Improving of Gesture Recognition Using Multi-hypotheses Object Association

Sebastian Handrich*, Ayoub Al-Hamadi, and Omer Rashid

Institute for Electronics, Signal Processing and Communications (IESK),
Otto-von-Guericke-University Magdeburg, Germany
{sebastian.handrich,ayoub.al-hamadi,omer.ahmad}@ovgu.de

Abstract. Gesture recognition plays an important role in Human Computer Interaction (HCI) but in most HCI systems, the user is limited to use only one hand or two hands under optimal conditions. Challenges are for instance non-homogeneous backgrounds, hand-hand or hand-face overlapping and brightness modifications. In this research, we have proposed a novel approach that solves the ambiguities occurred due to the hand overlapping robustly based on multi-hypotheses object association. This multi-hypotheses object association builds the basis for the tracking in which the hand trajectories are computed and this leads us to extract the features. The gesture recognition phase takes the extracted features and classifies them through Hidden Markov Model (HMM).

Keywords: hand tracking, multi hypotheses, HCI, gesture recognition.

1 Introduction

Multimodal human behavior analysis in modern human computer interaction (HCI) systems is becoming increasingly important, and it is supposed to outperform the single modality analysis. Like other modalities, for instance facial expression [1] and prosody, gestures play an important role, since they are very intuitive and close to natural human-human interaction [2]. The analysis of gestures in HCI systems requires a robust and realtime-capable hand tracking system. In our work we provide such a system. A lot of work has been done on hand tracking and gesture recogntion. An overview can be found in [3]. However, hand tracking is due to the high number of freedoms, self occlusions and possible overlappings a difficult task. Many HCI applications are therefore limited to only one hand [4] or require other contraints, e.g. that the hand is the most foreground object [5], there are no overlappings [6] or that the user wears colored gloves [7]. In [8] a system was proposed that can handle hand-hand-overlappings if the appearance of both hands do not change during the overlapping. In [9] the authors developed a multi hypotheses based tracking approach. Such an approach provides the possibility to automaticly correct false tracking results, which is important in a HCI environment.

* This work was supported by Transregional Collaborative Research Centre SFB/TRR 62 ("Companion-Technology for Cognitive Technical Systems") funded by the German Research Foundation (DFG).

A. Elmoataz et al. (Eds.): ICISP 2012, LNCS 7340, pp. 298–306, 2012.

2 System Architecture

Figure 1 presents the architecture of the proposed approach which comprises of two main modules namely 1) hand detection and tracking, and 2) feature extraction and classification. Moreover, the hand detection and tracking problem is divided in two phases. In the first phase, skin colored objects are segmented and then clustered using 3D-data (i.e. image and depth information) of these objects by utilizing the distance from the camera. Further, the location and orientation of each object is re-estimated using Expectation Maximization (EM) algorithm at each frame. The second step contains a multi-hypotheses based tracking approach in which the hypotheses are generated based on the detected objects and fitted to the model of human body. The features are extracted in the second module which are derived from the hand trajectories and are then classified using discrete Hidden-Markov models (DHMM).

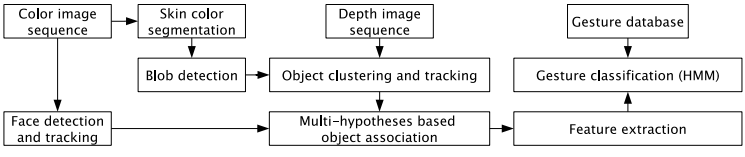

Fig. 1. System structure for hand gesture recognition

2.1 Hand Detection and Tracking

In the proposed approach, the data is acquired from Bumblebee2 camera which gives us 2D image and depth sequences. From these 2D images, the detection of skin colored objects starts with the classification of pixels as skin and non-skin pixels. For this purpose, a Gaussian mixture model (GMM) is trained using YCrCb color space. Moreover, a pixel is classified as skin color if $Y > 80$ and its probability $p(x)$ is above a threshold $p(skin) = 0.1$. $p(x)$ is determined as:

$$p(x) = \sum_{i=1}^{N} \pi_i \frac{e^{-0.5(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}}{2\pi\sqrt{|\Sigma_i|}}, \tag{1}$$

