varying pose – When the system is tested on images with different poses, such as like looking up, looking down, *et cetera*, the recognition rate tends to decrease.

Any automated Face Recognition is a three-step process: (i) Face Detection/Localization, (ii) Feature Extraction and (iii) Classification. Generally, the latter two combined is known as face recognition.

This paper deals with the face recognition part, along with the required preprocessing of roughly localized face, which includes exact localization of face image, and scaling and orientation correction of this localized face image. For exact localization we have adopted a bottom-up facial feature-based approach, with the facial feature being the eye. For this purpose we have used an eye detection algorithm described more fully in [5], which combines the efficiency of feature based approaches with the accuracy of template based approaches. This algorithm is discussed briefly in the next section. Then we explain a modular implementation (or facial feature block-wise implementation) of LEM technique [1], and compare it with normal LEM technique experimentally using the CalTech face database [6].

II. Line Edge Map Technique and Preprocessing

A. Face Recognition using Line Edge Map

Takács in [7] argued that edge images can be used for the recognition of faces without the involvement of high-level cognitive functions. Then Gao and Leung [1] proposed a suitable face feature representation, Line Edge Map (LEM), which extracts as features line segments from a face edge map. Line Edge Map integrates the structural information with spatial information of a face image by grouping pixels of face edge map to line segments. After thinning the edge map, a polygonal line fitting process known as 'dynamic two-strip algorithm' [8] is applied to generate the LEM of a face.

An example of a human frontal face LEM which we obtained by implementing this algorithm is shown in Figure 1. The basic unit of LEM is the line segment grouped from pixels of an edge map. They also introduced the primary line segment Hausdorff distance (H_{pLHD}) and the complete version of line segment Hausdoff distance (LHD), which they used to measure the similarity of face LEMs. LHD and H_{pLHD} are shape comparison measures based on LEMs, basically defining the 'distance' between two line sets. The classification is done by using this similarity measurement.

We also use LHD and H_{pLHD} in our work because this takes into account the orientation difference, parallel distance and perpendicular distance between two lines, and unlike correlation techniques measures proximity rather than exact superposition [1].

Fig. 1. Example of a face LEM.

B. Preprocessing for Face Recognition

I. Face Detection, Localization and Orientation Correction

Given an arbitrary image, the goal of face detection is to determine whether there are any faces in the image and, if present, to return the location and extent of each face [9]. Face localization is a simplified detection problem with the assumption that an input image contains only one face. Orientation correction is carried out on a localized face image by rotating it such that the line joining two eyes becomes horizontal. Before face recognition, it is necessary to perform (a) orientation correction and (b) scaling to bring all images to the same resolution.

II. Eye Detection

Technically it is advantageous to perform eye detection before the detection of other facial features, for the simple but important reason that the position of other facial features can be estimated using eye position [10]. In addition, the size, location and image-plane rotation of the face in the image can be normalized by using the positions of the two eyes.

Eye detection methods can be broadly classified into three categories: feature based methods, appearance based methods and template based methods [10].

Feature based methods explore certain eye characteristics -- such as edge and intensity of iris, the color distributions of the sclera and the flesh -- to identify some distinctive features around the eyes. Although these methods are usually efficient, they lack accuracy for the images that do not have high contrast.

In template based methods, a generic shape-based eye model is designed. This template is then matched to the face image pixel by pixel to find the eyes. Such methods can detect the eyes accurately, but they are usually time-consuming. In order to improve the efficiency of this method, a technique has been proposed in [10] that first roughly detects the two regions of eyes using a feature based method, and then performs template-matching on the reduced area, giving in effect a hybrid technique.

Appearance based methods detect eyes based on their photometric appearance. These methods usually need to collect a large amount of training data, representing eyes of different individuals under different conditions. This data is used to train some classifier, and detection is achieved by classification.

As mentioned earlier our eye detection approach efficiency of feature based approaches with the accuracy of template based approaches. By a feature based method, we first find pairs of small regions which can potentially be eye pairs. Then we apply template matching to get the final result. Template matching needs to be performed only a small number of times, once for each potential region pair. In the final phase, the two centers of eyes and the two centers of iris are located, the former in fact being more useful for exact face localization.

After detecting the eyes, other facial features can be located using the Golden Ratio (ϕ), since many key physiological proportions of the human face are based on this ratio [11].

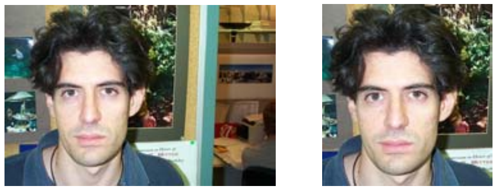
The rest of this paper is organized as follows:

Section 3 describes the proposed algorithm for eye detection, with experimental results of eye detection as well as the subsequent face localization. A 'block-wise' implementation of the LEM technique is presented in Section 4, with results, while Section 5 summarizes the present work.

