Robust Facial Feature Location on Gray Intensity Face

Qiong Wang, Chunxia Zhao, and Jingyu Yang

School of Computer Science and Technology
Nanjing University of Science and Technology, Nanjing, China
nustdaisy@gmail.com

Abstract. In this paper, we propose an efficient algorithm for facial feature location on gray intensity face. Complex regions in a face image, such as the eye, exhibit unpredictable local intensity and hence high entropy. We use this characteristic to obtain eye candidates, and then these candidates are sent to a classifier to get real eyes. According to the geometry relationship of human face, mouth search region is specified by the coordinates of the left eye and the right eye. And then precise mouth detection is done. Experimental results demonstrate the effectiveness of the proposed method.

Keywords: Facial feature location, image entropy, SVM classifier, maximum-minimum filter.

1 Introduction

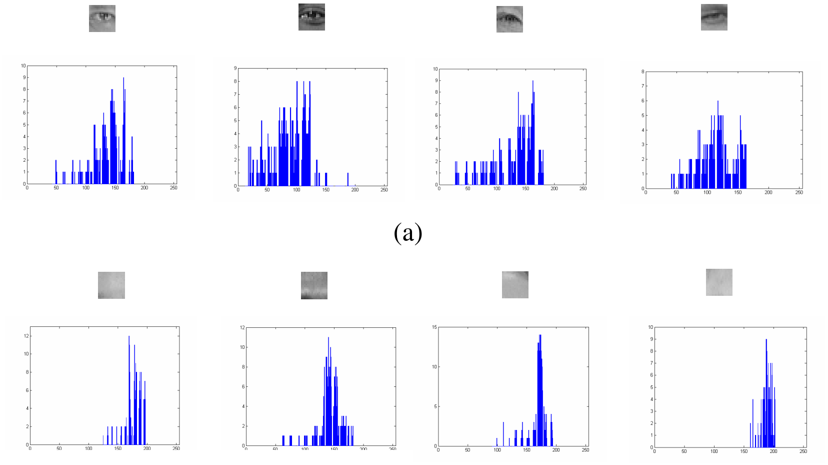
This paper addresses the problem of locating facial features (eyes, nose, mouth corners and so on) in images of frontal faces. Locating such features is an important stage in many facial image interpretation tasks (such as face verification, face tracking or face expression recognition).Generally, there are two types of information available for facial feature detection [1]: (1) local texture around a given feature, for example, the pixel values in a small region around an eye, and (2) the geometric configuration of a given set of facial features, e.g. both eyes, nose, mouth and etc. Many different methods for modeling these types of information have been proposed. In Ref. [1] a method for facial feature detection was proposed which utilizes the Viola and Jones face detection method [2], combined with the statistical shape models of Dryden and Mardia [3]. In Ref. [4] an efficient method was proposed for eye detection that used iris geometries to determine the region candidates which possibly contain the eye, and then the symmetry, for selecting the couple of eyes. In Ref. [5], Gabor feature is used to extract eyes. The EOF (entropy of likelihood) feature points are found to do feature selection and correspondence for face images in Ref. [6].

In this paper, we propose an efficient approach combining image entropy and classifier to precisely extract the eyes and locate the mouth with the coordinate information of eyes. We address the problem of facial feature detection, so our research work is based on face detection. The rest of this paper is organized as follows. In Section 2, the eye candidates extraction method will be introduced. Eyes verification will be presented in Section 3. In Section 4, mouth location algorithm will be introduced. Some experimental results will be demonstrated in Section 5 to corroborate the proposed approach. Section 6 concludes the paper.

T. Wada, F. Huang, and S. Lin (Eds.): PSIVT 2009, LNCS 5414, pp. 542–549, 2009.

2 Eye Candidates Extraction

Complex regions in a face image, such as the eyes, exhibit unpredictable local intensity and hence have higher entropy than skin region, as illustrated in Fig. 1. This fact leads us to use entropy as a measure for uncertainty and unpredictability. We use this characteristic to obtain eye candidates.



(a)

(b)

Fig. 1. Entropy comparison between eye region and skin region. (a) entropy of eye region is higher. (b) entropy of skin region is lower.

2.1 Image Entropy

The basic concept of entropy in information theory has to do with how much randomness there is in a signal or random event. An alternative way to look at this is to talk about how much information is carried by the signal.

Claude E. Shannon [7] defines entropy in terms of a discrete random event x, with possible states (or outcomes) $1...n$ as:

$$H(x) = \sum_{i=1}^{n} p(i)\log_2\left(\frac{1}{p(i)}\right) = -\sum_{i=1}^{n} p(i)\log_2 p(i) \qquad (1)$$

Conversion from probability $p(i)$ to entropy $h(i)$ is illustrated in Fig. 2, and shows that probabilities close to zero or one produce low entropy and intermediate values produce entropies near 0.5.

