

A STATISTICAL UPPER BODY MODEL FOR 3D STATIC AND DYNAMIC GESTURE RECOGNITION FROM STEREO SEQUENCES

Ara V. Nefian, Radek Grzeszczuk, Victor Eruhimov

Intel Corporation
 Microprocessor Research Labs
 Santa Clara, CA 95052
 {ara.nefian, radek.grzeszczuk, victor.eruhimov}@intel.com

ABSTRACT

This paper describes a hidden Markov model-based static and dynamic 3D gesture recognition system. The shape and position of the hands, segmented and tracked using a novel 3D statistical model for the upper body in stereo sequences, are used as observation vectors. The upper body model allows for accurate 3D localization of the hands in the presence of partial occlusions, self occlusions and different illumination conditions. The accuracy of our approach is reflected by the performance of our 3D gesture based editing system, that reaches 96% over 12 dynamic gestures and four static gestures.

1. INTRODUCTION

In recent years gesture recognition became an active research area with many applications for human computer interfaces [5], [10], sign language understanding [8], industrial control [6] or entertainment [2], [9]. A robust gesture recognition system must be able to recognize gestures independent of the scene, the user's pose, or the illumination conditions.

In this paper we describe a gesture-based 3D editing system that is able to interpret both static (gestures determined by the hand shape) [7], [1], and dynamic gestures (gestures determined by the hands trajectories) [5], [10], [8], [2]. Unlike previous approaches [10], in our system both static and dynamic gestures are modeled using hidden Markov models (HMM). A novel 3D statistical model is used to segment and track the upper body from stereo sequences. The accurate location of the hands in the 3D space generates a robust set of observations for the HMM-based static and dynamic gesture recognition system. The use of dense disparity maps together with colors increases considerably the robustness of our system to variations in illumination conditions. It also reduces the inherent depth ambiguity present in 2D images and therefore enables accurate segmentation under partial occlusions and self-occlusions. Recently, consumer-level stereo cameras are becoming more commonplace and the performance of personal computers is approaching the threshold where stereo computation can be done at reasonable frame rates.¹ As a result, robust techniques based on the use of stereo images and the depth information, have been already considered for various applications such as pointing [4], tracking [3], or static gesture recognition [1].

¹For example, stereo camera produced and sold by Point Grey Research, Inc. running on 1.5GHz PentiumTM 4 can compute 320x240 disparity maps at 11 frames per second.

2. THE UPPER BODY MODEL

In this section we present an upper body model that describes the foreground pixels in the image. The foreground pixels represent the pixels in the image for which the depth information is available and that are closer to the camera than a fixed distance threshold. This disparity-based segmentation is significantly more robust to non-stationary background or variations in illumination than color based segmentation methods. Figure 1(b) shows the result of the disparity-based background segmentation for the image in Figure 1(a). The statistical upper body model for the foreground



Figure 1: The original image and the same image after the disparity-based background subtraction.

pixels consists of a set of planar components describing the torso and the arms and a set of Gaussian components representing the head and the hands. Throughout the paper, we refer to the parameters of the m th planar components as π_m and the parameters of the n th Gaussian components as β_n . In addition we will refer to the set of planar and Gaussian components as the states of the upper body model $\Omega = \{\pi_m, \beta_n\}$, $m = 1, 2, 3$ and $n = 1, 2, 3$. The foreground observation vector $\mathbf{O}_{i,j}$ corresponding to the pixel in the i th row and j th column in the image consists of the three dimensional position of the pixel as obtained from the disparity maps $\mathbf{O}_{i,j}^d = \{x, y, z\}_{i,j}$ and its color in the image space $\mathbf{O}_{i,j}^c$. We found that using the hue value extracted from the HSV color space is sufficient to robustly describe the data and keeps a low computational complexity of the model. Formally, $\mathbf{O}_{i,j} = [\mathbf{O}_{i,j}^d, \mathbf{O}_{i,j}^c]$ is obtained through the concatenation of $\mathbf{O}_{i,j}^d$ and $\mathbf{O}_{i,j}^c$.

The probability of the observation vector $\mathbf{O}_{i,j}$ given one of the Gaussian components is:

$$P(\mathbf{O}_{i,j}|\beta) = \frac{1}{(2\pi)^{D/2}|\mathbf{C}|^{1/2}} \exp[-\frac{1}{2}(\mathbf{O}_{i,j} - \underline{\mu})^T \mathbf{C}^{-1}(\mathbf{O}_{i,j} - \underline{\mu})] \quad (1)$$

where the mean vector $\underline{\mu}$ and the covariance matrix \mathbf{C} are the parameters of the Gaussian components $\beta = (\underline{\mu}, \mathbf{C})$, and $D = 4$ is the size of the observation vector.