where $x = [Cr\ Cb]^T$ represents the color value of the pixel, $N = 4$ is the number of mixtures, and μ_i , Σ_i are the mean and covariance of each mixture. The trained parameters are shown in table 1. The skin color classification results in a mask image I_{skin}. Every skin-colored region (blob) in I_{skin} is detected using a blob-detection-algorithm. Since each blob B_i does not necessarily contain only one object, we create a histogram of depth values within each region. Finally, the initial number of objects and their positions are determined by peak-detection within each blob (see Fig.2). Each skin colored object O_k is described by $(\boldsymbol{\mu_k}, \Sigma_k)$, with $\boldsymbol{\mu_k} = [x_k\ y_k\ z_k]^T$ as 3D object position and Σ_k as 3x3 covariance matrix describing the spatial distribution of the 3D-points $\boldsymbol{q_i}$ that are assigned to the object (Fig. 3). Here, tracking means to re-estimate both $\boldsymbol{\mu}_k$ and Σ_k in each

Table 1. Trained parameters of the Gaussian mixture model for skin-detection. Samples were taken from a self created database with 8 different persons and 50 sample images per person (400 samples in total). The lighting conditions have remained stable due to the LED panels.

weight π_i	0.46	0.16	0.14	0.24
mean μ_i	$\begin{pmatrix} 131.1 & 141.2 \end{pmatrix}$	$\begin{pmatrix} 100.1 & 178.2 \end{pmatrix}$	$\begin{pmatrix} 110.6 & 158.8 \end{pmatrix}$	$\begin{pmatrix} 106.7 & 166.7 \end{pmatrix}$
covariance Σ_i	$\begin{pmatrix} 24.6 & 2.4 \\ 2.4 & 42.3 \end{pmatrix}$	$\begin{pmatrix} 10.3 & -13.1 \\ -13.1 & 30.6 \end{pmatrix}$	$\begin{pmatrix} 20.4 & -33.6 \\ -33.6 & 72.2 \end{pmatrix}$	$\begin{pmatrix} 17.0 & -20.6 \\ -20.6 & 40.2 \end{pmatrix}$

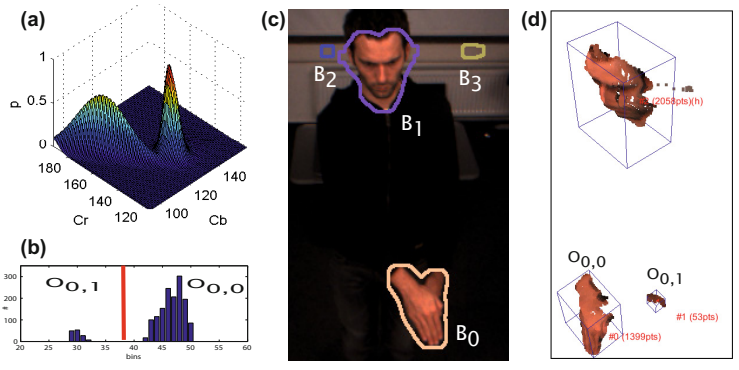

Fig. 2. Depth based object clustering: (a) Both hands share the same blob (B_0), (b) Depth-histogram of B_0, (c) Hands were detected as two separate objects

frame and is done by EM-algorithm. In the E-step, we determine the probability $p_{k,j}$ for each skin-colored 3D-point $\boldsymbol{q_j} = [x_j \ y_j \ z_j]^T$ that it belongs to object $O_k : p = N(\mu_k, \Sigma_k)$. In the M-step, we then update $\boldsymbol{\mu_k}$ and Σ_k of every object according to $p_{k,j}$. Moreover, to prevent the close objects to merge, only the M_k 3D-points with maximum probability of belonging to object O_k are used.

$$p_{k,j} = \frac{e^{-0.5(q_j-\mu_k)^T \Sigma_k^{-1}(q_j-\mu_k)}}{2\pi\sqrt{|\Sigma_k|}} \tag{2}$$

$$\mu_k = \frac{1}{|M_k|}\frac{\sum_j^{M_k} p_{k,j} \cdot \boldsymbol{q_j}}{\sum_j^{M_k} p_{k,j}}, \ \Sigma_k = \frac{1}{|M_k|}\frac{\sum_j^{M_k} p_{k,j} \cdot (\boldsymbol{q_j} - \boldsymbol{\mu_k})^2}{\sum_j^{M_k} p_{k,j}} \tag{3}$$

The E- and M-step are repeated until convergence. To avoid numerical instabilities, the diagonal elements of Σ_k are set to $\sigma_k^2 = max(\sigma_k^2, 10^{-4})$.

