

## Skin color segmentation method based on mixture of Gaussians and its application in Learning System for Finger Alphabet

Peter Gejgus, Jaroslav Placek, Martin Sperka

**Abstract:** *The paper deals with method used for segmentation of human skin from the image. Applications of the method are in the fields of gesture recognition, face detection or motion capture. After the detailed method description, the experimental interactive program designed for helping deaf students learn finger alphabet is presented. This program is part of the sign language system, which uses CCD camera, directed to the user hands with goal to capture the image and compute the matching of the posture with a desired shape.*

**Key words:** *Segmentation, Mixture of Gaussians, CCD camera, Sign Language, Finger alphabet.*

### INTRODUCTION

With the progress in technology of devices dedicated for image acquisition many computer vision applications emerged. Some of them are developed for the understanding the image where humans are subject of observation [1][2], for example gesture recognition, face detection, motion capture etc. In the first phase of image understanding process we need to separate objects which we want to analyze, i.e. select pixels from the image belonging to that objects. Simple segmentation by color thresholding may be insufficient in this case. This paper describes method suitable for segmentation of human skin based on mixture of Gaussians. In the second part of the text an application using this method designed to help deaf students during the learning finger alphabet is presented.

### SEGMENTATION BY MIXTURE OF GAUSSIANS

We have used stochastic skin-color method to segment skin-color areas (corresponding to the hands) in the image sequence. This method is robust to the various skin colors of different human races. Unimodal Gaussian stochastic model is not sufficient for the proper segmentation for this task.

The Gaussian mixture model is defined as follows: The probability distribution of a D-dimensional color vector  $\mathbf{x}$  (2-dimensional in our case, because we use chromatic color space) is represented as a weighted mixture of M basis functions (components) as:

$$p(\mathbf{x}) = \sum_{j=1}^M p(\mathbf{x} | j) \cdot P(j) \quad (1)$$

The mixing parameter  $P(j)$  corresponds to the prior probability that the data  $\mathbf{x}$  was generated by the component  $j$ . Each mixture component,  $p(\mathbf{x}|j)$ , is a Gaussian of the form:

$$p(\mathbf{x} | j) = \frac{1}{(2\pi)^{D/2} |\Sigma_j|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{x} - \boldsymbol{\mu}_j) \right\} \quad (2)$$

where  $\boldsymbol{\mu}_j$  is the mean and  $\Sigma_j$  is the covariance matrix,  $|\Sigma_j|$  is the determinant of  $\Sigma_j$ . We use the basis functions to represent regions of different color property in the Gaussian mixture color modeling. An Expectation-Maximization (EM) algorithm is utilized to determine the optimal parameters  $\Sigma_j$ ,  $\boldsymbol{\mu}_j$  and  $P(j)$ .

### EM (Expectation-Maximization algorithm)

Given the number  $M$  of Gaussian mixture in the region, the EM algorithm maximizes the likelihood:

$$L = \prod_{i=1}^N p(\mathbf{x}_i) \quad (3)$$

$N$  is the number of pixels in search region. Given  $N$  face pixels  $\mathbf{x}_i$ ,  $i = 1, \dots, N$ , Expectation-Maximization provides an effective maximum-likelihood algorithm for learning a Gaussian mixture model [3][4]. An expectation (E) step consists of evaluating the posterior probabilities  $P(j | \mathbf{x}_i)$  for each mixture component  $j$ :

$$P(j | \mathbf{x}_i) = \frac{P(j)P(\mathbf{x}_i / \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_i P(i)P(\mathbf{x}_i / \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)} \quad (4)$$

Let the sum of these probabilities be  $S_j = \sum_{i=1}^N P(j | \mathbf{x}_i)$ . A maximization (M) step then updates the mixture components as follows:

$$P(j)^{new} = \frac{S_j}{N} \quad \boldsymbol{\mu}_j^{new} = \frac{1}{S_j} \sum_{i=1}^N \mathbf{x}_i P(j | \mathbf{x}_i) \quad (5)$$

$$\boldsymbol{\Sigma}_j^{new} = \frac{1}{S_j} \sum_{i=1}^N (\mathbf{x}_i - \boldsymbol{\mu}_j^{new}).(\mathbf{x}_i - \boldsymbol{\mu}_j^{new})^T P(j | \mathbf{x}_i) \quad (6)$$

As seen in equation (4), the posterior probabilities depend on the Gaussian parameter estimates, which, according to equations (5), (6) depend on the posterior probabilities. The E and M steps are iterated until convergence. If  $M = 1$ , the parameters of the Gaussian are estimated directly. There are also techniques for initial number of Gaussian mixture estimation, described in [3].

