Universität Karlsruhe (TH)
Fakultät für Informatik
Institut für Theoretische Informatik (ITI)
Prof. Dr. A. Waibel

STUDIENARBEIT

Facial Feature Localization based on Multi-Stream Gaussian Mixture Model

Thorsten Blum

10.07.2007

Supervisors

K. Kumatani

Dr. R. Stiefelhagen

Prof. Dr. A. Waibel

Table of Contents

Abstract

In this thesis I present a technique for facial feature localization in 2D gray level images featuring a large variety of illumination, background and face size based on Gaussian Mixture Model.

The system estimates positions of eyes, nose and mouth corners simultaneously. In contrast to conventional systems, we use a 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 (PCA) is used to extract each facial feature. Furthermore we build a GMM which represents the structural information of a face like relative positions of facial features.

The combination of those two models is based on the multi-stream GMM framework and applied to search the regions of interest (ROI) in order to reduce computation time. This approach achieves the localization of facial features, being robust to scale and illumination variations. We will demonstrate the effectiveness of our algorithm through experiments on the public available BioID Face Database[1] using the FGnet Markup Scheme[2].

[1] http://www.bioid.com/downloads/facedb/

[2] http://www.bioid.com/downloads/facedb/downloads/bioid_pts.zip

1 Introduction

The localization of facial features is an important technology for many computer vision applications like facial animation, facial expression analysis, audio-visual speech recognition as well as for face and emotion recognition.

We have developed a coarse-to-fine strategy. Such algorithms first localize roughly different regions of interest (ROI) and then refine the estimated position with the more computationally intensive but more accurate method. The computation is significantly reduced by limiting search areas in the coarse localization stage. In order to decrease the search areas, many systems use structural information of a human being's face like the fact that the nose and the mouth corners are located below eyes' position. For example vertical and horizontal projections of an image can be seen as one using structural information [4] and [5]. Since a vertical projection function tends to have local minima around the eyes and nostrils, we can limit the search areas for those features around those points. However, those methods are not robust for illumination noises. Accordingly, many small rules are added. Those hard-decision rules make it difficult to maintain or improve the system.

Burl and Perona proposed a new approach that modeled the joint distribution of the feature coordinates with single Gaussian [11]. 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 structural information.

Those conventional systems use structural information in order to limit search areas or to select the most likely hypothesis. The final position is then estimated only with appearance information and structural information is dealt separately from appearance information. In contrast to that, we combined structural and appearance information stochastically in a soft-decision manner to improve localization accuracy. In fact, human beings do not use a deterministic rule. For example, we can still figure out where eyes are even if those are put on a strange position artificially. However, most of the current systems eliminate the hypothesis on the unusual position.

While researchers were dealing with structure and appearance information separately, the active appearance model (AAM) was a representative method to integrate two kinds of information [1]. However, since AAM is a fusion method at a vector level, we cannot easily introduce the weight which indicates the reliability of feature localization accuracy. Cristinacce and Cootes deployed AAM and template matching frameworks to localize facial features [2]. They also used a penalty term based on the log-likelihood of the shape model. Their underlying idea is similar to ours. The difference between their and our algorithms is that our method uses the multi-stream GMM for the integration.

We propose a new algorithm which combines two kinds of information stochastically with GMM. We calculate appearance feature vectors and a shape feature vector. Then we calculate the likelihood of observing those vectors. However, this implementation results in prohibitively expensive computational time. Therefore we propose a new search algorithm by assuming that the appearances of features are independently distributed. Then we apply the multi-stream GMM framework to the facial feature localization problem. Moreover our system does not require complicated adjustment of many parameters.

This Studienarbeit is divided into five chapters. I describe the use of shape and appearance GMMs and their combination in the multi-stream GMM framework in Chapter 2 and 3, respectively. Chapter 4 describes the dataset used in our experiments and results of facial feature localization experiments. Finally Chapter 5 concludes this Studienarbeit.