The second step in the proposed approach is the hand tracking. In this association problem, user's hands are identified from the observed objects O_k detected at frame t. It is a difficult task for several reasons:

- Usually three objects are to be expected (head and hands). However, there can be more objects, e.g. skin-colored clothes.
- There can be less than the three expected objects (e.g. hand is hidden).
- Hands are hard to distinguish after an overlap.

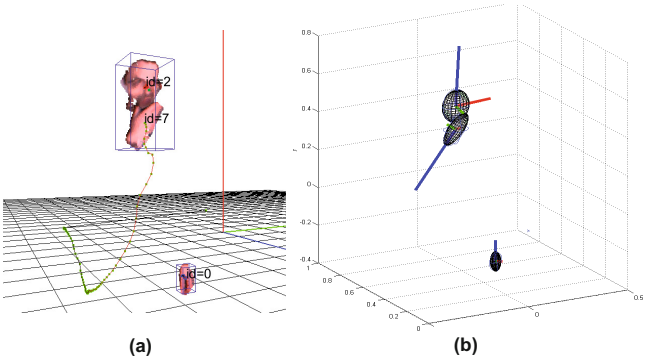

Fig. 3. Object tracking. (a) The user touches his head, but both are still separated objects. (b) the error ellipsoids of Σ_k and the corresponding eigen-vectors.

To tackle these issues, our approach contains the following steps:

- Head detection, based on face recognition. It is used for hypotheses creation.
- Based on the list of observed objects $\{O_k\}$, create a list $\{H_i\}$ of all possible hypotheses about the position of the hands at each time step t.
- Each hypothesis H_i is measured with a scoring function $S_p(H_i)$ which gives the probability of estimated pose according to model of the human body.
- Determine how each hypothesis matches the predictions of the N best hypotheses $\widehat{H_j}$ of the previous time step ($S_M(H_i, \widehat{H_j})$).
- Calculate the total score of all combinations and select the N best hypotheses. Discard all other hypotheses. In our system N was limited to 50 for computational reasons.

In HCI environments, where the user mostly faces the camera, it is unlikely that both elbows cross. So, each hypothesis contains the following assumptions: $H_i = \{x_h, x_{lh}, x_{rh}, x_{le}, x_{re}\}$, where x_h is the position of the head, determined by the face detector, x_{lh}, x_{rh} are the assumed positions of both hands and x_{le}, x_{re} are the estimated positions of the elbows. To estimate x_{le}, x_{re}, we assume that the elbow is in the direction of the largest hand extension. This is not always correct, however, sufficient for the validation of the hypotheses. So, for each skin-colored object $O_k = (\mu_k, \Sigma_k)$, we calculate the normalized largest eigenvector v_k of Σ_k and assume that the elbow is at $x_{e1,k1} = \mu_k \pm 0.3 \cdot v_k$, with 0.3 the estimated length of an underarm in meters. Figure 4 shows the generation of hypotheses in which three skin colored objects are observed with one object (H) recognized as head. This leads to a total of eight hypotheses. The next step is to calculate the score for each hypothesis and to discard the impossible hypotheses. So, first, we remove all the hypotheses where either the euclidian distance between the head and a hand is above 1.2meters or the hands are far behind the user($x_{hand}.z - x_h.z > 0.5m$). The score S_p of the pose is determined by assuming a ground position of both elbows relative to the head($z_{le/re} = x_h + [\pm 0.2 \ -0.4 \ 0.1]^T$) and calculating:

