3-D Articulated Pose Tracking for Untethered Diectic Reference

David Demirdjian and Trevor Darrell
MIT AI Lab, Cambridge MA 02139
demirdji@ai.mit.edu

Abstract

Arm and body pose are useful cues for diectic reference– users naturally extend their arms to objects of interest in a dialog. We present recent progress on untethered sensing of articulated arm and body configuration using robust stereo vision techniques. These techniques allow robust, accurate, real-time tracking of 3-D position and orientation. We demonstrate users' performance with our system on object selection tasks and describe our initial efforts to integrate this system into a multimodal conversational dialog framework.

1 Introduction

The ability to speak and point at an interface makes many practical interaction tasks much easier for users. Since the seminal work of Bolt's "put-that-there" system [5], it has been known that a real-time system to integrate body part pose estimation with spoken language processing would have many useful applications.

To date, most methods for integrating pose tracking with conversational dialog systems have relied on tethered interfaces. Virtual reality-based sensors (e.g., data gloves and magnetic position systems) were the first practical technique for tracking body configuration, and were successfully applied to multimodal interfaces for tasks such as map exploration [16]. Schemes with explicit markers attached to hands or fingers have also been proposed, as in systems for optical motion capture in computer animation. Unfortunately, the use of attached wires or markers has prevented these systems from being generally usable by casual users.

Untethered approaches to finger and hand tracking using contour and/or skin color detection have been popular techniques, but are limited to planar interactions [13, 18, 14]. An early system for interacting with virtual characters tracked hands and detected pointing gestures based on body silhouette and skin color cues, but was limited to gestures where the user simply pointed to the left or right [9].

Recently a system for 3-D tracking of hand and face features using stereo color cues was developed [3]. This system was successfully applied to detecting and classifying hand gestures in a conversational system [21, 6]. While the system was in real-time, it relied on an explicit initialization step (the user placed hands in a canonical configuration), and could sense only coarse "blob" features. Although it could estimate the relative position of hands and faces, it could not independently sense arm orientation (unless the user wore a short sleeved shirt). Hence it was of limited use in tracking natural pointing gestures, although it was able to recognize parametric gestures defined by the relative position of both hands [21].

To track natural pointing gestures, we wish to have a method that can track the pose of articulated arms. Approaches to track articulated models in monocular image sequences have been proposed. Due to the high dimensionality of the model, many researchers investigated stochastic optimization technics such as particle filtering [19, 20]. Though promising, these approaches are very time-consuming (typically requiring 1000 samples to track simultaneously) and cannot yet be implemented for real-time purposes.

Stereo correspondence can provide shape estimates that capture arm orientation; low-cost, real-time systems for dense stereo have recently become available [1, 2, 11]. An early effort to detect pointing gestures with real-time stereo used a generative mixture model to infer arm orientation [15]; this system worked well for gestures with a fully extended arm which could be modeled using two coarse shape "blobs". This system could not accurately sense arm configurations where the arm was not fully extended, nor could it sense rotations that did not change the apparent shape (but may change it's texture or appearance).

We have developed a system that uses motion stereo analysis to track pose in real-time. Our work is similar to [10]. We rely on 3-D shape estimates from stereo correspondence techniques and an approach based on the ICP (Iterative Closest Point) algorithm [4]. Our main contribution consists in a framework for ICP that implicitly satisfies joint constraints while requiring less computation.

In the following sections we present a method for track-

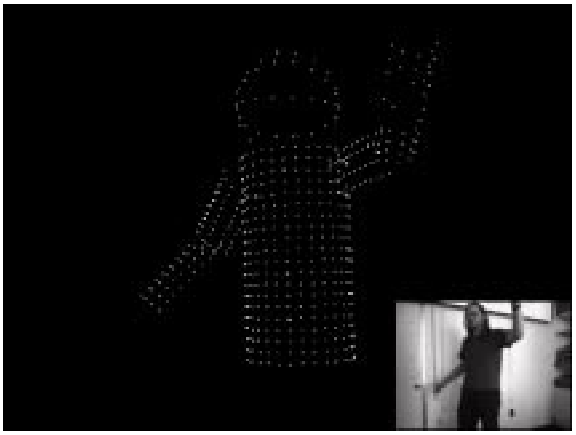

Figure 1. Articulated model for the user's upper body part.

ing articulated structures, and demonstrate its use in allowing interactive pointer control and diectic reference with arm gestures. Finally we describe the initial integration of our system with a conversational dialog system which can resolve utterance-level multimodal pointing references.

