Real-time Recognition of 3D-Pointing Gestures for Human-Machine-Interaction

Kai Nickel and Rainer Stiefelhagen

Interactive Systems Laboratories
Universität Karlsruhe (TH), Germany
nickel@ira.uka.de, stiefel@ira.uka.de

Abstract. We present a system capable of visually detecting pointing gestures and estimating the 3D pointing direction in real-time. We use Hidden Markov Models (HMMs) trained on different phases of sample pointing gestures to detect the occurrence of a gesture. For estimating the pointing direction, we compare two approaches: 1) The line of sight between head and hand and 2) the forearm orientation. Input features for the HMMs are the 3D trajectories of the person's head and hands. They are extracted from image sequences provided by a stereo camera. In a person-independent test scenario, our system achieved a gesture detection rate of 88%. For 90% of the detected gestures, the correct pointing target (one out of eight objects) was identified.

1 Introduction

In the concept of multi modal user interfaces, users are able to communicate with computers using the very modality that best suits their current request. Apart from mouse or keyboard input, these modalities include speech, handwriting or gesture. Among the set of gestures intuitively performed by humans when communicating with each other, pointing gestures are especially interesting for applications like smart rooms, virtual reality or household robots. The detection of pointing gestures is particularly useful in combination with speech recognition, as they can help to resolve ambiguities and specify parameters of location in verbal statements ("Switch *that* light on!").

In this paper, a pointing gesture is defined as a movement of the arm towards a pointing target. This is why we chose the trajectory of the hand as input feature for the gesture models. Our system was designed to function in natural environments, to operate in real-time, and to be person- and target-independent. The system performs three tasks:

- color- and range-based tracking of head and hands to gain input features for the gesture models;
- classification of the trajectories by means of a combination of Hidden Markov Models (HMMs) in order to detect pointing gestures in natural movements;
- determination of the pointing direction.

1.1 Related Work

There are numerous approaches for the extraction of body features by means of one or more cameras. In [1], Wren et al. demonstrate the system *Pfinder*, that uses a statistical model of color and shape to obtain a 2D representation of head and hands. Azarbayejani and Pentland [2] describe a 3D head and hands tracking system that calibrates automatically from watching a moving person. An integrated person tracking approach based on color, dense stereo processing and face pattern detection is proposed by Darrell et al. in [3].

Hidden Markov Models have been used for years in continuous speech recognition [10], and have also been applied successfully to the field of gesture recognition. In [4], Starner and Pentland were able to recognize hand gestures out of the vocabulary of the *American Sign Language* with high accuracy. Becker [5] presents a system for the recognition of *T'ai Chi* gestures based on head and hand tracking. In [6], Wilson and Bobick propose an extension to the HMM framework, that addresses characteristics of parameterized gestures, such as pointing gestures. Jojic et al. [7] describe a method for estimating the pointing direction in dense disparity maps.

2 Tracking of Head and Hands

In our approach we combine stereoscopic range information and skin-color classification in order to achieve a robust tracking performance. The setup consists of a fixed-baseline stereo camera connected to a standard PC. A commercially available library (see [8]) calculates a dense disparity map made up of pixel-wise disparity values, and provides 3D coordinates for each pixel (Fig. 1b). A histogram-based model represents the distribution of human skin color in the chromatic color space. In order to initialize and maintain the model automatically, we search for a person's head in the disparity map of each frame. Following an approach proposed in [3], we first look for a human-sized connected region, and then check its topmost part for head-like dimensions. Pixels inside the head region contribute to the skin-color model.

In order to find potential *candidates* for the coordinates of head and hands, we search for connected regions in the morphologically filtered skin-color map.