### SYSTEM FOR FINGER ALPHABET RECOGNITION DESCRIPTION

Here we will describe interactive system designed to help deaf student to learn finger alphabet [5]. This is a part of the sign language system used for expressing alphabet letters. Figure 1 shows few of them. All letters are static postures, i.e. there is no movement required in signing and this fact simplifies the recognition process. System is based on capturing image of user hand by CCD camera (figure 2), preprocessing the image, detecting contours and finally on recognition of the letter by its contour. The program computes the match of actually captured shape with the desired one – indication of the match can be seen on the green bar in the snapshot of computer screen in figure 3. The program can be used for learning. In this case the user selects the letter which he wants to train. Another application is testing, when computer selects the letters from pre-defined set composed by a teacher, and evaluates the signing of examined student.

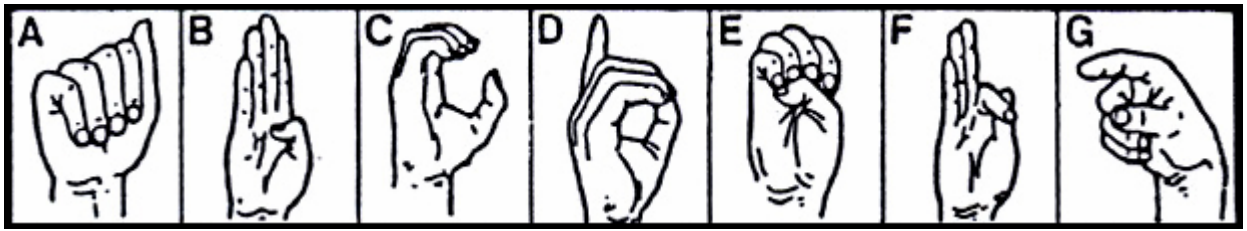


Figure 1: Several finger alphabet letters



Figure 2: Hardware setup of the system

### PROGRAM FUNCTIONALITY

The system has to be trained to be able to recognize letters. This training must be performed for every individual user because of distinctions in hand shapes of different users. After switching to the training mode, user is watching the image of his/her hands on the screen. The image is flipped horizontally giving the user the feeling that he is looking in the mirror, therefore he can control the hand shape more naturally. Hand image can be displayed with various options: original/segmented image, desired hand contour turned on/off, hand center turned on/off etc. After reaching desired shape for selected letter, the actual image contour is set as the reference contour for that letter. All other letters are trained in the same way. This phase assumes the presence of a human teacher, who helps the student to create a proper finger alphabet letter posture. However, in successive training or testing the user can exercise without him/her.



Figure 3: Using the system for finger alphabet recognition

**PROCESSING PIPELINE**

Processing queue diagram is illustrated in figure 4. After the image capture step the segmentation takes place where pixels belonging to user hands are separated from the background, composed from the user clothes, furniture etc. Before the contour detection the image preprocessing can be applied. The recognition algorithm is based on contours only, therefore it is quick enough to work in real time.