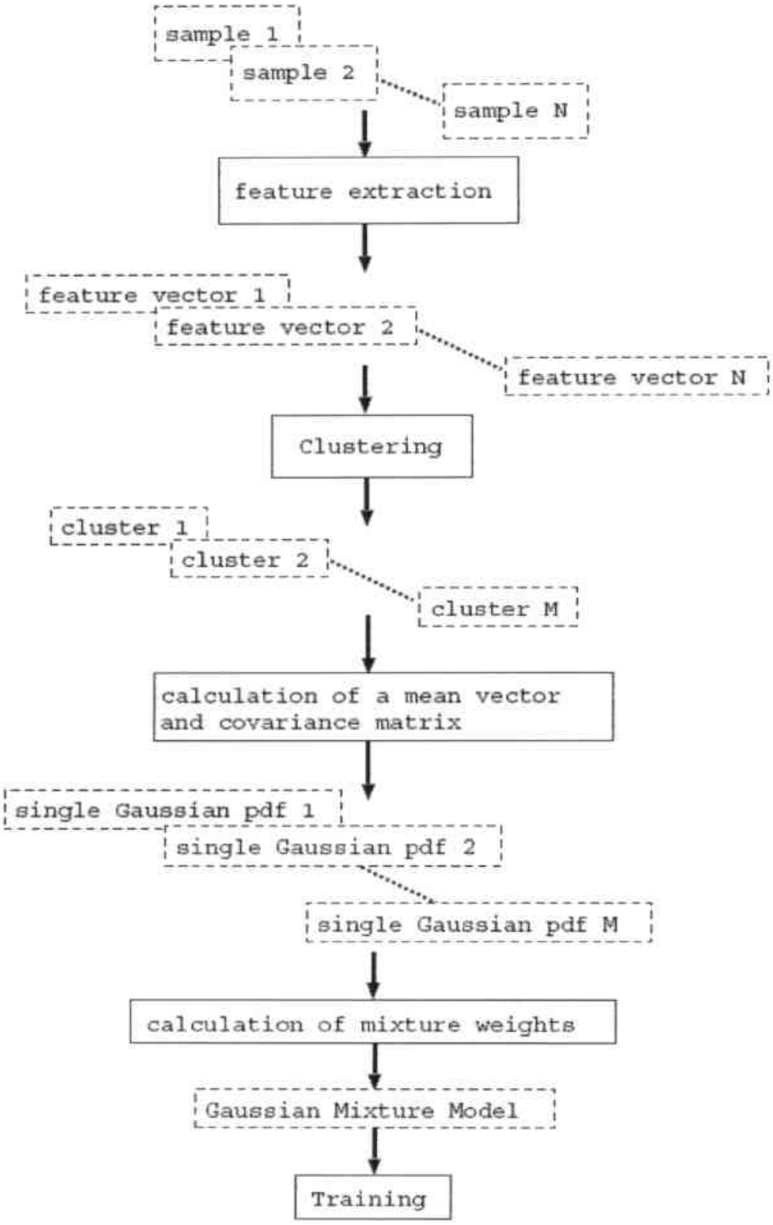

Figure 1: A basic flow chart.

2 Shape and Appearance GMMs

In this chapter I first describe how to train a GMM in general. Furthermore I describe how we have used structural and appearance information to build a shape and an appearance GMM.

2.1 Gaussian Mixture Model (GMM)

As shown in Figure 1, first N samples are collected. In our work those samples correspond to the cropped eye, nose and mouth corner images. Then a feature vector is extracted from each sample. This results in N feature vectors whose dimension is usually reduced from the original one. PCA is a typical feature extraction method. After feature extraction N feature vectors are classified into M classes by the K-mean clustering algorithm. A mean vector and a covariance matrix are calculated for each class. Then a single Gaussian probability density function (pdf) is constructed from those parameters. Single Gaussian pdf of observing a feature vector o can be expressed as

$$N(o,\mu,\Sigma) = \frac{1}{\sqrt{(2\Pi)^d|\Sigma|}}\exp[-(\frac{1}{2})(o-\mu)^T\Sigma^{(-1)}(o-\mu)] \qquad (2.1)$$

where d is the dimension of a feature vector, μ is a mean vector over all training vectors and Σ is a covariance matrix. A covariance matrix is usually diagonal to reduce computation time. The elements v_{ii} of the diagonal covariance matrix are calculated as

$$v_{ii} = E\{(y_i-\mu_i)(y_i-\mu_i)\} \qquad (2.2)$$

where E is the expectation operator. We use the ratio of the number of vectors in the *m*-th class to N as the mixture weight. Then the GMM can be written as

$$P(o) = \sum_{m=1}^{M}\omega_m N(o;\mu_m,\Sigma_m) \qquad (2.3)$$

where ω_m is the mixture weight and M is the number of mixtures that is empirically chosen. We use the ratio of the number of vectors in the m-th class to N as the mixture weight. After a GMM is initialized with the K-mean algorithm, the expectation maximization algorithm updates the parameters further [3]. We construct a shape GMM of

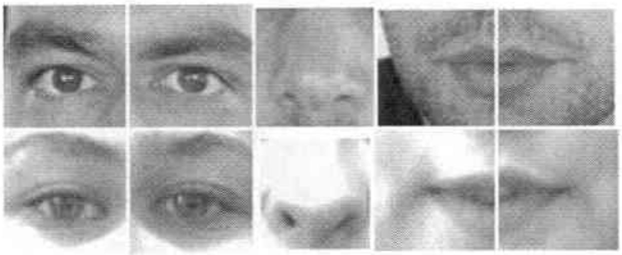

Figure 2: Templates for training appearance GMMs.

observing a shape feature vector and an appearance GMM for all the facial features from the training samples. Figure 2 shows templates of all facial features for training appearance GMMs.

