Çoklu Gauss Karışım Modeli Tabanlı Yüz Öznitelikleri Bulma Algoritması
Multi-stream Gaussian Mixture Model based Facial Feature Localization

Kenichi Kumatani[1]*, Hazım K. Ekenel*[1]*, Hua Gao*[1]*, Rainer Stiefelhagen*[1]*, Aytül Erçil*[2]

[1]Institut für Theoretische Informatik, Universität Karlsruhe (TH), Karlsruhe, Germany
[2]Faculty of Engineering and Natural Sciences, Sabancı University, İstanbul, Turkey
{kumatani,ekenel,hua.gao,stiefel}@ira.uka.de, aytulercil@sabanciuniv.edu

Özetçe

Bu bildiride göz, burun ve ağız kenarlarının koordinatlarını kestiren yeni bir yüz özniteliklerini bulma algoritması sunulmuştur. Çalışmada, yüzün görünüm ve yapısal özelliklerini betimleme amacı ile çoklu Gauss karışım modellerinden faydalanılmıştır. Her yüz öznitelik bölgesinin görünümünü modellemek için bir Gauss karışım modeli kullanılmıştır. Yüzün yapısal bilgisini modellemek için de bir Gauss karışım modeli eğitilmiştir. Bu modeller paralel bir yapı ile tümleştirilmiştir. Geliştirilen algoritmanin başarımı BioID yüz veritabanı üzerinde sınanmıştır.

Abstract

This paper presents a new facial feature localization system which estimates positions of eyes, nose and mouth corners simultaneously. In contrast to conventional systems, we use the multi-stream Gaussian mixture model (GMM) framework in order to represent structural and appearance information of facial features. We construct a GMM for the region of each facial feature, where the principal component analysis is used to extract each facial feature. We also build a GMM which represents the structural information of a face, relative positions of facial features. Those models are combined based on the multi-stream framework. It can reduce the computation time to search region of interest (ROI). We demonstrate the effectiveness of our algorithm through experiments on the BioID Face Database.

1. Introduction

Finding facial features is an important technology for many applications such as face registration, emotion recognition and audio-visual speech recognition.

Although many systems have been developed, most of them have been based on one concept, a coarse-to-fine strategy. Such algorithms first localize the region of interest (ROI) roughly, and then refine the estimated position with the more computationally expensive but accurate method. The total computation is significantly reduced by limiting a search area in the coarse localization stage. In order to decrease the search area, many systems used structural information of human's face such as the fact that nose and mouths are located below eyes' position. Vertical and horizontal projections of an image can be viewed as one of structural information [**?**, **?**]. Since a vertical projection function tends to have local minima around eyes and nostrils, we can limit search areas for those features around those points. However, those methods are not robust for illumination noises and subject's characteristics. Accordingly, many small rules are added. Those hard-decision rules make it difficult to maintain or improve the system. One must set new parameters or new rules empirically when a head pose orientation or illumination condition changes. Burl and Perona proposed a new approach that modeled the joint distribution of the feature coordinates with single Gaussian [**?**]. They calculated a cost function which contains the likelihood of observing a positional relation estimated by facial feature localization. Based on the value of the cost function, they selected the best hypothesis. In other words, their method rejects the unlikely results by using the probabilistic model of structure information. Those conventional methods use structure information in order to limit a search area or select the most likely hypothesis. The final position is then decided with appearance information only, and it is dealt separately from structure information. However if we combine structure and appearance information stochastically in a soft-decision manner, we might improve localization accuracy further.

In this work, we propose a new algorithm which combines two kinds of information stochastically with GMM. We calculate appearance feature vectors and a shape feature vector. We then calculate the likelihood of observing those vectors. However, this direct implementation leads prohibitively expensive computation. We therefore propose a new search algorithm by assuming that the appearances of features are independently distributed. We then apply the multi-stream GMM framework to facial feature localization problem. Another property of our proposed system is, it doesn't require complicated adjustment of many parameters.

2. Shape Feature Extraction

A facial structure such as a relative position between eyes is useful for facial feature localization although it depends on a person. In the most of previous work, such information was used to limit a search area for localization in heuristic manners or reject an incorrect hypothesis [**?**]. We however represent the structure information as a *shape feature vector*, and then stochastically combine it with the *appearance feature* vector explained in Section 3.