Shannon shows that any definition of entropy satisfying his assumptions will be of the form:

$$-K\sum_{i=1}^{n} p(i)\log p(i) \qquad (2)$$

where K is a constant (and is really just a choice of measurement units).

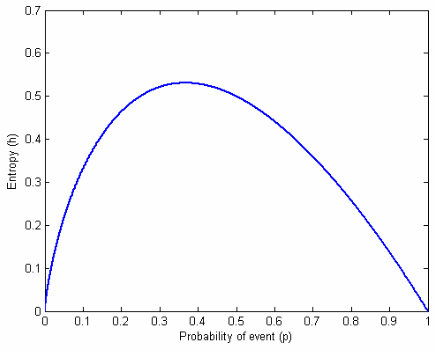


Fig. 2. Conversion from probability pi to entropy hi

The texture of the input image can be characterized by using the entropy which is a statistical measure of randomness.

For an image x, quantised to M levels, the entropy H_x is defined as:

$$H_x = \sum_{i=0}^{M-1}\left(p_i \log_2\left(\frac{1}{p_i}\right)\right) = -\left(\sum_{i=0}^{M-1}(p_i \log_2 p_i)\right) \tag{3}$$

where p_i($i = 0 \ldots M-1$) is the probability of the i^{th} quantiser level being used (often obtained from a histogram of the pixel intensities). For grey image, the value of M is 256.

2.2 Eye Candidates Extraction

Our work is focus on facial feature location, so face area is detected by using upright frontal face detector [2]. Then eye candidates are extracted on the detected face area.

A square window moves on the upper part of detected face to extract eye candidates by calculating the entropy value in each window. The size of moving window is calculated according to the face size, Eq. (4) gives the relationship.

$$\mathrm{w}in\_eye = win\_face / 4.6 \tag{4}$$

where win_eye is the slide length of moving square window; win_face is the slide length of detected face region.

The areas where their local entropy is above average are considered as eye candidates and sent to the eye verifier. Examples of eye candidates extraction are shown in Fig. 3.

In order to detect faces in different scales, the facial image is repeatedly scaled by a factor of 1.2. In each scale, all eye pair candidates are extracted and verified by the eyes verifier which will be described in the next section. Consequently, all the faces in one image can be detected.

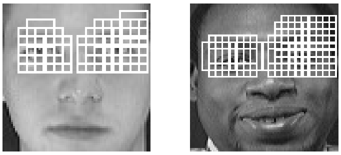


Fig. 3. Examples of eye candidates extraction

3 Eye Verification

After eye candidates are extracted by calculating entropy, an eye verifier is applied to obtain real eyes. We train a SVM classifier to do eye verification.

3.1 Support Vector Machine

In this paper, we choose the SVM as the classifying function. One distinctive advantage this type of classifier has over traditional neural networks is that SVMs achieve better generalization performance.

SVM is a patter classification algorithm developed by V. Vapnik and his team [8]. It is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization. Given a set of N examples:

$$(x_1,y_1), ...(x_i,y_i), ...(x_N,y_N) \quad x_i \in \mathrm{R}^{\mathrm{N}}, y_i \in \{-1,1\}$$

In case of linear separable data, maximum margin classification aims to separate two classes with hyperplane that maximizes distance of supports vectors. This Optimal Separating Hyperplane can be expressed as following formula:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i (x_i^T x) + b \tag{5}$$

This solution is defined in terms of subset of training samples (supports vectors) whose α_i is non- zero.

In the case of linearly non-separable patterns, SVM is to perform non-linear mapping of input vector into high dimensional dot product space F . In general, however, the dimension of the feature space is very large, so we have the technical problem of computing high dimensional spaces. Kernel method gives the solution to this problem. In Eq. (5), substituting $x_i^T x$ to $\varphi^T(x_i)\varphi(x)$ leads to the following formula:

$$f(x) = \mathrm{sgn}[\sum_{i=1}^{N} y_i \alpha_i \varphi^T(x_i)\varphi(x) + b] \tag{6}$$

This kernel method is backed up by Mercer's theorem. Thus the formula for nonlinear SVM with kernel is

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i k(x_i, x) + b \tag{7}$$

The requirement on the kernel $k(x_i, x)$ is to satisfy Mercer's theorem. Within this requirement there are some possible inner product kernels. There are Gaussian Radial Basis Functions, polynomial functions, and sigmoid polynomials whose decision surfaces are known to have good approximation properties. In this paper, we choose Gaussian radial basis function as the kernel function.