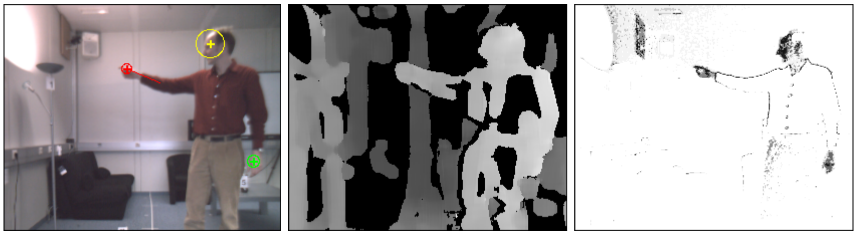
a. Left camera image
b. Disparity map
c. Skin color map

Fig. 1. In the disparity map, the brightness of a pixel is associated with its distance to the camera. In the skin color map, dark pixels represent hight skin color probability.

For each region, we calculate the centroid of the associated 3D pixels. If the pixels belonging to one region vary strongly with respect to their distance to the camera, the region is split by applying a k-means clustering method. We thereby separate objects that are situated on different range levels but accidentally merged into one object in the 2D image.

The task of tracking consists in finding a good hypothesis s_t for the positions of head and hands at time t. The decision is based on the current observation O_t (the 3D skin-pixel clusters) and the hypothesis for the preceding frame s_{t-1}. With each new frame, all combinations of the clusters' centroids are evaluated to find the hypothesis s_t that maximizes the product of the following 3 scores:

- The *observation score* $P(O_t|s_t)$ is a measure for the extent to which s_t matches the observation O_t. $P(O_t|s_t)$ increases with each pixel that complies with the hypothesis.
- The *posture score* $P(s_t)$ is the prior probability of the posture. It is high if the posture represented by s_t is a frequently occurring posture of a human body. To be able to calculate $P(s_t)$, a model of the human body was built from training data.
- The *transition score* $P(s_t|s_{t-1})$ is a measure for the probability of s_t being the successor of s_{t-1}. It is higher, the closer the positions of head and hands in s_t are to their positions in s_{t-1}.

Our experiments indicate that by using the method described, it is possible to track a person robustly, even when the camera is moving and when the background is cluttered. The tracking of the hands is affected by occasional dropouts and misclassifications. Reasons for this can be temporary occlusions of a hand, a high variance in the visual appearance of hands and the high speed with which people move their hands. Due to the automatic updates of the skin-color model, the system does not require manual initialization.

3 Detection of Pointing Gestures

When looking at a person performing pointing gestures, one can identify three different phases in the movement of the pointing hand:

- Begin (B): The hand moves from an arbitrary starting position towards the pointing target.
- Hold (H): The hand remains motionless at the pointing position.
- End (E): The hand moves away from the pointing position.

We examined pointing gestures performed by different persons, and measured the length of the separate phases. The average length of a pointing gesture was 1.8 sec. Among the three phases, the hold phase shows the highest duration variance (from 0.1sec up to 2.5sec).

For estimating the pointing direction, it is crucial to detect the hold phase precisely. Therefore, we model the three phases separately: Three dedicated HMMs (M_B, M_H, M_E) were trained exclusively on data belonging to their phase. We choose the same HMM topology (3 states, left-right) for each of the three models. For each state, a mixture of 2 Gaussian densities represents the

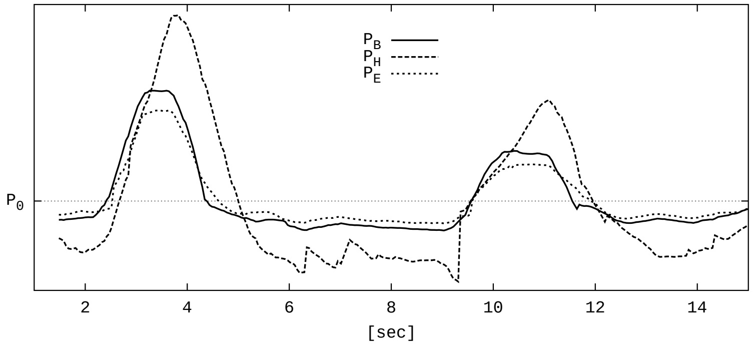

Fig. 2. Output probabilities of the phase-models during a sequence of two pointing gestures

output probability. To get a reference value for the output of the phase models, we train a *null model* M_0 on short feature sequences ($0.5sec$) which do *not* belong to a pointing gesture. For M_0, we choose an ergodic HMM with 3 states and 2 gaussians per state. The models were trained with hand-labeled BHE-phases using the Baum-Welch reestimation equations (see [10]).