2 Arm tracking

In this section, we are interested in estimating pointing gestures and therefore track the arms of a user.

In order to model human bodies, we use a 3D cylindrical model of articulated appearance, as shown in Figure 1: limbs (head, torso, arms, forearms) are modeled as rigid bodies connected by spherical joints.

We have developed an algorithm for estimating articulated motion based on rigid motion estimates of the articulated model's constituent parts. We use the well-known ICP algorithm to coarsely align two clouds of 3D points and estimate an initial rigid motion between body parts [4, 7]. We have developed an algorithm that combines the ICP algorithm with a joint constraint reinforcement step. The advantage of our approach is that the computation of the articulated motion is performed on reduced size equation systems though the articulated model has a lot of d.o.f..

2.1 ICP-based articulated body tracking

Since we are interested in tracking articulated figures, we must add joint constraints to our estimation process. When tracking arms, the standard ICP algorithm [4] is applied to each limb $\mathcal{L}_k$ giving a motion transformation δ_k and associated covariance Λ_k.

Let $\mathbf{R}$ and $\boldsymbol{t}$ be the rotation and translation associated with a motion transformation δ. Here we assume small motions[1] and therefore rotations $\mathbf{R}$ can be approximated at the first order by $\mathbf{R} = \mathbf{I}_3 + [\boldsymbol{w}]$ where $[\boldsymbol{w}]$ denotes the antisymmetric matrix associated with vector $\boldsymbol{w}$. As a consequence, motion transformation δ are parameterized such that:

$$\delta = \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{t} \end{pmatrix}$$

It is clear that the set of motion transformations $\{\delta_k\}$ does not necessarily satisfy the spherical joints constraint. A correct set of motion transformations $\{\delta'_k\}$ that satisfies the spherical joints constraint is found by minimizing:

$$E^2 = \sum_k (\delta'_k - \delta_k)^\top \Lambda_k^{-1} (\delta'_k - \delta_k) \qquad (1)$$

subject to (spherical joint) constraints $\delta'_i(M_{ij}) = \delta'_j(M_{ij})$, where $\{M_{ij}\}$ are the joints between limbs $\mathcal{L}_i$ and $\mathcal{L}_j$.

Minimizing eq.(1) gives a set of motion transformation $\{\delta'_k\}$, which once applied to the articulated model, minimizes the Euclidean distance between the 3-D points and the articulated model while satisfying the spherical joint constraints. The method used to perform the constraint minimization (1) uses only linear technics and is described in the next sections.

2.2 ICP step

In order to recover the articulated model pose from 3D data, we use an approach based on ICP, a standard 3D registration algorithm. Given a set of 3D data and a 3D model of a rigid object to register, ICP estimates the motion transformation between the 3D model and the rigid object. The algorithm can briefly be described as follows:

1. For each point P_i of the 3D model, find the closest point P'_i in the 3D data. The 3-vector $\overrightarrow{f_i} = \overrightarrow{P_i P'_i}$ is the local displacement between the 3D model and the rigid object.

2. Estimate the motion transformation δ by integrating the local displacement $\overrightarrow{f_i}$ over the entire object.

3. Apply the motion transformation δ to the 3D model.

4. If the error criterion ϵ is less than a threshold then quit, otherwise go to step 1.

The ICP algorithm is applied to each limb $\mathcal{L}_k$, giving a motion transformation δ_k, and its covariance matrix Λ_k

[1] In practice, the tracking system is performed at about 10Hz and the hypothesis of small motions is well satisfied

(which can easily be estimated as in step 2. of the ICP algorithm). Only front-facing points of the limbs $\mathcal{L}_k$ were considered.

As stressed in the previous section, the set of motion transformation $\{\delta_k\}$ does not necessarily satisfy the spherical joints constraint. A correct set of motion transformation $\{\delta'_k\}$ that satisfy the spherical joints constraint is found by minimizing eq.(1).