2.2 Shape GMM

Structural information like the distance between two eyes is useful for facial feature localization although it might depend on an individual. In [4] and [5] this information is used to limit a search area for the localization with heuristic methods. Since those methods usually take only a few samples into account, they are not robust for various patterns. Therefore we constructed a probabilistic model of structural information.

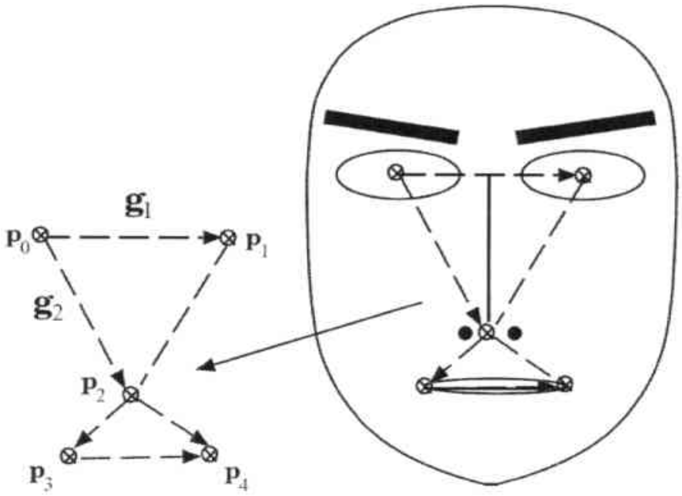

Figure 3: Structural information.

Figure 3 shows the structural information used in this work. In Figure 3, $p_{0,}$ $p_{1,}$ $p_{2,}$ p_3 and p_4 correspond to the left eye position, the right eye position, the center between the two nostrils, the left mouth corner and the right mouth corner. A vector between facial feature points represented by a broken line in Figure 3 can be written as

$$g_i = p_j - p_0$$

where

$$i = 1, 2, 3, 4$$

indicates each facial landmark. We normalize the scale of each vector with g_1 . The normalized vector $g_i^{(n)}$ can be expressed as

$$g_i^{(n)} = [\frac{|g_i|}{|g_1|}\cos(\theta_i), \frac{|g_i|}{|g_1|}\sin(\theta_i)]^T \quad i = 2,..,4 \qquad (2.4)$$

where θ_i is an angle between g_1 and g_i . Obviously this vector is not affected by translation, rotation and scale (TRS). By concatenating the vectors of all facial features we finally obtain a shape feature vector $o^{(s)}$ that can be written as

$$o^{(s)} = [g_2^{(n)T}, ..., g_4^{(n)T}]^T \ . \qquad (2.5)$$

Using equation 2.5, we can obtain a GMM which observes a feature vector $o^{(s)}$.

2.3 Appearance GMM

In order to obtain the appearance feature vectors from facial features (right and left eye, nose as well as right and left mouth corners) we use the principal component analysis (PCA). Illumination normalization is first applied in order to avoid mismatches between lighting conditions of the test and the training set. Therefore we use the histogram equalization and gradient correction [6] [7].

The PCA can reduce the dimension of a vector by retaining those characteristics of the dataset that contribute most to its variance by keeping lower-order principal components and ignoring higher-order ones. A vector f from the normalized image can be calculated as follows:

1. Calculate the mean of the vectors f .

2. Build the covariance matrix C out of the vectors f .

3. Get the eigenvectors ϕ_k and the corresponding eigenvalues λ_k of C , where $\lambda_k \geqslant \lambda_{k+1}$.

We compute a PCA matrix for each facial feature. Then a feature vector for a facial feature i can be represented as

$$o_i^{(a)} = \Phi_i^T (\mathrm{f}_i - \bar{\mathrm{f}}_i) \qquad (2.6)$$

where the matrix Φ_i consists of the eigenvectors corresponding to the largest eigenvalues. Then we can write an entire appearance feature vector which consists of all the facial feature vectors as follows

$$o^{(a)} = [o_1^{(a)T}, \dots, o_5^{(a)T}]^T \ . \qquad (2.7)$$

3 Feature Localization based on Multi-Stream GMM

In this chapter I introduce the facial feature localization problem and describe how we integrated shape and appearance features in a stochastic framework to reduce computational time.