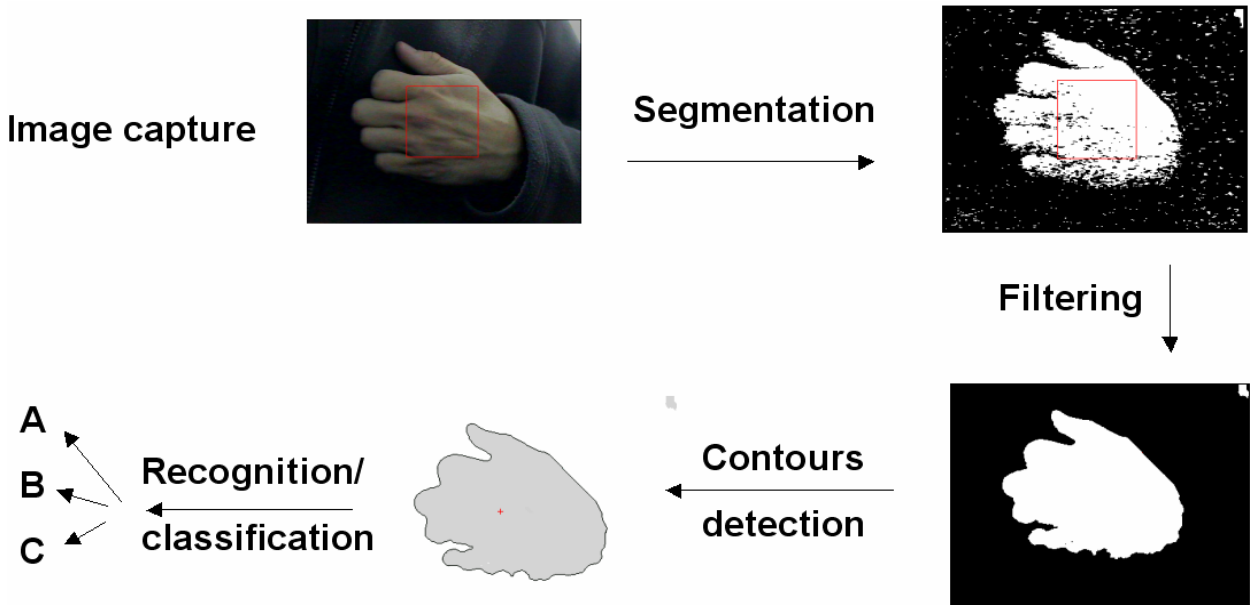


Figure 4: Processing queue diagram

The segmentation is crucial part of the process, because if we do not properly segment the image, further analysis may be impossible. Described segmentation method exploiting mixture of Gaussians gives much more acceptable results even when input image is not of high quality (lighting). Example hand image (upper left image in figure 4) is hard to segment because of heterogenous hand color. At the bottom part there are darker areas, the upper part of the hand contain much brighter pixels.

Figure 5 contains comparison of three segmentation methods. Upper three images represent hand image after segmentation by the simple threshold, multiple color sample segmentation and skin color method. To emphasize the difference between methods, bottom images contain segmentation result after consecutive application of median filter.

Segmentation by simple thresholding of color channels [6] is strongly dependent on homogeneity of hand color. If one part of hand is brighter than the other parts, it is either unselected, or the unwanted parts of the image are selected. In our case, threshold values could not be set to separate hand precisely because of different lightness of the hand. Multiple colors sample segmentation method is based on selection of few representative colors – in our case colors from the red rectangle interior are considered – see figure 4, and in segmentation process only pixels with color very similar to the given sample are selected. The results are better than by simple thresholding. Third method is segmentation by mixture of Gaussians. In [7] it was shown that mixtures of Gaussians model outperforms unimodal Gaussian model. From the figure 5 it is clear that this segmentation image method is much more usable for the next processing than previous two. Three Gaussian mixtures were used in this case.

In our program, the usage of median filter is not suitable because of computational complexity. Instead of this, contours detected from pure segmented image are filtered. Because the contours are 1-dimensional arrays of XY coordinates with length varying from 1000 to 4000 points, the filtering takes only negligible fragment of processor time in

comparison with full image median filter. The system was implemented in C++ Builder 5.0 and for median filtering and contour extraction OpenCV library was used .