2.3 Enforcing joint constraint

In this section we describe how the spherical joint constraints are implicitly enforced.

Let M_{ij} be a spherical joint between the rigid bodies $\mathcal{L}_i$ and $\mathcal{L}_j$. Let δ'_i and δ'_j be the respective motion transformation applied to the rigid bodies $\mathcal{L}_i$ and $\mathcal{L}_j$. Let $\mathbf{R}'$ and $\boldsymbol{t'}$ be the rotation and translation associated with a motion transformation δ'.

The spherical joint constraint on M_{ij} can be written:

$$\begin{array}{ll} & \delta'_i(M_{ij}) = \delta'_j(M_{ij}) \\ \Rightarrow & (\mathbf{R}'_i - \mathbf{R}'_j)M_{ij} + \boldsymbol{t}'_i - \boldsymbol{t}'_j = 0 \\ \Rightarrow & [\boldsymbol{w}'_i - \boldsymbol{w}'_j]M_{ij} + \boldsymbol{t}'_i - \boldsymbol{t}'_j = 0 \\ \Rightarrow & -[M_{ij}](\boldsymbol{w}'_i - \boldsymbol{w}'_j) + \boldsymbol{t}'_i - \boldsymbol{t}'_j = 0 \end{array} \quad (2)$$

Let Δ' be the articulated motion transformation written as:

$$\Delta' = \begin{pmatrix} \boldsymbol{w}'_1 \\ \boldsymbol{t}'_1 \\ \vdots \\ \boldsymbol{w}'_N \\ \boldsymbol{t}'_N \end{pmatrix}$$

Let $\mathbf{S}_{ij}$ the 3x(6N) matrix defined by:

$$\mathbf{S}_{ij} = (0_3 \ldots \underbrace{-[M_{ij}]}_{i} \underbrace{I_3}_{i+1} \ldots 0_3 \ldots \underbrace{[M_{ij}]}_{j} \underbrace{-I_3}_{j+1} \ldots 0_3)$$

Eq.(2) equivalates to:

$$\mathbf{S}_{ij}\Delta' = 0 \quad (3)$$

Similar equations can be written for each joint constraint. By stacking eqs.(3) into a single matrix $\mathbf{\Phi}$, the spherical joint constraints are simultaneously expressed by the equation:

$$\mathbf{\Phi}\Delta' = 0 \quad (4)$$

Eq.(4) implies that the articulated motion transformation Δ' lies in the kernel of the matrix $\mathbf{\Phi}$. Let K be the size of $kernel\{\mathbf{\Phi}\}$ and $\boldsymbol{v}_k$ be a basis of $kernel\{\mathbf{\Phi}\}$. In our study the basis $\boldsymbol{v}_k$ is estimated from $\mathbf{\Phi}$ using a SVD-based approach and is orthogonal. There exists a set of parameters λ_k such that Δ' can be written:

$$\Delta' = \lambda_1 \boldsymbol{v}_1 + \ldots + \lambda_K \boldsymbol{v}_K \quad (5)$$

Let Δ'_{red} be a vector and $\mathbf{V}$ a matrix such that:

$$\Delta'_{red} = (\lambda_1 \ldots \lambda_M)^\top \quad \mathbf{V} = (\boldsymbol{v}_1 \ldots \boldsymbol{v}_M)$$

Eq.(5) can be rewritten:

$$\Delta' = \mathbf{V}\Delta'_{red} \quad (6)$$

2.4 Articulated motion estimation

Let Δ be the global motion transformation estimated by applying the standard ICP algorithm to each of the rigid bodies. As stressed in Section 2.1, Δ does not satisfy the joint constraints. Let Λ a block-diagonal matrix such that: $\Lambda = diag(\Lambda_1, \Lambda_2, \ldots)$. Eq.(1) gives:

$$\begin{array}{ll} E^2 & = (\Delta' - \Delta)^\top \Lambda^{-1} (\Delta' - \Delta) \\ & = (\mathbf{V}\Delta'_{red} - \Delta)^\top \Lambda^{-1} (\mathbf{V}\Delta'_{red} - \Delta) \end{array} \quad (7)$$

By derivation of the previous equation w.r.t. Δ'_{red}, it can be shown that the minimum of E^2 is reached at:

$$\Delta'_{red} = (\mathbf{V}^\top \Lambda^{-1} \mathbf{V})^{-1} \mathbf{V}^\top \Lambda^{-1} \Delta$$

Finally the correct articulated motion Δ' is estimated using eq.(6).