3.2 Eye Verifier

We apply SVM classifier to verify the eye candidates. The training data used for generating eye verification SVM consists of 600 images of each class (eye and non-eye). Selection of proper non-eye images is very important to train SVM because performance of SVM is influenced by what kind of non-eye images is used. In the initial stage of training SVM, we use non-eye images similar to eyes such as eyebrows, nostrils and other eye-like patches. And we generate non-eye images using bootstrapping method [9].

4 Mouth Location

Mouth location is also an important part in facial expression recognition and face states recognition. After the real eyes are obtained, mouth is located sequentially. Firstly, mouth region is calculated according to the coordinates of left eye and right eye so that the searching region for mouth location is effectively reduced. On this basis, precise mouth location is done.

4.1 Mouth Search Region Calculation

A mouth search region is specified by the positions of the detected eyes regarding the geometric information of a face. That is, the eyes and mouth are located statistically [10].

The mouth search region is represented with two coordinates (M_{left}, M_{top}) and (M_{right}, M_{bottom}) by equation (8).

$$\begin{bmatrix} M_{left} \\ M_{right} \\ M_{top} \\ M_{bottom} \end{bmatrix} = \begin{bmatrix} 0.965x_{left} + 0.035x_{right} \\ 0.035x_{left} + 0.965x_{right} \\ y_{eye} + 0.64(x_{right} - x_{left}) \\ y_{eye} + 1.44(x_{right} - x_{left}) \end{bmatrix} \tag{8}$$

where (x_{left}, y_{left}) and (x_{right}, y_{right}) are the coordinates of left eye center and right eye center respectively, and $y_{eye} = \dfrac{y_{left} + y_{right}}{2}$.

In Fig. 4, each white rectangle windows on the mouth is the mouth region calculated by two eyes coordinates.

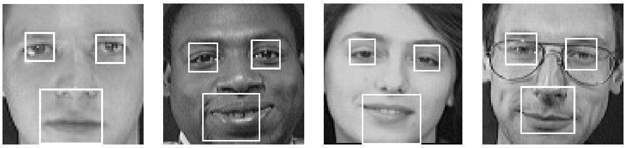


Fig. 4. Mouth region extraction

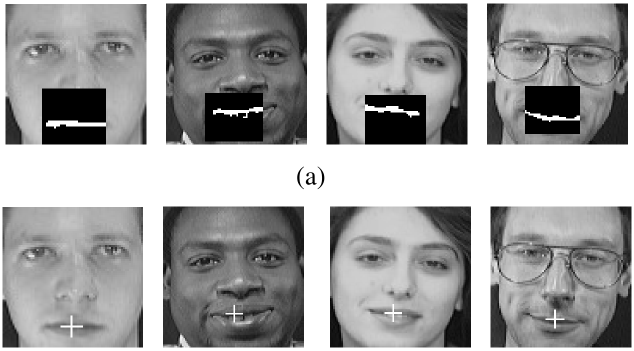
4.2 Precise Mouth Location

Once the mouth search region is extracted, precise mouth location can be done by further image processing. Because mouth has lower pixel value in mouth search region, using binary image to segment mouth is feasible. Maximum filter and Minimum filter are applied to the mouth region image as in Eq. (9).

$$f' = MinFilter(MaxFilter(f)) - f \qquad (9)$$

where f is original image and f' is differential image.

Then thresholding and close operation is applied to the differential image and mouth can be segmented, as shown in Fig. 5(a). Mouth center will be located by calculating gravity center of connected component. Fig. 5(b) shows some mouth location results.



(a)

(b)

Fig. 5. Mouth Location. (a) mouth segmentation. (b) precise mouth location.

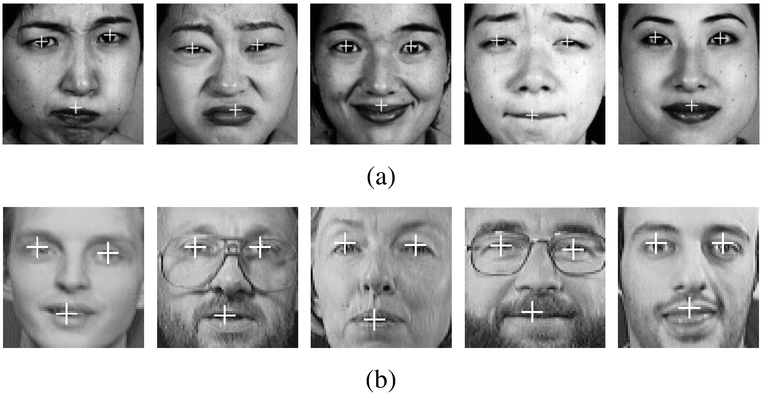
5 Experimental Results

The proposed approach was tested on the JAFFE face database and ORL face database. The JAFFE database consists of 213 frontal face images. The ORL database consists of 400 frontal face images from 40 individuals. Face is firstly detected, and then eyes and mouth are detected.