Since the color distribution and the 3D position can be considered independent random variables, the probability of the observation vectors $\mathbf{O}_{i,j}$ given the planar components can be decomposed as:

$$P(\mathbf{O}_{i,j}|\pi) = P(\mathbf{O}_{i,j}^d|\pi)P(\mathbf{O}_{i,j}^c|\pi) \quad (2)$$

In our method, for simplification, $P(\mathbf{O}_{i,j}^c|\pi)$ is described by a uniform distribution over the entire range of hue values [1, ..., 255]. The probability of the observation $\mathbf{O}_{i,j}^d$ given the planar component π is defined by the following *planar* pdf:

$$P(\mathbf{O}_{i,j}^d|\pi) = \frac{1}{\sqrt{2\pi\sigma_z^2}} \exp\left(-\frac{(z_{ij} - (ax_{ij} + by_{ij} + c))^2}{2\sigma_z^2}\right) \quad (3)$$

From the above equation it can be seen that the planar pdf describes in fact a Gaussian distribution with mean $\mu = ax_{ij} + by_{ij} + c$ and variance σ_z^2 . Unlike the Gaussian distribution, the mean μ for the planar distribution varies depends on the coordinates of the observation vector (i, j) . Throughout this paper, the parameters of the planar components are defined as $\pi = (a, b, c, \sigma_z^2)$.

Assuming that all the observation vectors in the foreground region F are independent, the probability of the sequence of foreground observation vectors $\underline{\mathbf{O}}_F$ given the upper body model Ω is defined as:

$$P(\underline{\mathbf{O}}_F|\Omega) = \prod_{i,j \in F} \left\{ \sum_{m=1}^3 P_{0,\pi_m}(\mathbf{O}_{i,j})P(\mathbf{O}_{i,j}|\pi_m) + \sum_{n=1}^3 P_{0,\beta_n}(\mathbf{O}_{i,j})P(\mathbf{O}_{i,j}|\beta_n) + u_{i,j} \right\} \quad (4)$$

where $u_{i,j}$ is a uniform distribution that models the image noise and P_{0,π_m} and P_{0,β_n} are the a priori probabilities of the planar and Gaussian states of the upper body model.

3. THE ESTIMATION OF THE UPPER BODY MODEL PARAMETERS

The optimal set of parameters for the upper body model, described in Equation 4, are obtained through the EM algorithm by setting the derivatives of $E\{P(\underline{\mathbf{O}}_F|\Omega) \log P(\underline{\mathbf{O}}_F|\Omega)\}$ with respect to the model parameters Ω to zero. In the E step of the algorithm the a posteriori probabilities of the states of the model $s = \{\beta_k, \pi_k, k = 1, 2, 3\}$ given the observed data are computed as follows:

$$\gamma_{i,j}(s) = \frac{P_{0,s}(\mathbf{O}_{i,j})P(\mathbf{O}_{i,j}|s)}{P(\underline{\mathbf{O}}|\Omega)} \quad (5)$$

In the M step, the re-estimated parameters of the Gaussian components, are obtained from:

$$\underline{\tilde{\mu}} = \sum_{all\ i,j \in F} \frac{\gamma_{i,j}(\beta)\mathbf{O}_{i,j}}{\sum_{all\ s} \gamma_{i,j}(s)} \quad (6)$$

$$\tilde{\mathbf{C}} = \sum_{all\ i,j \in F} \frac{\gamma_{i,j}(\beta)(\mathbf{O}_{i,j} - \underline{\tilde{\mu}})(\mathbf{O}_{i,j} - \underline{\tilde{\mu}})^T}{\sum_{all\ s} \gamma_{i,j}(s)} \quad (7)$$

Since in our approach the color observation has a uniform distribution for the planar components it can be disregarded from the estimation of the planar component parameters. The re-estimated parameters for the planar components $\tilde{\pi} = \{\tilde{a}, \tilde{b}, \tilde{c}, \tilde{\sigma}_z\}$ are obtained by solving the following M-step equations:

$$\tilde{a} = \frac{C_{yy}C_{xz} - C_{yz}C_{xy}}{C_{yy}C_{xx} - C_{xy}^2} \quad (8)$$