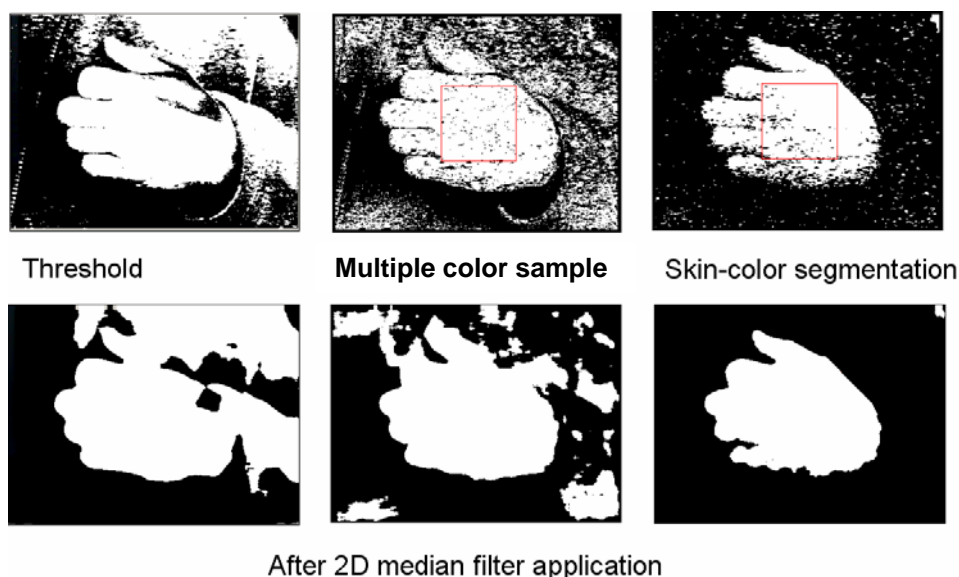


Figure 5: Comparison of segmentation methods

### CONCLUSIONS AND FUTURE WORK

As it was mentioned, segmentation is crucial part of most computer vision applications. The data which are lost due to improper segmentation can be hardly recovered in later phases of image understanding process. The selection of segmentation method strongly depends on image content. For detection of human hands, or skin generally, segmentation by mixture of Gaussians gives acceptable results. This was a reason why we have chosen this method of segmentation in the interactive application for finger alphabet recognition.

### REFERENCES

- [1] A. Niklova, J. Placek, M. Sperka. Interactive Learning System for Sign Language Configurations and Finger Alphabet. In: *International Workshop & Project Festival Computer Vision, Computer Graphics, New Media* : Graz, Austria, September 2002. pp. 233-234.
- [2] P. Gejgus, M. Sperka. Face tracking in color video sequences. In: Joy K.I. - Szirmay-Kalos, I. /EDS./: Proceedings of the Spring Conference on Computer Graphics 2003. Bratislava: UK 2003, pp. 268-273.
- [3] A. Gupta. EM Algorithm. <http://www.cse.unsw.edu.au/~akgu380/EM/EM.html>.
- [4] J. A. Bilmes. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report ICSI-TR-97-021, University of Berkeley, 1998.
- [5] A. Niklova. Using PC technology for learning of finger signs, Diploma Thesis, University of Comenius, Bratislava, 2003.
- [6] M. Sonka, V. Hlavac, R. Boyle. Image Processing, Understanding, and Machine Vision, 2nd edition, PWS Boston, 1998.
- [7] M. Sedlacek. Evaluation of RGB and HSV Models in Human Faces Detection. Central European Seminar on Computer Graphics, Budmerice, Slovakia, 2004. pp.125-131.

**ABOUT THE AUTHORS**

Peter Gejgus, PhD. student, Faculty of Mathematics, Physics and Informatics, University of Comenius, Bratislava, E-mail: [gejgus@fractal.dam.fmph.uniba.sk](mailto:gejgus@fractal.dam.fmph.uniba.sk)

Jaroslav Placek, PhD. student, International Laser Center, Bratislava, E-mail: [placek@prover.sk](mailto:placek@prover.sk)

Assoc. prof. Martin Sperka, PhD. Faculty of Informatics and Information Technology, Slovak University of Technology, Bratislava, E-mail: [martin.sperka@fiit.stuba.sk](mailto:martin.sperka@fiit.stuba.sk)