3.1 Facial Feature Localization Problem

Let p be a set of positions of facial features which we search. We define $o_p^{(s)}$ as a shape feature vector calculated from the set of positions p and $o_{w(p)}^{(a)}$ as an appearance feature vector cropped with windows $w(p)$ depending on positions p . By using Bayes' Rule, the facial feature localization problem can be considered as of searching a set of positions p :

$$argmax_p P(o_p^{(s)}, o_{w(p)}^{(a)}|M) \quad (3.1)$$

where M is a model for the facial features. However it is prohibitively expensive to compute equation 3.1 directly. If the size of a test image is $(w \times h)$ we would have to calculate the probability of observing 5 facial features $(w \times h)^5$ times. Therefore we use an approximation solution by considering facial features independent of each other.

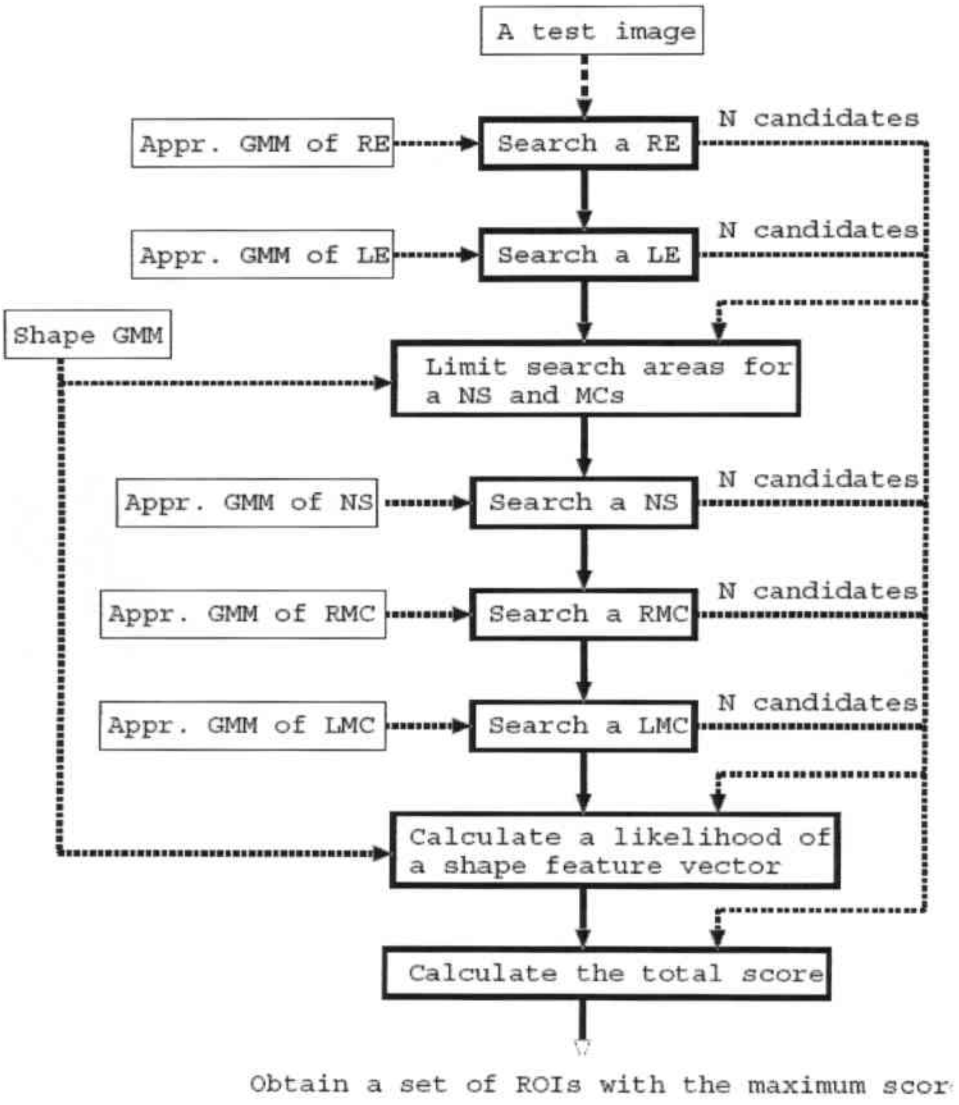

Figure 4: Block diagram of the search algorithm.