To evaluate the precision of eye localization, a scale independent localization criterion [11] is used. This relative error measure compares the automatic location result with the manually marked locations of each eye. Let C_l and C_r be the manually extracted left and right eye positions, C_l' and C_r' be the detected positions, d_l be the Euclidean distance between C_l' and C_l, d_r be the Euclidean distance between C_r' and C_r, d_{lr} be the Euclidean distance between the ground truth eye centers. Then the relative error of this detection is defined as follows:

$$err = \frac{\max(d_l, d_r)}{d_{lr}} \tag{10}$$

JAFFE contains only female faces and there is no mustache occlusion, the mouth detection rate is high. When $err < 0.1$, the eye detection rate is 99.13%, the mouth detection rate is 99.32% based on eye detection. Our algorithm outperforms the Ref. [12] and Ref. [13]. Some detection results are shown in Fig. 6 (a). However, some faces in ORL dataset contain glasses and mustache. When the glisten of glasses is too strong, the eye detection will fail, also when the occlusion on mouth is heavy, the mouth detection will fail. When $err < 0.1$, the eye detection rate is 90.67%, the mouth detection rate is 97.76% based on eye detection. Fig. 6 (b) shows some detection results.


(a)

(b)

Fig. 6. Some location results. (a) results on JAFFE database. (b) results on ORL database.

6 Conclusions and Future Research

In this paper, an efficient facial features location method for gray intensity face is presented. Experimental results show that entropy measure can extract eye candidates effectively. Based on the precise eye location, mouth search region can be calculated by the coordinates of two eyes. This makes mouth location much easier. The experimental results demonstrate its efficiency. Future work will focus on resolving the occlusion on faces and the influence of face pose to improve the algorithm performance.

Acknowledgment

This work was supported by National Natural Science Foundation of China (Grant No. 60503026, 60632050), and the Project of Science and Technology Plan of Jiangsu Province (Grant No. BG2005008).

References

1. Cristinacce, D., Cootes, T.: Facial Feature Detection Using AdaBoost with Shape Constraints. In: Proceedings of British Machine Vision Conference, pp. 231–240 (2003)
2. Viola, P., Jones, M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In: Proceedings of Computer Vision and Pattern Recognition Conference, vol. 1, pp. 511–518 (2001)
3. Dryden, I., Mardia, K.V.: The Statistical Analysis of Shape. Wiley, London (1998)
4. D'Orazio, T., Leo, M., Cicirelli, G., Distante, A.: An Algorithm for Real Time Eye Detection in Face Images. In: Proceedings of 17th International Conference on Pattern Recognition, vol. 3, pp. 278–281 (2004)
5. Du, S., Ward, R.: A Robust Approach for Eye Localization Under Variable Illuminations. In: Proceedings of International Conference on Image Processing, vol. 1, pp. 377–380 (2007)
6. Toews, M., Arbel, T.: Entropy-of-likelihood Feature Selection for Image Correspondence. In: Proceedings of 9th International Conference on Computer Vision, vol. 2, pp. 1041–1047 (2003)
7. Shannon, C.E., Waver, W.: A Mathematical Theory of Communication. Bell System Technical Journal 27, 379–423 (1948)
8. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
9. Sung, K.K., Poggio, T.: Example-based Learning for View-based Human Face Detection. IEEE Trans. Pattern Anal. Mach. Intell. 20(1), 39–51 (1998)
10. Oh, J.-S., Kim, D.-W., Kim, J.-T., Yoon, Y.-I., Choi, J.-S.: Facial component detection for efficient facial characteristic point extraction. In: Kamel, M.S., Campilho, A.C. (eds.) ICIAR 2005. LNCS, vol. 3656, pp. 1125–1132. Springer, Heidelberg (2005)
11. Jesorsky, O., Kirchberg, K.J., Frischholz, R.W.: Robust face detection using the hausdorff distance. In: Bigun, J., Smeraldi, F. (eds.) AVBPA 2001. LNCS, vol. 2091, pp. 90–95. Springer, Heidelberg (2001)
12. Zhou, Z.H., Geng, X.: Projection Functions for Eye Detection. Pattern Recognition 37, 1049–1056 (2004)
13. Ma, Y., Ding, X.Q., et al.: Robust Precise Eye Location under Probabilistic Framework. In: Proceedings of FGR, pp. 339–344 (2004)