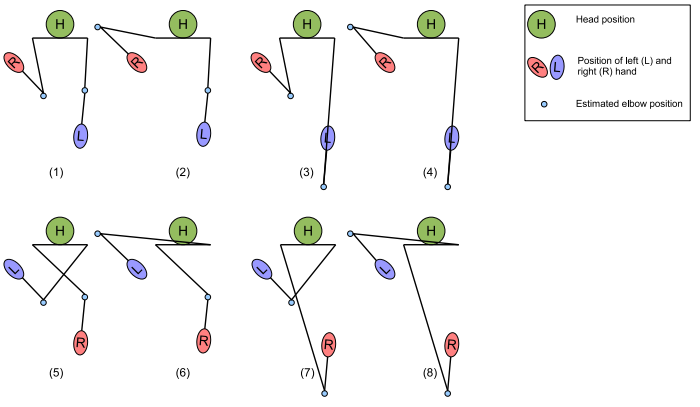

Fig. 4. Hand tracking: Based on the observed skin-colored objects, a list of all possible hypotheses about the body pose is created

$$S_p = e^{-|x_{le}-z_{le}|/\sigma_e} \cdot e^{-|x_{re}-z_{re}|/\sigma_e}. \tag{4}$$

The parameter $\sigma_e = 1$ was chosen empirically. So, the pose is less likely to be higher than the distances between the estimated and assumed elbow positions. In the final step, for each hypotheses H_i of the current timestep, the score $S_M(H_i, \widehat{H_j})$ is calculated which determines how well it matches N best hypotheses $\widehat{H_j}$ of the previous timestep. This score is based on the euclidian distances between the predicted positions of the last hypothesis and the current one. This prediction is done by assuming that the user is moving his head and hands with the same velocities as in the previous timestep. So, the velocities are the euclidian distances between each hypotheses $\widehat{H_j}$ and its parent hypothesis H_j. $S(H_i, \widehat{H_j})$ is given by:

$$S_M(H_i, \widehat{H_j}) = 1 - \frac{1}{d_{max}} \sum_{k=1}^{5} w_k \cdot d_k \tag{5}$$

with weights $w_k = [0.3\ 0.25\ 0.25\ 0.1\ 0.1]^T$ and d_k the distances between the prediction of H_j and H_i for all five hypothesis-elements, limited to d_{max}. So, the total score $S(H_i, \widehat{H_j})$ is calculated as: $S(H_i, \widehat{H_j}) = S_M(H_i, \widehat{H_j}) \cdot S_p(H_i)\ \forall i, j = 1 \ldots M, N$ with M the number of valid hypotheses in the current timestep and N the number of the best selected hypotheses of time step $t-1$. Finally the hypotheses of $\{H_i\}$ with the N highest scores are selected: $\widehat{H} \leftarrow H(S = max_N(S))$.

2.2 Feature Extraction and Classification

The features are extracted from the detected hand trajectories at each frame. There are three features used in the proposed approach as:

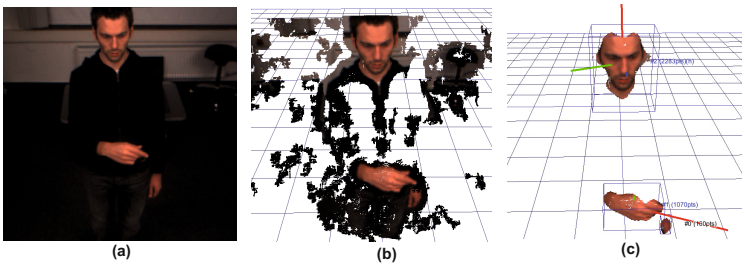

Fig. 5. Preprocessing: (a) A typical scene in the HCI environment. (b) Partially incomplete 3D data of the scene. (c) Extracted skin-colored objects (head and hands).

- Cylindrical coordinates of both hands relative to the head: $F_c = [h, r, \phi]$
- Velocities of the hands in m/s:$F_v = [v_x, v_y, v_z]$.
- Spatial orientation $F_o = \{diag(\Sigma_k)\}$ of the hands. Herefore, we used the diagonal elements of the covariance matrix Σ_k of objects, representing hands.