3.2 Computation Reduction with Multi-Stream GMM

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

$$P(o_p^{(s)}, o_{w(p)}^{(a)}|M) \simeq P(o_p^{(s)}|M^{(s)}) \times \prod_{i=1}^{5} P(o_{w_i(p)}^{(a)}|M_i^{(a)}) \qquad (3.2)$$

where $M^{(s)}$ represents a GMM for a geometric feature, $o^{(a)}_{w_i(p)}$ is an appearance feature vector of the i-th facial landmark, and $M^{(a)}_i$ is an appearance GMM which represents the i-th facial feature. The classification error can be reduced further by using exponent weights, that is

$$P(o_p^{(s)}|M^{(s)})^{\lambda s}\times\prod_{i=1}^{5}P(o_{w_i(p)}|M_i^{(a)})^{\lambda a,i} \quad . \qquad (3.3)$$

Those exponent values are empirically chosen in experiments. It is well-known in audio-visual speech recognition that mismatches between training and testing conditions can be prevented by controlling exponent weights for audio and visual streams [8].

The biggest merit of the assumption of equation 3.2 is that we can search each facial feature separately. Here we describe a new search algorithm for equation 3.1.

Figure 4 represents a block diagram of the proposed algorithm. It consists of the following steps:

1. Search right and left eyes independently while moving search windows and calculating the likelihoods given by the appearance GMMs and keep the *N*-best candidates for each one.

2. Limit search areas for nose and mouth corners by using a shape GMM and the estimated positions of left and right eyes.

3. Localize the nose and mouth corners with each appearance GMM and keep the *N*-best candidates.

4. Calculate shape feature vectors for all possible combinations of the candidates obtained in the first and the third step and compute the likelihoods for them.

5. Finally calculate the total score as indicated in equation 3.3.

In the second step we limit the search areas with a single Gaussian distribution of a shape feature only. Since we have obtained vector $g_i^{(n)}$ of equation 2.4 in the first step, we can estimate positions of facial features from a mean vector of the single Gaussian distribution. Our system does not search positions whose observation probability is less than 0.0001.

Although the proposed algorithm does not calculate equation 3.3 faithfully, we can reduce the computation time efficiently. It takes about 10 seconds to process a single image on an Intel Pentium 4 running at 2.4GHz with 1GB of memory. We used 4 candidates for the left and the right eye, 16 candidates for the nose and also 16 candidates for the left and the right mouth corner.

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 other facial features, the degradation should be small.

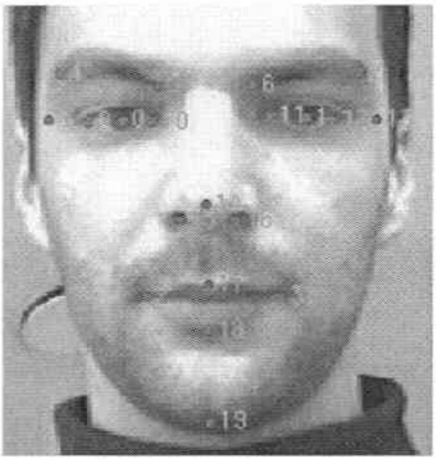

Figure 5: FGnet Markup Scheme.

4 Experimental Results

In this chapter I describe the dataset used in our experiments, the training and the testing set, the measure of evaluation and finally results of facial feature localization experiments.

4.1 Dataset

For our experiments we used the public available BioID Face Database. The database consists of 1521 gray level images with a resolution of 384x286 pixels. Each one shows the frontal view of a face out of 23 different subjects. Furthermore we used the FGnet Markup Scheme for comparison reasons. As depicted in Figure 5 the FGnet Markup Scheme provides 20 manually placed points on each of the 1521 images.

4.2 Training and Testing Set

We require to have a non-intersecting training and testing set. Unfortunately the BioID Face database is not adjusted according to the different test persons.