2.5 Pose initialization

The tracking algorithm requires an initial estimate of the body pose. This initialization is provided by a coarse stereo-based multiple-person tracking system developed in our group [8] that gives an estimate of the (x, y) location of multiple people.

Once a person is detected (and tracked), a simple body model is fit to this person. This model consists in 3 cylinders (1 for head+torso, 1 for left arm+forearm, 1 for right arm+forearm) and assumes that people are standing straight arms stretched. In practice, the simple body fit is done as follow. First, the 3 cylinders are incrementally searched in 3-D reconstructed foreground points. Then an EM algorithm is run in order to refine the model estimation.

This initialization procedure does not constraint the user to a particular pose (arms only have to be stretched) and, though simple, gives a correct pose estimation.

2.6 Pointer estimation

The pointer position is computed as the intersection of the screen plane and the line formed by the corresponding forearm. In addition, a Kalman filter is applied to reduce high frequency tracking jitter. This stabilizes the pointer position and compensates for involuntary motion (*e.g.* shaking) and articulated body tracking instability.

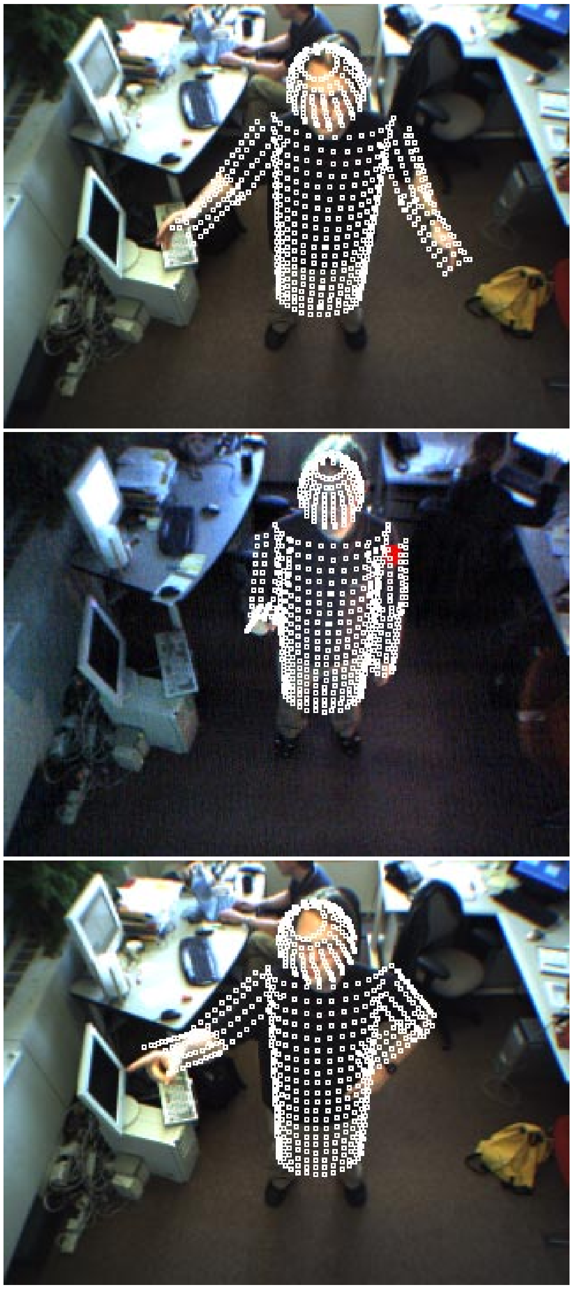

Figure 2. Images of a user with matched articulated body model. In the last two images, the user is pointing at the screen placed below the camera.