The test set for all the experiments carried out consists of 112 face images of 14 individuals from the CalTech face database [6], which were randomly selected out of 450 images in the database.

III. EYE DETECTION USING LEM TEMPLATE

In the presence of many promising face detection methods [9], we assume that the face region in the input image has been roughly localized, given that the roughly localized face image may also contain some background. An original image from [6], shown in Figure 2(A), was manually cropped to get the roughly localized face image shown in Figure 2(B).



(A) (B)

Fig. 2. (A) Face image from [6], (B) Roughly localized face image.

Our proposed algorithm is tolerant to some amount of background, and therefore the preceding process of face detection need not be ‘perfect’. If necessary, it may allow an increase in the detected face extent to ensure that complete face is included.

The following algorithm outlines the proposed LEM & template-based eye detection method:

Algorithm for eye detection:

```
Input: Roughly localized face image.
Output: Two eyes, eye centers, iris centers

// Feature based part
Select potential regions for eyes.
Make pairs of selected regions.
For each pair:
Begin
     Perform orientation correction
     according to the pair.
     Test the pair 1) geometrically, 2)
     for distinctive features and 3) for
     symmetry.
      If all the tests are passed
         Store the pair as a potential
        region pair (PRP).
     If a sufficient number of PRPs are
     obtained
          Break.
End

// Template matching
For each PRP:
Begin
     Generate eye LEM template and perform
     matching
End
Select the best matched PRP as the eye
pair.
Use recursive centroid finding to get eye
centers.
Find iris centers using eye centers.
```

The template used here is the LEM of an artificial pair of eyes including eyebrows. It is constructed according to the distance between the two centroids of the PRP with which it is to be matched, based on the golden ratio ϕ [11], as shown in Figure 3. Here, matching is done by primary line segment Hausdorff distance (H_{pLHD}). Edge map is obtained by Sobel method, details of the various highlighted steps of this algorithm can be found in our previous work [5].

Fig. 3. Artificial eye LEM Template

We use the positions of eye centroids obtained by this algorithm with ϕ (golden ratio) for face localization i.e. to get the part of face from eyebrow to chin and between two ears.

Experimental Results

With the test set and the method of computation of normalized error [12], the eye detection algorithm gave 100% result for normalized error less than 0.1, and 94.6% result for normalized error less than 0.07.

For subsequent face recognition using LEM technique, out of 112 images (14 subjects) we used 42 images as test images and rest 70 were taken as model images. Model images are correctly identified images in the database, against which a test image is matched, for the purpose of recognition. LEMs of each of test and model images were obtained and each test LEM was compared to each model LEM using Hausdorff distance and the closest one was considered the output. The success rate of recognition was 97.6% in this experiment.

IV. Block-wise and I-LEM Implementation

When all other features except eyes, eyebrows, nose and lips (with some part below) are removed from the face LEM, what remains is an I-shaped face LEM which is here referred to as I-LEM of face. Experiments were conducted in which I-LEMs were used instead of face LEM, the rest of the algorithm being unchanged. In block-wise implementation, the face LEM (or even I-LEM) is divided into three blocks, containing respectively the eye region, nose region and lip region. Now the LEMs of the corresponding parts of each test face are compared with those of each model face. Extracted I-LEM of a face and its blocks are shown in Figure 4.

Fig. 4. I-face LEM and Blocks of LEM

For obtaining block LEMs and I-LEM we also need to first find the separating line between nose and lip region. Eye region is extracted using ϕ, but the nose region and lip region cannot be extracted by using ϕ alone, since the face may have a slight upward or downward tilt.

Therefore horizontal projection is performed on all the pixels in the region containing nose and lips to find the separating line between nose and eyes; here the global minimum should lie somewhere between nose and lips. For the projection, we convolved an appropriately selected matrix of all ones with the rectangular region between containing nose and lips. This is shown in Figure 5.

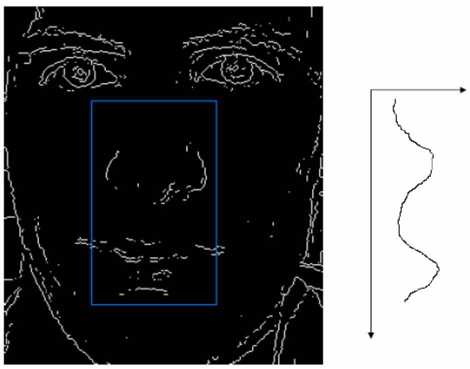


Fig. 5. Finding separating line between nose and lip region

Experimental Results

Experiments were performed with (i) LEM (ii) I-LEM, (iii) block-wise LEM and (iv) block-wise I-LEM. But with the experimental setup explained in section 3, the results obtained were essentially the same in these four cases.