All the detected features are normalized to $[-1\ 1]$ and are combined to one single feature vector at every frame F_t. A complete gesture path is then described as a temporal sequence of feature vectors: $G = \{F_{t-0} \dots F_{t-T}\}$. Since, we use discrete HMM for gesture classification, the feature vectors F_t are quantized to obtain discrete symbols z_t (vector quantization). It is done by using k-means clustering algorithm. The cluster index is then used as input to the DHMM. We have used a DHMM with LRB-architecture (left-right-banded) where Baum-Welch algorithm is used for training and Viterbi algorithm for evaluation.

3 Experimental Results

We tested our tracking and gesture recognition system on some videos taken from a database, which has been created at our university in the context of a research project that focuses on the development of companion-technology. In Fig. 5(a), a typical scene in an HCI environment is shown, in which the user is performing a gesture in front of the system. Fig. 5(b) shows the corresponding 3D-data captured with the Bumblebee2 stereo camera. Within the regions of user's cloth, 3D-data is partially incomplete. In Fig.5(c), the results of skin-segmentation and object-clustering are shown. Our proposed hand tracking system is tested on app. 120,000 frames (ca. 90min) and provided very good results. One of the complicated cases is to track hands when the user crosses arms after a previous hand-hand-overlapping as shown in Fig. 6. The red (purple) circle represents the currently best assumed position of the right (left) hand. Starting from an initial position (t=0), there is a hand-hand contact (t=23). After that, the user crosses his arms (t=26) and during these time steps, the assignment of the hands was correct. However, a false assignment can occur during the hand-hand contact when the positions of the elbows are incorrectly estimated (Fig.7, t=5). However, due to the multi-hypotheses approach, the system is able to correct this false

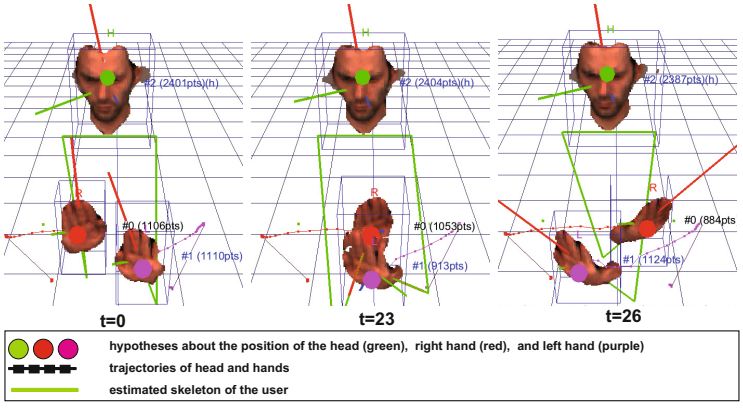

Fig. 6. Correct hand tracking in the case of hand-hand overlappings followed by a crossing of both arms

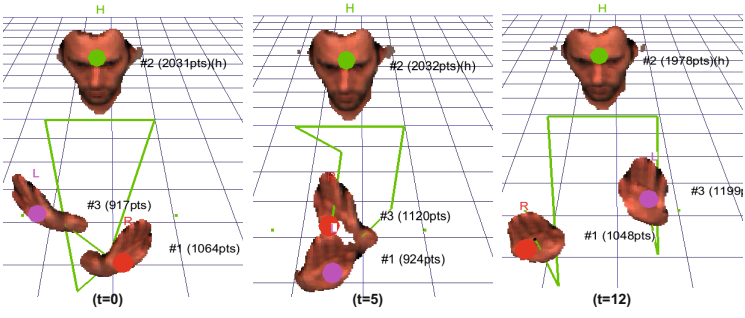

Fig. 7. Correction of false hypotheses during hand-tracking. In time step t=5 the assumed hand positions are incorrect (red/purple circles refer to left/right hand rather than vice versa. This is corrected in time step t=12.

assumption, since they become very unlikely (t=12). Since the hand detection is only based on skin-color, the system is only able to handle persons wearing long sleeves. Here, additional work to seperate arms and hands has to be done. We combined our tracking module with a basic gesture recognition system. In Fig.8, row 1-3, time course of features (Section 2.2) for two consecutive gestures and the corresponding cluster results (row 4) are shown. Best results are achieved with K=5 clusters. Although, in Fig.8, the user performed an identical gesture twice, the features and so the cluster results (t=0) (t=5) (t=12) differ. However, this problem is handled by HMM. In the gesture recognition, HMM was trained on a dataset (40 sequences) and tested on a different set (40 sequences, different users). Best results are achieved with 5-state HMM, where 95% of all performed gestures were correctly recognized. An exemplary sequence is shown in Fig. 9.