Figure 1 shows the shape feature vector used in this work. In Figure 1, $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$ and $\mathbf{p}_4$ correspond to a right eye position, a left eye position, the middle points of nostrils, a left mouth corner and a right mouth corner, respectively. A position vector between facial feature points which corresponds to a broken arrow in Figure 1 can be written as

$$\mathbf{g}_i = \mathbf{p}_i - \mathbf{p}_0 \tag{1}$$

978-1-4244-1999-9/08/$25.00 ©2008 IEEE

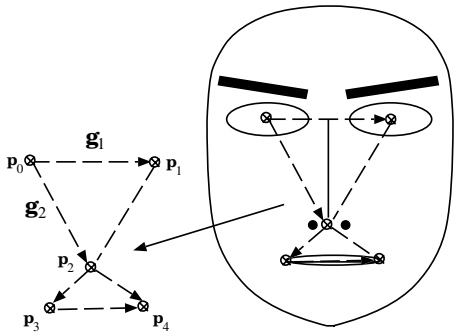

Figure 1: Shape feature vector.

where $i = 1, 2, 3, 4$ indicates each facial landmark. We normalize the scale of each vector with $\mathbf{g}_1$. Normalized vector $\mathbf{g}_i^{(n)}$ can be expressed as

$$\mathbf{g}_i^{(n)} = \left[\frac{|\mathbf{g}_i|}{|\mathbf{g}_1|} \cos(\theta_i), \frac{|\mathbf{g}_i|}{|\mathbf{g}_1|} \sin(\theta_i) \right]^T \quad i = 2, ..., 4. \tag{2}$$

where θ_i is an angle between $\mathbf{g}_1$ and $\mathbf{g}_i$. Clearly this vector is not affected by the translation, rotation and scale (TRS). By concatenating the vectors of all the facial features, we finally obtain a *shape feature vector* :

$$\mathbf{o}^{(s)} = [\mathbf{g}_2^{(n)T}, \ldots, \mathbf{g}_4^{(n)T}]^T. \tag{3}$$

3. Appearance Feature Extraction

We use the principal component analysis (PCA) in order to obtain the *appearance feature vectors* from facial features, right eye, left eye, nose and mouth corners. The illumination normalization is first performed in order to avoid the mismatch between lighting conditions of test and training. We use the histogram equalization and gradient correction as the preprocessing [**?**, **?**].

Let $\mathbf{f}$ denote the vector converted from the normalized image. We compute a PCA matrix for each facial feature. After these values are calculated, a feature vector for facial feature i can be represented as:

$$\mathbf{o}_i^{(a)} = \mathbf{\Phi}_i^T (\mathbf{f}_i - \bar{\mathbf{f}}_i) \tag{4}$$

where the matrix $\mathbf{\Phi}_i$ consists of the t eigenvectors corresponding to the largest eigenvalues. Then we can write an entire appearance feature vector which consists of all the facial feature vectors as:

$$\mathbf{o}^{(a)} = \left[\mathbf{o}_1^{(a)T}, \ldots, \mathbf{o}_5^{(a)T} \right]^T. \tag{5}$$

4. Probability Model for Facial Feature Localization

Let $\mathbf{p}$ be a set of positions of facial features which we search. We then define $\mathbf{o}_{\mathbf{p}}^{(s)}$ as a shape feature vector calculated from position set $\mathbf{p}$ and $\mathbf{o}_{\mathbf{w(p)}}^{(a)}$ as an appearance feature vector cropped with windows $\mathbf{w}(\mathbf{p})$ depending on positions $\mathbf{p}$.

By using Bayes' rule, the facial feature localization problem can be regarded as that of searching a set of positions $\mathbf{p}$:

$$\text{argmax}_{\mathbf{p}} P(\mathbf{o}_{\mathbf{p}}^{(s)},\ \mathbf{o}_{\mathbf{w(p)}}^{(a)} | M) \tag{6}$$

Figure 2: Templates for training appearance GMMs.

where M is a model for all the facial features. However it is prohibitively expensive to compute Equation 6 directly. If a size of a test image is $W \times H$, we have to calculate the probability of observing five facial features $(W \times H)^5$ times. We therefore use an approximate solution by considering that facial features are independent of each other.