3.1 Classification

As we want to detect pointing gestures on-line, we have to analyze the observation sequence each time a new frame has been processed. The length of the BHE-phases varies strongly from one gesture to another. Therefore, we classify not only one, but a series of subsequences $s_{1..n}$, each one starting at a different frame in the past and ending with the current frame t_0 (see also [5]). The lengths of the sequences are chosen to be within the minimum/maximum length of a pointing gesture. For each of the phase models, we search for the subsequence $\hat{s}_{B,H,E}$ that maximizes the probability of being produced by the respective model. As $P(\hat{s}|M_0)$ represents the probability, that $\hat{s}$ is *not* part of a pointing gesture, we use it to normalize the phase-models output probabilities:

$$\begin{aligned} \hat{s}_{B,H,E} &= argmax\ logP(s_{1..n}|M_{B,H,E}) \\ P_{B,H,E} &= logP(\hat{s}_{B,H,E}|M_{B,H,E}) - logP(\hat{s}_{B,H,E}|M_0) \end{aligned} \tag{1}$$

In order to detect a pointing gesture, we have to search for three subsequent time intervals that exhibit high output probabilities P_B, P_H and P_E. Ideally, the respective model would significantly dominate the other two models in its interval. But as Fig. 2 shows, M_H tends to dominate the other models in the course of a gesture. That is why we detect a pointing gesture whenever we find three points in time, $t_B < t_H < t_E$, so that

$$\begin{aligned} P_E(t_E) > P_B(t_E) \wedge P_E(t_E) > 0 \\ P_B(t_B) > P_E(t_B) \wedge P_B(t_B) > 0 \\ P_H(t_H) > 0 \end{aligned} \tag{2}$$

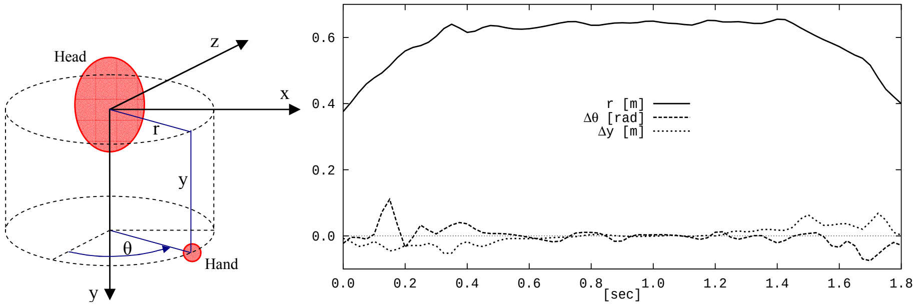

Fig. 3. The hand position is transformed into a cylindrical coordinate system. The plot shows the feature sequence of a typical pointing gesture.

3.2 Features

We evaluated different transformations of the feature vector, including cartesian, spherical and cylindrical coordinates[1]. In our experiments it turned out that cylindrical coordinates of the hands (see Fig. 3) produce the best results for the pointing task. The radius r represents the distance between hand and body, which is an important feature for pointing gesture detection. Unlike its counterpart in spherical coordinates, r is independent of the hand's height y. The origin of the coordinate system is set to the center of the head, to achieve invariance with respect to the person's location. Since we want to prevent the model from adapting to absolute hand positions – as these are determined by the specific pointing targets within the training set – we use the *deltas* (velocities) of θ and y instead of their absolute values. The final feature vector is $(r, \varDelta\theta, \varDelta y)$.