$$\tilde{b} = \frac{C_{yz} - \tilde{a}C_{xy}}{C_{yy}} \quad (9)$$

$$\tilde{c} = \mu_z - \tilde{a}\mu_x - \tilde{b}\mu_y \quad (10)$$

$$\tilde{\sigma}_z^2 = \sum_{all\ i,j \in F} \frac{(z - \tilde{a}x - \tilde{b}y - \tilde{c})\gamma_{i,j}(\pi)}{\sum_{all\ s} \gamma_{i,j}(s)} \quad (11)$$

where the mean vector $\underline{\mu}_\pi = [\mu_x, \mu_y, \mu_z]^T$, and the covariance matrix \mathbf{C}_π

$$\mathbf{C}_\pi = \begin{pmatrix} C_{xx} & C_{xy} & C_{xz} \\ C_{yx} & C_{yy} & C_{yz} \\ C_{zx} & C_{zy} & C_{zz} \end{pmatrix} \quad (12)$$

are obtained from:

$$\underline{\mu}_\pi = \sum_{all\ i,j \in F} \frac{\gamma_{i,j}(\pi)\mathbf{O}_{i,j}^d}{\sum_{all\ s} \gamma_{i,j}(s)} \quad (13)$$

$$\mathbf{C}_\pi = \sum_{all\ i,j \in F} \frac{\gamma_{i,j}(\pi)(\mathbf{O}_{i,j}^d - \underline{\mu})(\mathbf{O}_{i,j}^d - \underline{\mu})^T}{\sum_{all\ s} \gamma_{i,j}(s)} \quad (14)$$

The EM algorithm is repeated until convergence, i.e. until $P(\mathbf{O}_F|\Omega)$ at consecutive iteration falls below a convergence threshold.

4. INITIALIZATION OF THE UPPER BODY MODEL

Since the EM algorithm, applied in the previous section to our upper body model, is in essence a local optimization algorithm, its convergence to the global solution depends heavily on the initial estimate of the model parameters. To obtain a robust set of initial estimates of the parameters, a simplified model, derived from the upper body model is applied locally to an image region R . The simplified image model used for the estimation of the parameters of each state of the upper body model assumes that all pixels in the image region R , are generated by either the state s of the upper body model or by a "residual" class of uniform distribution $u_{i,j}$. Assuming that the foreground observation vectors in region R are independent random variables, the probability of the sequence $\underline{\mathbf{O}}_R$ in a region R is given by:

$$P(\underline{\mathbf{O}}_R) = \prod_{all\ i,j \in R} \{P(\mathbf{O}_{i,j}|s) + u_{i,j}\} \quad (15)$$

The parameters of the model state s are re-estimated using the EM algorithm. In the E step the a posteriori probability $\gamma_{i,j}(s)$ is computed from

$$\gamma_{i,j}(s) = \frac{P(\mathbf{O}_{i,j}|s)}{P(\mathbf{O}_{i,j}|s) + u_{i,j}} \quad (16)$$

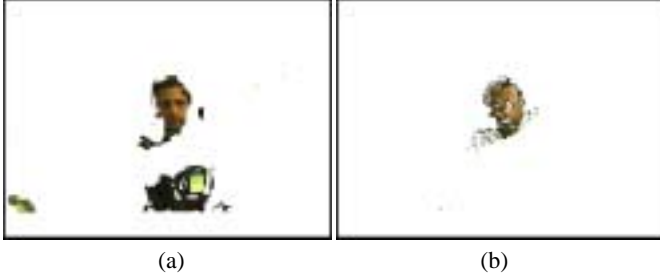


Figure 2: Results of torso (a) and head (b) segmentation.

In the M step the Gaussian and the planar components are re-estimated using the equations Equations 6, 7 and respectively Equations 8- 14. Given the re-estimated state parameters all pixels for which $P(\mathbf{O}_{i,j}|s) > u_{i,j}$, are assigned to the state s . The initialization step is in essence a sequence of two-class classification problems, with a well defined order, repeated for each component of the model. The data assigned to the residual class becomes the input to the next classification step where it is re-assigned to the next body component or becomes a part of the new residual class. This process is repeated until all the data is classified or until all the upper body components are initialized.

4.1. Torso Segmentation

The first component to be segmented is the torso plane. The position of the torso determines, based on the natural structure of the upper body, the regions of search for the remaining components of our model. Figure 2(a) shows an example of the torso segmentation obtained via the EM algorithm for the simplified image model applied to all the foreground pixels. It can be seen that the arm is well rejected from the torso plane, which justifies in fact our approach of using planar distribution for torso modeling instead of Gaussian blobs. From the same example it can be seen that the head and some regions of the background are included in the torso plane. This is due to the fact that although a planar model describes better the torso than a Gaussian blob, it lacks the size constraints of a Gaussian distribution. One essential condition for the convergence of the above EM algorithm to the correct set of parameters is that the torso represents the largest region of the upper body. Under a large variety of situations, excluding strong occlusions of the torso by the arms, this condition is met.