4.1. Gaussian Mixture Model(GMM)

Gaussian probability density function (pdf) is widely used in many applications because it is simple and fits on many cases. We thus use its mixture model.

Single Gaussian pdf of observing a feature vector $\mathbf{o}$ can be expressed as

$$N(\mathbf{o}; \mu, \mathbf{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d |\mathbf{\Sigma}|}} \exp \left[-\frac{1}{2} (\mathbf{o} - \mu)^T \mathbf{\Sigma}^{-1} (\mathbf{o} - \mu) \right] \tag{7}$$

where d is the dimension of a feature vector, μ is a mean vector over all training vectors, and $\mathbf{\Sigma}$ is a covariance matrix. A diagonal covariance matrix is used in this work.

Then, Gaussian Mixture Model(GMM) can be written as

$$P(\mathbf{o}) = \sum_{m=1}^{M} w_m N_m(\mathbf{o}; \mu_m, \mathbf{\Sigma}_m) \tag{8}$$

Where M is the number of mixtures.

4.2. Computation Reduction with Multi-stream GMM

Assuming that appearance features are stochastically independent of each other, we can modify

$$P(\mathbf{o}_{\mathbf{p}}^{(s)},\ \mathbf{o}_{\mathbf{w(p)}}^{(a)} | M) \simeq P(\mathbf{o}_{\mathbf{p}}^{(s)} | M^{(s)}) \times \prod_{i=1}^{5} P(\mathbf{o}_{w_i(\mathbf{p})}^{(a)} | M_i^{(a)}) \tag{9}$$

where $M^{(s)}$ represents a GMM for a shape feature, $\mathbf{o}_{w_i(\mathbf{p})}^{(a)}$ is an appearance feature vector of the i-th facial landmark, and $M_i^{(a)}$ is an appearance GMM which represents the i-th facial feature.

The classification error can be further reduced by using exponent weights, that is

$$P(\mathbf{o}_{\mathbf{p}}^{(s)} | M^{(s)})^{\lambda s} \times \prod_{i=1}^{5} P(\mathbf{o}_{\mathbf{w}_i(\mathbf{p})} | M_i^{(a)})^{\lambda a, i} \tag{10}$$

Those exponent values are empirically chosen in experiments. It is well-known in audio-visual speech recognition that mismatches between training and test conditions can be circumvented by controling exponent weights for audio and visual streams [**?**].

The biggest merit of the assumption of Equation 9 is that we can search each facial feature separately. Here we describe

a new search algorithm for Equation 6. It consists of the following steps:

1. search independently right and left eyes while moving search windows and calculating the likelihoods given appearance GMMs, take the N-best candidates for each one,
2. limit search areas for a nose and mouth corners by using a shape GMM and the estimated positions of left and right eyes,
3. localize a nose and mouth corners with each appearance GMM, keep the N-best candidates,
4. calculate shape feature vectors for all the possible combinations of the candidates obtained in step 1 and step 3, compute the likelihoods for them, and
5. calculate the total score as indicated in Equation 10.

In step 2, we limit the search areas with a single Gaussian distribution of a shape feature only. Since we have vector $\mathbf{g}_i^{(n)}$ of Equation 2 obtained in step 1, we can estimate positions of facial features from a mean vector of the single Gaussian distribution. Our system doesn't search positions whose observation probability is less than 0.0001.

Although the proposed algorithm doesn't calculate Equation 10 faithfully, we can reduce the computation time efficiently.

One might think that this method depends on the estimation accuracy of eyes because it first estimates eye positions and limits the other search areas based on those results. However, since the localization accuracy is better than that of the other facial features, the degradation should be small.

5. Experiments

We used publicly available BioID database [**?**] in training and testing the systems. Shape and appearance GMMs were trained with 1001 images. The localization accuracy of the proposed algorithm was tested on 501 images. The test subjects were not included in the training data.

The criterion of localization accuracy is the normalized distance between the points obtained using automated methods and manually labelled ground truth [**?**], defined as:

$$m_{e,i} = \frac{d_i}{s} \tag{11}$$

where d_i is the point to point errors for each feature localition, and s is the inter-ocular distance of the ground truth between the left and right eye pupils. We localize five features, two eyes, nose and two mouth corners.

