Model-Based Estimation of 3D Human Motion

Ioannis Kakadiaris, *Member*, *IEEE*, and
Dimitris Metaxas, *Senior Member*, *IEEE*

Abstract—This paper presents the formulations and techniques that we have developed for the three-dimensional, model-based, motion estimation of human movement from multiple cameras. Our method is based on the spatio-temporal analysis of the subject's silhouette and it has the advantage that the subject does not have to wear markers or other devices. We present tracking results from experiments involving the recovery of complex motions in the presence of significant occlusion.

Index Terms—Motion estimation, human motion estimation, deformable models, model-based tracking, physics-based modeling.

1 Introduction

IMAGE-BASED, three-dimensional, human shape estimation and motion tracking are important and challenging research problems and their importance stems from numerous applications such as: 1) posture and gait analysis for training athletes and physically challenged persons, 2) human body, hands, and face animation, and 3) automatic annotation of human activities in video databases. Although for some applications (e.g., determining if a person is moving towards or away from you), information about the movement of the centroid of the silhouette is adequate, for others, detailed shape and motion information for the body parts is required. For example, the use of complex models such as those of humans or other articulated objects and the modeling of their motions is often necessary. In such cases, even the most skilled modelers and animators are not able to accurately reproduce the respective shapes and motions. If synthesized motion is to be compelling, we must create virtual actors that appear realistic when they move. Other applications, such as vision-based computer interfaces [27], [30], ergonomics, anthropometry, and human factors design require additional analysis of the data to facilitate certain tasks or activities. An example of such an analysis is the performance measurement of athletes, as well as patients with psychomotor disabilities and the rapid prototyping of rehabilitation aids [21].

Currently, motion tracking is performed using mechanical, electro-magnetic, and image-based techniques. A comprehensive review of the mechanical and electromagnetic systems can be found in [10]. Image-based systems can be lumped into three broad categories: techniques that use active markers, techniques that use passive markers, and techniques that do not use markers. Since placing markers on the subjects is cumbersome and alters their natural motion, a nonintrusive sensory method based on vision is preferable. Indeed, there is a need for accurate measurement of the three-dimensional motion of moving body parts without interfering with the dynamics of the moving bodies. In the following, we offer only a very brief review of the marker-free image-based human tracking methods due to space limitations. The reader is referred to [17], [1], [11] for comprehensive reviews.

Recently, a number of alternative solutions to tracking of human bodies have been proposed. Gavrila [12] formulated the pose-recovery as a search problem and he employed a hierarchical decomposition approach to cope with the high dimensionality of the search space. Wachter [31] presented a model-based approach, where a prior human body model is fit to the image by an iterated extended Kalman filter using both edge and region measurements. Bregler [4] developed a region-based estimation framework (using twists and exponential maps), whose cost function is based on the optic flow. In Yamamoto's work [33], tracking is performed by estimating the pose increment of the body parts from multiple images by assuming small increments in pose change, while in [34], scene constraints are employed to reduce the model's degrees-of-freedom. Wren [32] coupled a 3D dynamic model of the human body with models of human behavior to track end-effectors like the head and the hands using a blob-based approach. Cham [5] developed a probabilistic multiple-hypothesis framework for tracking articulated objects, where the key insight is the representation and tracking the modes in the posterior state density function. Delamarre [6] presented a force-based method for tracking humans, which is based on the ideas of force-based tracking presented in [18] and elaborated in this paper. Deutscher [7] developed a modified particle filter for search in high dimensional configuration spaces that resulted in very robust human motion tracking. Finally, Ormoneit [25] presented an analysis method based on statistical learning and Bayesian tracking. However, for many of the systems cited above, no information is provided about the accuracy of the tracking. Concerning model-based tracking of self-occluding articulated objects, Rehg and Kanade [29] present a novel representation of self-occlusion in configuration space by using a kinematic model to predict occlusions and windowed appearance templates to track partially occluded objects. In our work, occlusion is computed dynamically based on the predicted motion and the tracking is based on the occluding contours. The singularity problems that arise in tracking articulated objects were studied in [24]. However, the effect of the shape of the parts was not taken in consideration in that study. Concerning initial posture estimation, methods have been presented that use either one [4], [22], or multiple cameras [2], [6], [12], [16], [18]. Recently, we proposed a technique [3] for simultaneously estimating a human's anthropometric measurements (up to a scale parameter) and pose from a single image that can be used for initializing video-based, three-dimensional, human motion tracking. Concerning model acquisition, existing approaches use models of the human body whose parts are either approximated with simple shapes and their dimensions have been manually measured [13], [29], or models whose shape and/or dimensions have been determined based on camera input data. In this second category, methods have been developed to obtain models of human body parts from multiple cameras [14], [19], [15], [28] or range data [8]. Specifically, to overcome problems that stem from using approximate shape models for the estimation of 3D human motion, we have previously developed [19] a method for estimating the shape of a subject's body parts that allows the creation of an anthropometrically correct graphical model of a subject. The method is based on fusing observations (from multiple cameras) from a subject performing a set of motions according to a protocol designed to reveal the structure of the human body.