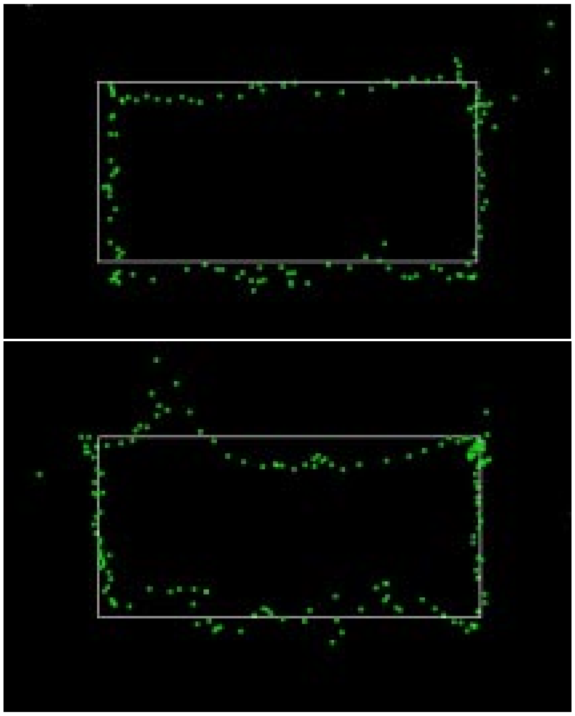

Figure 3. Typical trajectories for the rectangle contour tracing task. (top) user at 1.5m from the screen (bottom) user at 2.5m from the screen

In our initial prototype, clicking events are triggered when the pointer remains still for a certain amount of time (typically 2*sec.*). Though this approach has been found quite useful for detecting clicking events, we are still investigating more natural ways of performing selection tasks (*e.g.* using speech and gesture recognition).

2.7 Target selection experiment

We applied the articulated-body-based pointer to the task of selecting targets on a large projected display. The system consisted of a stereo camera. The stereo images were estimated using [11]. The complete tracking algorithm (stereo + articulated body tracking) was run on a Pentium 4 (2GHz) at 10Hz.

An experiment was run in an interactive room setup. Users were standing about 2.0 meters away from a 2.1m x 1.5m projection screen, subtended a horizontal angle of about 100 degrees and a vertical angle of about 80 degrees. Subjects were asked to perform a target selection task (see Figure 2). The task consisted of pointing at randomly-generated squares. Squares appeared one-at-a-time and remained on the screen until the user has actually been point-

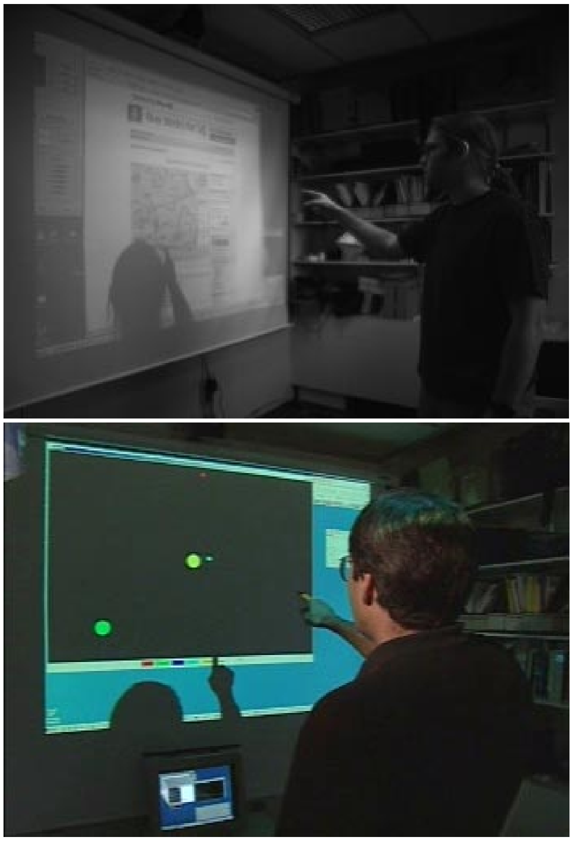

Figure 4. Users interacting with different applications using the integrated system WebGalaxy/pointing detector.

ing at them for a certain period of time τ ($\tau = 2sec.$ in these experiments).

The average pointing accuracy (defined as the average of the distance between the target and the pointer during time τ) estimated during the task was about $20pix.$, what is good enough for target selection.