We first compare our system with the conventional method which localizes each facial feature individually. In the baseline system, search areas for both eyes are limited to upper-right and upper-left regions, and those for mouth corners are limited to bottom-right and bottom-left portions. Table 1 shows successfully localized rates within 20% of the inter-ocular separation of the proposed and baseline system. In tables, RE, LE, NS, RMC and LMC indicate right eye, left eye, nose, right mouth corner and left mouth corner, respectively. It is shown that using structural information improves localization accuracy of right and left mouth corners which don't have enough discriminant appearance feature. Insufficient appearance information can be compensated by a shape feature in our system. A shape feature can also limit search areas for nose and mouth corners efficiently.

system	RE	LE	NS	RMC	LMC
proposed	90.04	92.63	85.26	77.29	79.48
baseline	86.06	89.04	85.06	58.17	67.33

Table 1: The proposed system vs. baseline system.

dimension	RE	LE	NS	RMC	LMC
36	82.87	92.83	82.47	71.91	73.71
48	89.24	93.63	83.07	76.89	78.49
60	89.44	93.03	86.06	76.89	77.09

Table 2: Correctly localized rate within 20 % of the inter-ocular separation with 6 mixtures.

Our system doesn't need complicated empirical rules. However, the localization accuracy of the proposed algorithm depends on the numbers of dimensions, the number of mixtures and exponent weights of Equation 10. We examined those effects. Table 2 shows successfully localized rates for each number of dimensions of an appearance feature, where every appearance GMM has six mixtures. We can confirm from Table 2 that the higher dimension doesn't always lead to better localization performance. This is because the high dimensional part of an appearance feature vector doesn't have useful information for the localization. We can conclude from the results that 48 dimensional vectors are mostly enough in this experiment.

We also analyzed how the number of mixtures gives an effect on the localization performance. Table 3 shows the accuracy with 60-dimensional vectors for each number of mixtures. We can see from Table 3 that too many mixtures decrease localization accuracy because of data sparseness.

We conducted experiments with different stream weights in Equation 10. Table 4 presents localization rates with 4 sets of weight values. In Table 4, each component in the second column indicates a stream weight corresponding to each one of the first column, and the third columns shows localization rates within 20 % of the inter-ocular distance. For example, results in the bottom box were obtained when λ_s = 0.9, $\lambda_{a,1}$ = 0.9, $\lambda_{a,2}$ = 0.9, $\lambda_{a,3}$ = 0.85, $\lambda_{a,4}$ = 0.8 and $\lambda_{a,5}$ = 0.8. Those weights in the bottom box were determined based on localization accuracy for an individual feature. It is not clear from these results what kind of measure is good for an automatic stream estimation. However, we can see that lower stream weights of mouth corners improve total accuracy a little since mouth templates wouldn't have significant apperance feature. Each optimum stream weight may depend on localization accuracy of an individual feature.

Figure 3 shows cumulative distribution of point to point error measure. The localization accuracy for eyes were the best, whereas mouth corners were less successfully localized. This is because a mouth has a variety of looks because of make-up, speaking and moustaches while it has few discriminant features. The positional relation changes very much. Results in Figure 3 shows the difficulty to localize mouth corners.

We finally present estimated regions and points on test data images in Figure 4.

mixtures	RE	LE	NS	RMC	LMC
4	88.25	92.83	85.06	74.50	77.29
6	89.44	93.03	86.06	76.89	77.09
12	86.85	91.83	83.07	73.31	75.90
24	88.25	89.84	83.86	72.91	75.50

Table 3: Correctly localized rate within 20 % of the inter-ocular separation with 60-dimensional appearance feature vectors.

	Weight	Correct rate	Weight	Correct rate
Shape	1.0		2.0	
RE	1.0	89.24	1.0	90.04
LE	1.0	93.63	1.0	92.23
NS	1.0	83.07	1.0	85.26
RMC	1.0	76.89	1.0	76.49
LMC	1.0	78.49	1.0	78.69
Shape	1.0		0.9	
RE	2.0	90.24	0.9	90.04
LE	2.0	92.63	0.9	92.63
NS	2.0	85.26	0.85	85.26
RMC	2.0	77.49	0.8	77.29
LMC	2.0	79.08	0.8	79.48

Table 4: Correctly localized rate within 20 % of the inter-ocular separation for various exponent weights.