3.3 Estimation of the Pointing Direction

We explored two different approaches to estimate the direction of a pointing gesture: 1) the line of sight between head and hand and 2) the orientation of the forearm. The estimate of the pointing direction is based on the mean value of the head and hand measurements (resp. forearm measurements) within the hold phase of the respective gesture.

In order to identify the orientation of the forearm, we calculate the covariance matrix C of the 3D-pixels within a 20cm radius around the center of the hand. The eigenvector v^1 with the largest eigenvalue (first principal component) of C denotes the direction of the largest variance of the data set. As the forearm is an elongated object, we expect v^1 to be a measure for the direction of the forearm (see Fig. 4). This approach assumes that no other objects are present within the critical radius around the hand, as those would influence the shape of the point set[2].

[1] See [11] for a comparison of different feature vector transformations for gesture recognition.

[2] We found that in the hold phase, this pre-condition is satisfied, as the distance between hand and body and between hand and target object is generally sufficient.

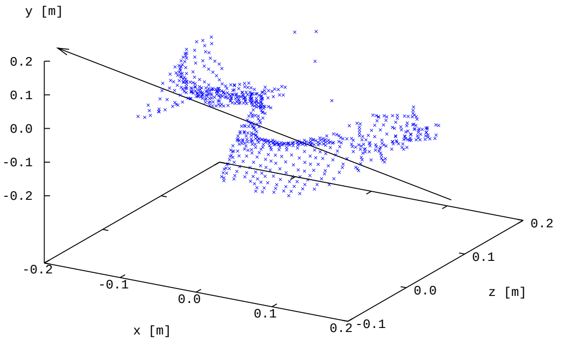

Fig. 4. The first principal component (depicted by an arrow) of the 3D-pixel cloud around the hand is used as an estimate for the forearm orientation.

4 Experiments and Results

In order to evaluate the performance of our system, we prepared an indoor test scenario with 8 different pointing targets (see Fig. 5). Ten test persons were asked to imagine the camera was a household robot. They were to move around within the camera's field of view, every now and then showing the camera one of the marked objects by pointing on it. In total, we captured 206 pointing gestures within a period of 24 min.

4.1 Pointing Direction

The head-hand line and the forearm line were evaluated on hand-labeled H-phases in order to avoid errors caused by the gesture detection module. Nevertheless, an error was induced by the stereo vision system as the camera's coordinates did not comply perfectly with the manual measurements of the target positions. Table 1 summarizes the results. The good results of the head-hand line indicate that most people in our test set intuitively relied on the head-hand line (and not the forearm line) when pointing on a target. The test persons were pointing with an outstretched arm almost every time, thus reducing the potential benefit even of a more accurate forearm measurement[3].

	Avg. error angle	Target identified
Head-hand line	14.8°	99.1%
Forearm line	42.8°	69.6%

Table 1. Accuracy of the pointing direction: a) average angle between the extracted pointing line and the ideal line between hand and target, b) the percentage of gestures for which the correct target (1 out of 8) was identified.

[3] Unlike the relatively stable head position, the forearm measurements vary strongly during the H-phase.

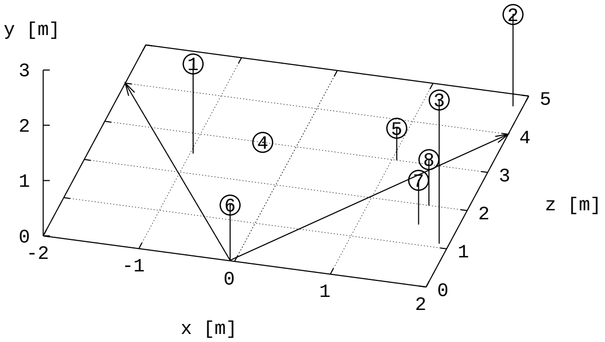

Fig. 5. Positions of the 8 targets in the test scenario. The minimum distance between two targets was 50cm. The arrows depict the camera's field of view.