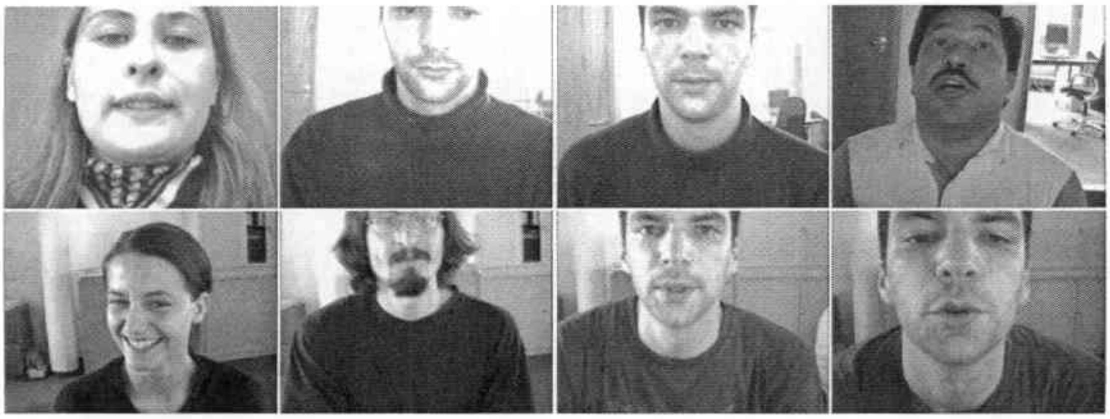

Figure 6: Manually rejected samples.

Training Set	Testing Set
994 images	507 images
17 subjects	6 subjects

Table 1: Training and testing set.

We divided the dataset manually in a training and a testing set. Due to bad conditions 20 images have been rejected from the dataset. Figure 6 shows rejected samples. They have been rejected because they do not contain an entire face and because of rotated head poses. As depicted in Table 1 shape and appearance GMMs were trained with 994 images and the localization accuracy was tested on 507 images. The test subjects have not been included in the training data. Figure 7 shows the test images which are cropped around a face based on a label.

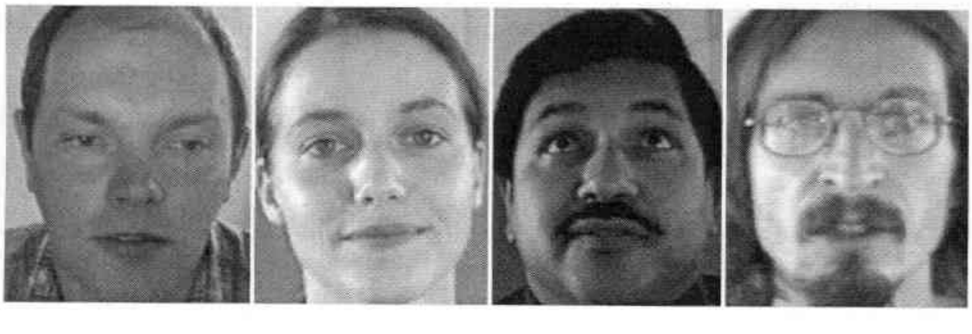

Figure 7: Test samples.

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 2: Proposed system vs. Baseline system.

RE = right eye; LE = left eye; NS = nose

RMC = right mouth corner; LMC = left mouth corner

4.3 Measure of Evaluation

The criterion of localization accuracy is the normalized distance between the automatically estimated points and the manually placed labels, defined as

$$m_{e,i} = \frac{d_i}{s} \qquad (4.1)$$

where d_i are the point to point errors for each feature localization and s is the distance between the left and the right eye pupils. We localize 5 features, left and right eye, nose as well as left and right mouth corners.

4.4 Localization Experiments

First we compare our system with the conventional method which localizes each facial feature individually. In the baseline system, search areas for both eyes are limited in upper-right and upper-left regions, and those for mouth-corners are limited within the bottom-right and the bottom-left portions. The size of a search window is two times the average size of a template used for training, that is 74x74 pixels for eyes, 66x66 pixels for a nose and 72x72 pixels for mouth corners.

Table 2 shows successfully localized rates within 20% of the interocular separation of the proposed and the baseline system. It is shown that the localization accuracy of left and right mouth corners improves by using structural information which do not 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.

Our system does not need complicated empirical rules. However, we examined that the localization accuracy of the proposed algorithm depends on the number of dimensions, the number of mixtures and the exponent weights of equation 3.3.

Table 3 shows successfully localized rates for each number of dimensions of an appearance feature, where every appearance GMM has 6 mixtures. From Table 3 we can confirm that a higher dimension does not lead to a better localization performance. This is because the high dimensional part of an appearance feature vector does not have useful information for the localization. From the results we can conclude that 48 dimensional vectors are mostly enough in this experiment.

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 3: Correct localized rate within 20% of the interocular separation with 6 mixtures.

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 4: Correctly localized rate within 20% of the interocular separation with 60-dimensional appearance feature vectors.

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

Table 5: Correctly localized rate within 20% of the interocular separation for various exponent weights.

We also analyzed how the number of mixtures effects the localization performance. Table 4 shows the accuracy with 60-dimensional vectors for each number of mixtures. We can see that too many mixtures decrease the localization accuracy because of data sparseness.