4.2. Head segmentation

The parameters of the head are initialized by applying the EM algorithm for the simplified Gaussian model to the region above the torso. However, it is often possible that the head is included in the torso plane (Figure 2.a) and the area above the torso contains a small number of noisy points. In this case, the system looks for the head in the upper region of the torso. The size of the region where the simplified Gaussian model is applied is calculated from the distance and orientation of the torso plane from the camera, and the average size of heads in real coordinates. Figure 2(b) illustrates the head segmentation results for the original image in Figure 1(a).

4.3. Arm Segmentation

The regions of search for left and right arm consist of pixels on the left and right side of the torso center that were not previously assigned to the torso or head. Although a more sophisticated model



Figure 3: Results of the left arm (a) and left hand (b) segmentation.

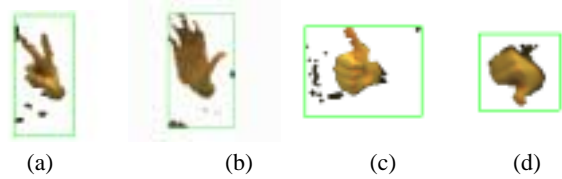


Figure 4: Example of hand segmentation results for four static gestures: (a) "v" sign, (b) stop, (c) thumb up, (d) thumb down.

(including a set of two linear density functions) can better describe the arms, we found that for our application, the planar components model the arms with reasonable accuracy while preserving the natural degrees of freedom in the arm motion. Figure 3.(a) illustrates the results of the left arm segmentation via the EM algorithm for the simplified planar components. As expected the number of pixels assigned to the right arm fall below the noise threshold, and consequently the right arm is not found.

4.4. Hand Segmentation

Several approaches to hand gesture recognition use a priori information about the skin color to detect the hands or face in an image [10], [5]. However, these approaches often fail in environments characterized by strong variations in illumination conditions. Our approach to hand segmentation is performed in the following steps. First, all the foreground pixels, not assigned to the head or torso and that have hue values close to the head hue statistics are assigned to the hand regions. In the second step, the parameters of the left and right hand are estimated from the EM algorithm for the simplified Gaussian model applied to the left and right side regions of the torso. The EM algorithm is initialized using the statistics of the hands obtained in the first step. The results of the hand segmentation algorithm for the foreground pixels in Figure 1(b) are shown in Figure 3(b). As expected, the number of points assigned to the right hand fall below the noise threshold. The accuracy of the hand segmentation, critical in static hand gesture recognition, is improved by assigning to the hand regions the pixels for which no depth information was computed but have hue values close to the statistics of the face and are in the 2D vicinity of the centers of the hands. The result of the hand segmentation is exemplified in Figure 4. Small noise regions are further removed by means of connected component analysis.

5. TRACKING THE UPPER BODY MODEL

The initial parameters of the upper body, obtained individually as described in the previous sections, provide a robust segmentation of the upper body. Over consecutive frames, the parameters of the

upper body model are tracked by estimating them simultaneously through the EM algorithm (Equations 5- 14). The a priori probabilities $P_{0,\pi_m}(\mathbf{O}_{i,j})$ and $P_{0,\pi_m}(\mathbf{O}_{i,j})$ of the observation vectors given each of the states of the model are calculated for the model parameters estimated from the previous frame through a Kalman predictor. The pixels for which

$$P(\mathbf{O}_{i,j}|s_p) > \max_{\text{all } k \neq p} \{ \max_{k \neq p} P(\mathbf{O}_{i,j}|s_k), u_{i,j} \} \quad (17)$$

are assigned to component s_p . If the number of pixels assigned to a body component fall below a noise threshold, the system decides that the body component is not visible in the current frame, and its parameters are re-initialized in the following frame as described in Section 4.