4.2 Gesture Detection

Two measures were used to determine the quality of the gesture detection:

- the detection rate (*recall*) is the percentage of pointing gestures detected correctly,
- the *precision* of the gesture detection is the ratio of the number of correctly detected gestures to the total number of detected gestures (including false positives).

We performed the evaluation with the *leave-one-out* method to make sure that the models were evaluated on sequences that were not used for training. Here, we measured the quality of the extracted pointing direction using the head-hand line on automatically detected H-phases. See Table 2 for the results.

While the detection rate is similar in both cases (88%), the person-dependent test set has a lower number of false positives compared to the person-independent test set, resulting in a higher classification accuracy. In addition, the estimation of the pointing direction is more accurate in the person-dependent case, so that 97% of the targets were identified correctly. This indicates that it is easier to locate the H-phase correctly when the models are trained individually for each subject. However, even in the person-independent case, 90% of the targets were identified correctly.

	Detection rate (Recall)	Precision	Avg. error angle	Target identified
person-dependent	88.2%	89.3%	12.6°	97.1%
person-independent	87.6%	75.0%	20.9°	89.7%

Table 2. Evaluation of the quality of pointing gesture detection. The person-independent results are the average results on ten subjects. For the person-dependent case, average results on three subjects are given (see text for details).

5 Conclusion

We have demonstrated a real-time[4] 3D vision system which is able to track a person's head and hands robustly, detect pointing gestures, and to estimate the pointing direction. By using dedicated HMMs for different gesture phases, high detection rates were achieved even on defective trajectories. In an evaluation, our system achieved a gesture detection rate of 88%. For 90% (97% person-dependent) of the gestures, the correct pointing target could be identified. For estimating the pointing direction, we compared the line of sight between head and hands and the forearm orientation. With an average error of 14.8°, the head-hand line turned out to be a good estimate for the pointing direction.

Acknowledgements

We would like to thank Christian Fuegen for insightful discussions on the use of HMMs. Also thanks to everybody who participated in our data collection. This research is partially supported by the German Research Foundation (DFG) as part of the Sonderforschungsbereich 588 "Humanoide Roboter".

References

1. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: Real-Time Tracking of the Human Body. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 1997.
2. Azarbayejani, A., Pentland, A.: Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. Proceedings of 13th ICPR, 1996.
3. Darrell, T., Gordon, G., Harville, M., Woodfill, J.: Integrated person tracking using stereo, color, and pattern detection. IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, 1998.
4. Starner, T., Pentland, A.: Visual Recognition of American Sign Language Using Hidden Markov Models. M.I.T. Media Laboratory, Perceptual Computing Section, Cambridge MA, USA, 1994.
5. Becker, D.A.: Sensei: A Real-Time Recognition, Feedback and Training System for T'ai Chi Gestures. M.I.T. Media Lab Perceptual Computing Group Technical Report No. 426, 1997.
6. Wilson, A.D., Bobick A.F.: Recognition and Interpretation of Parametric Gesture. Intl. Conference on Computer Vision ICCV, 329-336, 1998.
7. Jojic, N., Brumitt, B., Meyers, B., Harris, S., Huang, T.: Detection and Estimation of Pointing Gestures in Dense Disparity Maps. IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, France, 2000.
8. Konolige, K.: Small Vision Systems: Hardware and Implementation. Eighth International Symposium on Robotics Research, Hayama, Japan, 1997.
9. Yang, J., Lu, W., Waibel, A.: Skin-color modeling and adaption. Technical Report of School of Computer Science, CMU, CMU-CS-97-146, 1997.
10. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. IEEE, 77 (2), 257-286, 1989.
11. Campbell, L.W., Becker, D.A., Azarbayejani, A., Bobick, A.F., Pentland, A.: Invariant features for 3-D gesture recognition. Second International Workshop on Face and Gesture Recognition, Killington VT, 1996.

[4] The system runs at 10-15 FPS on a 2.4GHz Pentium PC.