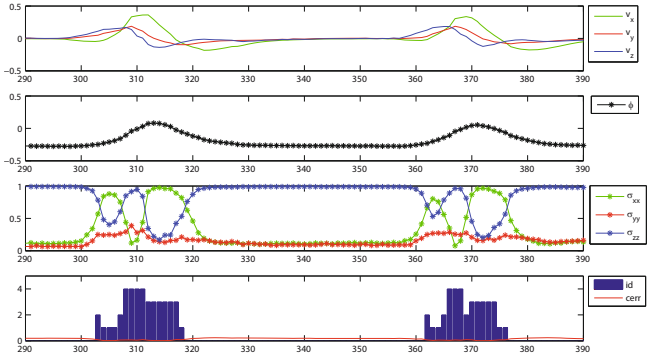

Fig. 8. Features for gesture recognition. Row 1-3 show the time course of features for two performed gestures. Bottom row: Result of k-Means algorithm used to obtain discrete observation symbols as input for the HMM.

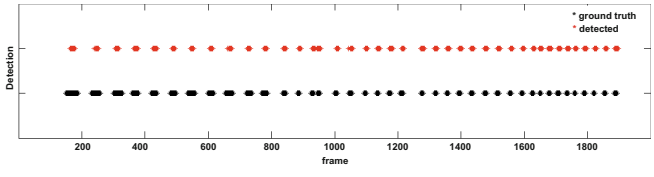

Fig. 9. Gesture recognition. The lower line (black stars) shows points in time at which the user performs a gesture (manually labeled). The upper line (red stars) shows the result of the automatic gesture recognition. All performed gestures were recognized.

4 Conclusions

Multimodal human behavior analysis in HCI environments is becoming increasingly important. Different modalities are involved in such analysis for instance facial expression, gestures and prosody. A gesture recognition system requires a robust hand tracking system. In our work, which is part of a HCI companion system, we provided such a system, which is able of tracking hands in non-trivial situations, for instance during hand-hand overlappings or hand-face-contacts. As an application we combined our tracking module with a basic gesture recognition system. The system was able to work in realtime (20 fps, 400 by 300px).

References

1. Niese, R., Al-Hamadi, A., Panning, A., Michaelis, B.: Emotion recognition based on 2d-3d facial feature extraction from color image sequences. JMM 5 (2010)
2. Hassanpour, R., Wong, S., Shahbahrami, A.: Vision based hand gesture recognition for human computer interaction: A review. In: Int. Conference Interfaces and Human Computer Interaction, pp. 125–134 (2008)

3. Shan, C., Tan, T., Wei, Y.: Real-time hand tracking using a mean shift embedded particle filter. Pattern Recognition 40, 1958–1970 (2007)
4. Suk, H.I., Sin, B.K., Lee, S.W.: Hand gesture recognition based on dynamic bayesian network framework. Pattern Recognition 43, 3059–3072 (2010)
5. Van den Bergh, M., Van Gool, L.: Combining rgb and tof cameras for real-time 3d hand gesture interaction. In: Applications of Computer Vision, WACV (2011)
6. El-Sawah, A., Joslin, C., Georganas, N., Petriu, E.: A framework for 3d hand tracking and gesture recognition using elements of genetic programming. In: Canadian Conference on CRV, pp. 495–502 (2007)
7. Keskin, C., Erkan, A., Akarun, L.: Real time hand tracking and 3d gesture recognition for interactive interfaces using hmm. In: ICANN, pp. 3–6 (2003)
8. Saeed, A., Niese, R., Al-Hamadi, A., Michaelis, B.: Solving the Hand-Hand Overlapping for Gesture Application. In: Chora ́s, R.S. (ed.) Image Processing and Communications Challenges 3. AISC, vol. 102, pp. 343–350. Springer, Heidelberg (2011)
9. Nickel, K., Stiefelhagen, R.: Visual recognition of pointing gestures for human-robot interaction. Image and Vision Computing 25, 1875–1884 (2007)