So, to compare these different approaches, the experimental setup was made more stringent. Specifically, the 42 test images in first experiment were used as model images and 70 model images as test images. This makes recognition more difficult because:

(a) We have a smaller number of model images of each individual to find a match, and
(b) The 42 earlier test images (which are now model images) are having larger variations.

As an example, Figure 6 shows the model images of one subject.

For block-wise algorithms, the decision was made by polling. Weights were assigned empirically to face LEM (or I-LEM), eyes, nose and lip regions in the ratio 2:2:1:1 respectively. Weight given to the eye region is equal to that of the face LEM (or I-LEM) because eyes are less affected by pose variation (compared to nose and mouth) and also by change in expression (compared to mouth). The results are presented in Table I.

TABLE I
FACE RECOGNITION
RESULTS OF DIFFERENT IMPLEMENTATIONS OF LEM TECHNIQUE

Method	Result	
	Correctly detected faces	Accuracy
Face LEM	58	82.86%
I-face LEM	62	88.57%
Face LEM block-wise	61	87.14%
I-face LEM block-wise	62	88.57%

With this new experimental setup the result for the face LEM method decreased from 97.6% to 82.86%, whereas the results for other three methods are less sensitive to decrease in the number of model faces.

Interestingly, the I-face LEM method gives better result than face LEM, perhaps because it is not affected by the reflection on the cheeks which can be misguiding in variable illumination. Implementing face LEM block-wise also improves the performance from 82.86% to 87.14%.

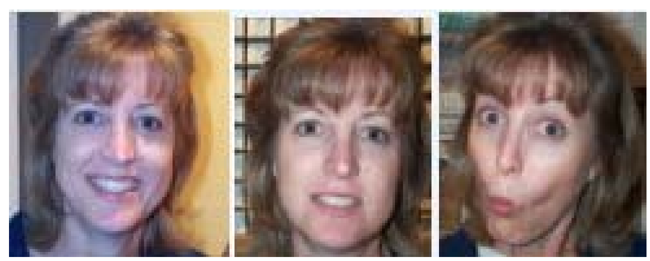


Fig. 6. Three model images of a subject - from [6]

V. CONCLUSION

Different Line Edge Map based approaches for Face recognition, namely block-wise LEM, I-face LEM, block-wise I-face LEM, are presented in this paper. We also present eye detection based solution for preprocessing of roughly localized face, which includes exact localization of face image, and scaling and orientation correction of this localized face image. Finally, the experimental results for eye detection and following face recognition are presented. The three proposed LEM based approaches are compared with the LEM technique experimentally using CalTech face database. Face recognition results of I-LEM and block-wise implementation show an improvement of 5-6 percent over normal LEM technique, and thus look promising.

REFERENCES

[1] Y. Gao and K.H. Leung, “Face recognition using line edge map,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 6, June 2002.

[2] R. Chellappa , C.L. Wilson, and S. Sirohey, “Human and Machine Recognition of Faces: A Survey,” Proc. IEEE, vol. 83, no. 5, pp. 705-740, 1995.

[3] A.P. Pentland, “Looking at People: Sensing for Ubiquitous and Wearable Computing,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp. 107-119, Jan. 2000.

[4] W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips, “Face Recognition: A Literature Survey,” Technical Report CAR-TR-948, Univ. of Maryland, CfAR, 2000.

[5] Jain, M., Mitra, S and Jotwani, N, “Eye Detection using Line Edge Map Template”, to be published, VISAPP 2008: International Conference on Computer Vision Theory and Applications, Portugal, January 2008.

[6] CalTech Face Database, http://www.vision.caltech.edu/Image _Datasets/ faces/faces.tar

[7] B. TakaÂcs, ªComparing Face Images Using the Modified Hausdorff Distance,º Pattern Recognition, vol. 31, pp. 1873-1881, 1998.

[8] M.K.H. Leung and Y.H. Yang, “Dynamic two-strip algorithm in curve fitting,” Pattern Recognition, vol. 23, pp 69-79, 1990.

[9] Ming-Hsuan Yang, Kriegman. D. J, Ahuja. N., “Detecting faces in images: a survey” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no.1, Jan 2002, pp 34-58.

[10] Kun Peng, Liming Chen, Su Ruan, Georgy Kukharev, “A robust algorithm for eye detection on gray intensity face without spectacles”, JCS&T, vol. 5, no. 3, Oct 2005, 127-132.

[11] http://goldennumber.net/face.htm

[12] Lizuo Jin, Xiaohui Yuan, Satoh, S., Jiuxian Li, Liangzheng Xia, “A hybrid classifier for precise and robust eye detection”, Pattern Recognition, 2006. ICPR. 1$8^{th}$ *International Conference*, vol. 4, no., 731-735.