We conducted experiments with different stream weights in equation 3.3. Table 5 presents localization rates with 4 sets of weight values. Each component in the second column indicates a stream weight and the third column shows localization rates within 20% of the interocular distance. For example, results in the last box where obtained when $\lambda_s = 0.9,\ \lambda_{a,1} = 0.9,\ \lambda_{a,2} = 0.9,\ \lambda_{a,3} = 0.85,\ \lambda_{a,4} = 0.8,\ \lambda_{a,5} = 0.8$. Those weights in the last box were determined based on the localization accuracy for an individual feature. From these results it is not clear 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 appearance features. Each optimum stream weight may depend on localization accuracy of an individual feature.

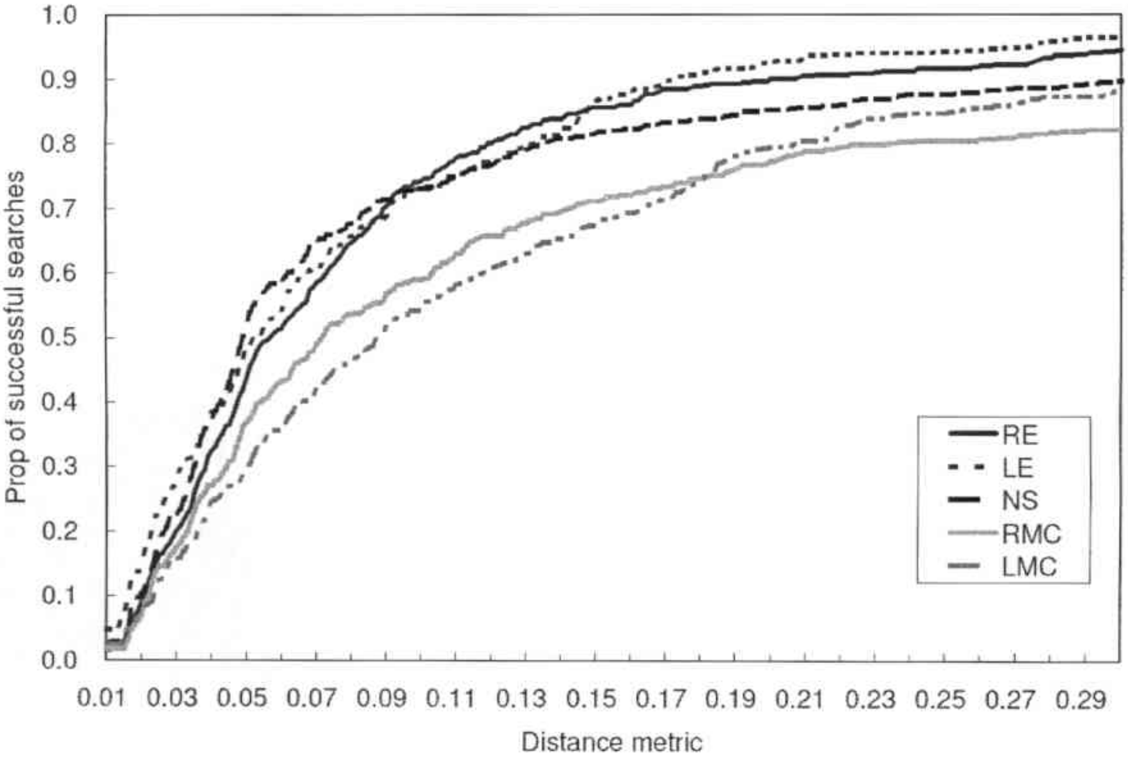

Figure 8: Cumulative distribution of point to point error measure.

I finally present estimated regions and points on test data images in Figure 9.

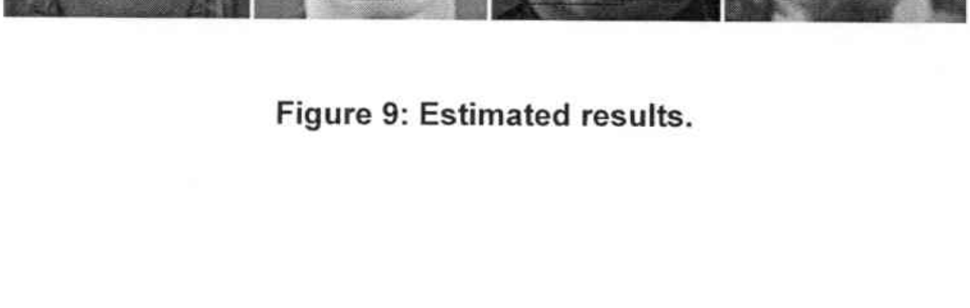

Figure 9: Estimated results.