6. Conclusions

We proposed a new algorithm for facial feature localization. Our technique combines appearance and shape information based on a multi-stream GMM framework. We also proposed a new search algorithm which finds the set of ROIs with the maximum likelihood. The search algorithm reduces computation time considerably.

By training more data, our system can improve the localization performance further. We will use more data and do experiments on other databases. We also have a plan to evaluate other feature extraction methods such as a block DCT feature [**?**] and classifier like a support vector machine [**?**]. In addition, we are going to apply exponent estimation algorithms [**?**] to our system. Then, we will integrate our facial feature localization system to an audio-visual speech recognition system.

7. Acknowledgments

This work was supported by the European Union under the integrated project CHIL, *Computers in the Human Interaction Loop*, contract number 506909, and under the FP6-2004-ACC-SSA-2016684 SPICE project.

8. References

[1] G. Antonini, V. Popovici, and J.-P. Thiran. Independent component analysis and support vector machine for face feature extraction. *4th Intl. Conf. on AVBPA, Guildford, UK*, pages 111–118, 2003.

[2] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Trans. on PAMI*, 15, 1993.

[3] M. C. Burl and P. Perona. Recognition of planar object classes. *CVPR, San Francisco*, pages 223–230, 1996.

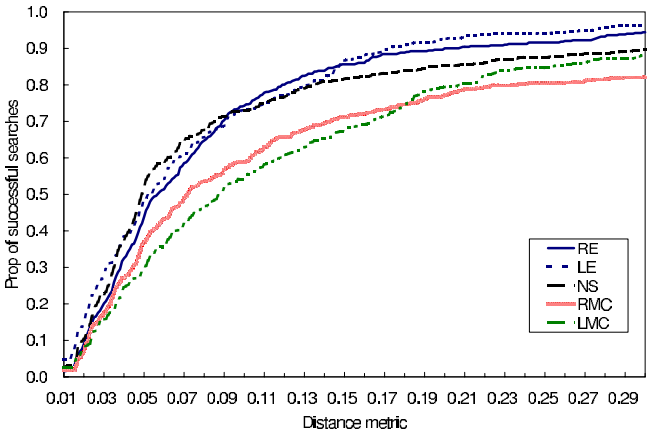

Figure 3: Cumulative distribution of point to point error measure.

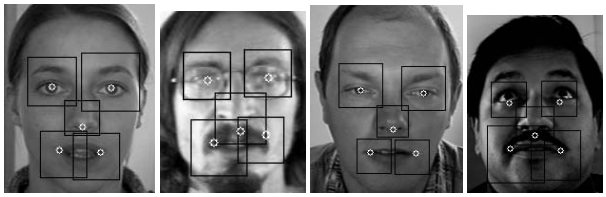

Figure 4: Estimated results.

[4] D. Cristinacce and T. F. Cootes. Facial feature detection and tracking with automatic template selection. *7th IEEE Intl. Conf. on Automatic Face and Gesture Recognition*, 2006.

[5] H. K. Ekenel and R. Stiefelhagen. Block selection in the local appearance-based face recognition scheme. *CVPR Biometrics Workshop*, 2006.

[6] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust face detection using the hausdorff distance. *Intl. Conf. on AVBPA*, 2001.

[7] J. H. Lai, P. C. Yuen, W. S. Chen, S. Lao, and M. Kawade. Robust facial feature point detection under nonlinear illuminations. *IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 168–174, 2001.

[8] G. Potamianos, C. Neti, J. Luettin, and I. Matthews. Audio-visual automatic speech recognition: An overview. in: Issues in visual and audio-vi sual speech processing. *MIT Press*, 2004.

[9] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. on PAMI*, 20:23–38, 1998.

[10] X. Zhu, J. Fan, and A. K. Elmagarmid. Towards facial feature extraction and verification for omni-face detection in video/images. *ICIP*, 2:113–116, 2002.