Another task consisted in tracing the contour of a rectangle drawn on the screen. Figure 3 shows typical pointer trajectories when the user is respectively at 1.5m and 2.5m from the screen.

3 Application

We are integrating our pointing detection system in with the MIT SLS system WebGalaxy [12]. WebGalaxy is a flexible multi-modal user interface system that allows wide access to selected information on the World Wide Web (WWW) by integrating spoken and typed natural language queries and hypertext navigation. Our pointer detector is being used for the hypertext navigation instead of using a mouse. Figure 4 shows users interacting with different applications using the integrated system WebGalaxy/pointing detector. In these applications, events are triggered either by pointing detection (target selection), speech (actions) or both (*e.g.* move-this-there).

4 Discussion

We described an approach for real-time articulated body tracking. This framework uses stereo information only and is therefore robust to illumination dynamic. The articulated body pose provides the orientation of the arms as well as the coordinates of the pointer on a screen.

For direct manipulation tasks such as driving cursors and selecting objects, the articulated body tracking system is accurate enough. The arm tracking system, though less accurate than the stereo head tracking developed in our group [17], appeared in practice to be a more natural way to select targets on a screen. We believe this type of system will be an important module in designing perceptual interfaces for screen interaction and cockpit applications by providing natural human-computer interaction.

Currently the head and arm tracking systems have been implemented and evaluated as separate applications. We are merging the implementations and expect to evaluate them jointing on the WebGalaxy application this summer.

References

[1] *Pt Grey Inc.* http://www.ptgrey.com.

[2] *Videre design*. http://www.videredesign.com.

[3] A. Azarbayejani, C. Wren, and A. Pentland. Real-time 3-d tracking of the human body. In *IMAGE'COM*, 1996.

[4] P.J. Besl and N. MacKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256, 1992.

[5] R. Bolt. Put-that-there: Voice and gesture at the graphics interface. In *ACM SIGGRAPH*, 1980.

[6] J. Cassell. Nudge nudge wink wink: Elements of face-to-face conversation for embodied conversational agents. In *Embodied Conversational Agents*, 2000.

[7] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *Image and Vision Computing*, pages 145–155, 1992.

[8] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb. Plan-view trajectory estimation with dense

stereo background models. In *2001 International Conference on Computer Vision*, 2001.

[9] T. Darrell, P. Maes, B. Blumberg, and A. Pentland. A novel environment for situated vision and behavior. In *IEEE Workshop on Visual Behaviors*, 1994.

[10] Quentin Delamarre and Olivier D. Faugeras. 3d articulated models and multi-view tracking with silhouettes. In *ICCV (2)*, pages 716–721, 1999.

[11] D. Demirdjian. *E-stereo: Real-time dense stereo processing*. http://www.ai.mit.edu/ demirdji/download/.

[12] D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue. Galaxy: A human-language interface to on-line travel information. In *Int'l Conference on Spoken Language Processing '94*, 1994.

[13] D. Hall, C. Le Gal, J. Martin, O. Chomat, and J. L. Crowley. Magicboard: A contribution to an intelligent office environment. In *Intelligent Robotic Systems*, 2001.

[14] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *ECCV'98*, 1998.

[15] N. Jojic, M. Turk, and T.S. Huang. Tracking articulated objects in dense disparity maps. In *International Conference on Computer Vision*, pages 123–130, 1999.

[16] D. Koons, C. Sparrell, and K. Thrisson. Integrating simultaneous input from speech, gaze and hand gestures. *Intelligent Multimedia Interfaces, ed. by M. Maybury, MIT Press*, pages 257–276, 1993.

[17] L.P. Morency and T. Darrell. Stereo tracking using icp and normal flow. In *Int. Conf. on Pattern Recognition*, 2002.

[18] K. Oka, Y. Sato, and H. Koike. Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2002.

[19] Hedvig Sidenbladh, Michael J. Black, and David J. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV (2)*, pages 702–718, 2000.

[20] Cristian Sminchisescu and Bill Triggs. Covariance scaled sampling for monocular 3d body tracking. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*. IEEE Computer Society Press, Dec 2001.

[21] A. Wilson and A. Bobick. Parametric hidden markov models for gesture recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999.