5 Conclusion and Future Work

In this Studienarbeit we proposed a new algorithm for facial feature localization. Although many systems have been developed, most of them have only been based on a coarse-to-fine strategy. Those conventional systems use structural information in order to limit search areas or to select the most likely hypothesis. The final position is then estimated only with appearance information and structural information is dealt separately from appearance information. In contrast to that, our technique combines shape and appearance 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. We compared our proposed system with a baseline system and showed that by using structural information the localization accuracy of left and right mouth corners can be improved considerably. In addition, 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. Furthermore our system does not need complicated empirical rules. We demonstrated the effectiveness of our algorithm through experiments on the public available BioID Face Database so that it is comparable with other approaches.

Although we could decrease the computation time considerably by our new search algorithm, this approach is currently not applicable for real-time use. It still takes about 10 seconds to process a single image on an Intel Pentium 4 running at 2.4GHz with 1GB of memory.

As future work, it would be useful to train more data to improve the localization performance further. One might use more data and do experiments on other databases to test the effectiveness of our proposed algorithm. Furthermore it would be useful to combine our proposed system with other feature extraction methods such as block DCT feature [9] and other classifiers like a support vector machine [10]. In addition it would be interesting to apply exponent estimation algorithms [8] to our system.

Abbreviations

ROI	Region Of Interest
GMM	Gaussian Mixture Model
PCA	Principal Component Analysis
DCT	Discrete Cosine Transformation
AAM	Active Appearance Model
RE	Right Eye
LE	Left Eye
NO	Nose
RMC	Right Mouth Corner
LMC	Left Mouth Corner

Acknowledgments

I would like to gratefully acknowledge the support of the Computer Vision and Pattern Analysis Laboratory[3] at Sabanci University[4] in Istanbul, Turkey. Especially I would like to thank Prof. A. Erçil and her whole team. They helped me with their ingenious suggestions and their encouragement. Their interest and valuable hints were of great help.

I have furthermore to thank K. Kumatani for the discussions and help on this topic. I would also like to thank him and Dr. R. Stiefelhagen since they looked closely at the final version of this Studienarbeit and offered suggestions for improvement.

[3] http://www.vpalab.org/

[4] http://www.sabanciuniv.edu/

Bibliography

[1] T. F. Cootes, G. J. Edwards and C. J. Taylor. “Active appearance models”. H. Burkhardt and B. Neumann, editors, *Proceedings of the 5th European Conference on Computer Vision*, Vol. 2, pp. 484-498, 1998.

[2] D. Cristinacce and T. F. Cootes. “Feature Detection and Tracking with Constrained Local Models”. *Proceedings of the British Machine Vision Conference*, Vol. 3, pp. 929-938, 2006.

[3] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. O. and D. Ollason, D. Povey, V. Valtchev and P. Woodland. *The HTK Book (Version 3.4),* Cambridge 2006.

[4] J. Lai, P. C. Yuen, W. Chen, S. Lao and M. Kawade. “Robust Facial Feature Point Detection Under Nonlinear Illuminations”. *Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pp. 168-174, 2001.

[5] X. Zhu, J. Fan, A. K. Elmagarmid. “Towards facial feature extraction and verification for omni-face detection in video/images”. *Proceedings 2002 International Conference on Image Processing*, Vol. 2, pp. 113-116, 2002.

[6] R. Brunelli and T. Poggio. “Face Recognition: Features versus Templates”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, pp. 1042-1052, 1993.

[7] H. Rowley, S. Baluja and T. Kanade. "Neural Network-Based Face Detection". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, pp. 23-38, January, 1998.

[8] G. Potamianos, C. Neti, J. Luettin and I. Matthews. "Audio-Visual Automatic Speech Recognition: An Overview". *Issues in Visual and Audio-Visual Speech Processing*. MIT Press, 2004.

[9] H. K. Ekenel, R. Stiefelhagen. "Block Selection in the Local Appearance-based Face Recognition Scheme". *CVPR Biometrics Workshop*, New York, USA, June 2006.

[10] G. Antonini, V. Popovici and J. Thiran. "Independent Component Analysis and Support Vector Machine for Face Feature Extraction". *4th International Conference on Audio- and Video-Based Biometric Person Authentication, Guildford, UK*, Vol. 2688, pp. 111-118, 2003.

[11] M. C. Burl and P. Perona. "Recognition of Planar Object Classes". *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition*. pp. 223-230, San Francisco, USA, June 1996.