6. GESTURE RECOGNITION

HMMs emerged as a popular tool for the classification of dynamic gestures because of their flexibility in modeling of one dimensional signals while preserving the essential structure of the hand gestures. In this paper, we present an HMM-based recognition system for both dynamic and static gesture recognition. The recognition is carried in two stages. In the first stage the observation vectors of each example consist of the 3D velocities of the hands obtained from the upper body tracking. In this stage the static gestures are considered as one category of dynamic gestures. The recognition is carried out via the Viterbi algorithm and the gestures are classified as either one of the dynamic gestures or as one static gesture. In the second stage of our recognition system, the sequences previously assigned to the class of static gestures are further classified as one of the categories of static gestures. In this stage, the observation sequence consist of the first six normalized moments of the hand [1]. Each category of static gestures is modeled using a HMM, and the recognition is carried via the Viterbi algorithm. A DigiClopsTM camera system [11] was used to acquire stereo sequences of 12 dynamic and four static gestures. The dynamic gestures represent translations and rotations in the image plane and in the plane perpendicular to the image plane. The static gestures used in our experiment are shown in Figure 4. The sequences were captured over the period of two months and therefore vary in illumination and the environment settings. All the sequences were written to the disk drive at 15 fps at the resolution of 320x240. The disparity maps for all the frames were computed off line. In total, we captured 640 sequences of dynamic and static gestures, i.e., 40 examples for each gesture. Each of the dynamic gestures is modeled by a continuous, five states left-to-right states HMM. No skip states are allowed and each state is modeled by a mixture of three Gaussian density functions. Each gesture was trained using 30 sequences and tested on the remaining 10 sequences. The recognition rate, obtained from testing our system on 160 examples (10 examples per gesture) is 96%.

7. CONCLUSIONS AND FUTURE WORK

This paper presents a 3D statistical model for the upper body from stereo sequences, and shows the performance of the model in the context of an HMM-based static and dynamic 3D gesture recognition system. The parameters of our statistical model for the upper body are initialized and then tracked over consecutive frames using an EM algorithm derived for our specific model. Unlike other

tracking systems, that require a user guided initialization, our approach for upper body segmentation makes use of a minimal set of assumptions of the relative position of the user to the camera. The use of the dense disparity maps together with colors in our statistical framework improves the performance of the system by making the segmentation robust to illumination changes and by providing the full three dimensional motion data for our model of the upper body. The accuracy of the upper body modeling is revealed by the performance of the recognition system which achieves 96% on 12 dynamic and four static gestures. We believe that the high accuracy of the gesture recognition system presented in this paper will lead to a lot of interesting and stimulating work on natural unobtrusive vision-based user interfaces.

Further research will be directed towards building an articulated upper body model and a more refined model of the arms. Further we will consider the integration of the continuous parametric gesture recognition under the framework described in this paper.

8. REFERENCES

- [1] R. Grzeszczuk, G. Bradski, M.H. Chu, and J.Y. Bouguet. Stereo based gesture recognition invariant to 3D pose and lighting. In *International Conference on Computer Vision and Pattern Recognition*, pages 826–833, 2000.
- [2] Y. Iwai, H. Shimizu, and M Yachida. Real-time context-based gesture recognition using HMM automaton. In *International Workshop on Recognition, Analysis, and Tracking of Faces in Real-Time Systems*, pages 127–134, 1999.
- [3] N. Jovic, B. Brumitt, B. Meyers, S. Harris, and T. Huang. Tracking self-occluding articulated objects in dense disparity maps. In *International Conference on Computer Vision*, pages 123–130, 1999.
- [4] N. Jovic, B. Brumitt, B. Meyers, S. Harris, and T. Huang. Detection and estimation of pointing gestures in dense disparity maps. In *International Conference on Face and Gesture Recognition*, pages 468–475, 2000.
- [5] S. Marcel, O. Bernier, J-E Viallet, and D. Collobert. Hand gesture recognition using input-output hidden markov models. In *International Conference on Automatic Face and Gesture Recognition*, pages 456–461, 2000.
- [6] S. Muller, S. Eickeler, and G. Rigolli. Crane gesture recognition using pseudo 3-D hidden Markov models. In *International Conference on Automatic Face and Gesture Recognition*, pages 398–402, 2000.
- [7] J. Segen and S. Kumar. Fast and accurate 3D gesture recognition interface. In *Fourteenth International Conference on Pattern Recognition*, volume 1, pages 86–91, 1998.
- [8] T. Starner, J. Weaver, and A. Pentland. A wearable computer based american sign language recognizer. In *First International Symposium on Wearable Computers*, pages 130–137, 1997.
- [9] T. Watanabe and M. Yachida. Real-time gesture recognition using KL expansion of image sequence. In *International Conference on Intelligent Robots and Systems*, pages 973–979, 1997.
- [10] H-S. Yoon, B-W. Min, j. Soh, Y-I. Bae, and H.S Yang. Human computer interface for gesture-based editing system. In *IEEE International Conference on Computational Cybernetics and Simulation*, volume 5, pages 4232–4235, 1997.
- [11] Point Grey Research, DigiClops Stereo System, <http://www.ptgrey.com/products/digiclops>.