In most applications, the human subject is available. Therefore, to capture the motion of an unencumbered subject and to overcome the problems due to variations among subjects in the shape of their body, we propose the following approach: First, automatically acquire a three-dimensional model of the subject using computer vision techniques [19]. Then, employ this model to

- *I. Kakadiaris is with the Department of Computer Science, University of Houston, Houston TX 77204. E-mail: ioannisk@uh.edu.*
- *D. Metaxas is with the Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104. E-mail: dnm@central.cis.upenn.edu.*

Manuscript received 14 Mar. 1998; revised 24 June 1999; accepted 7 June 2000.
Recommended for acceptance by R. Szeliski.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107679.

track the unconstrained movement of a subject's body parts by using multiple cameras and by actively selecting the subset of the cameras that provide the most informative views. In this paper, we develop a formal framework for tracking the motion of human body parts from one or multiple cameras based on information extracted from the occluding contours. In particular, our system has a motion analysis and a motion playback part. The analysis part is based on the spatio-temporal analysis of the subject's silhouette from image sequences acquired simultaneously from multiple cameras. The input to the analysis is a sequence of grayscale images taken from multiple views. The output of the analysis part are the three-dimensional positions and orientations of the subject's body parts at each time instant, from which the trajectories, velocities, and accelerations can be computed. The motion playback part couples the estimated motion parameters with a customized physics-based graphical model of the participant resulting in real-time human motion animation. The motion playback system is presented in detail at [20]. Since the system is noninvasive, the person under observation (the subject) can move freely without interference from the system and without markers annoying him/her. In the current implementation, we assume that the background of the scene is static, the subject is wearing tight fitting clothes, no significant bulging of the muscles is observable and that the parts being tracked are detectable.

The main contributions presented in this paper are the extensions of deformable model-based tracking framework to: 1) track 3D articulated objects from 2D contours using the algorithm in Section 3 and 2) handle articulated objects with parts which are possibly occluding each other using information from multiple cameras. In particular, we present an active approach to the ordering/selection of a set of cameras that is based on the visibility of each part and the observability of its predicted motion from a given camera. Our system advances the state of the art in human motion tracking because:

1. our method is based on the use of occluding contours and it obviates the need for markers or other devices,
2. the accuracy of tracking of body parts in live video data is comparable to the accuracy achieved using markers,
3. the animated graphical model is anthropometrically correct and resembles the subject,
4. the ordering of the cameras based is accomplished in an active and time varying fashion,
5. since we are using multiple cameras, we can mitigate the difficulties arising due to occlusion among body parts and body movements that are ambiguous from one view can be disambiguated from another view, and
6. the motion capture is based on the whole apparent contour rather than just a few points.

We used our system to capture the motion of the upper-body extremities of several subjects performing complex three-dimensional motions in the presence of significant occlusion. However, there is no theoretical reason that restricts the application of our system to upper body only.

This paper is organized as follows:[1] Section 2 briefly reviews the deformable model framework while Section 3 presents the elements of the motion analysis. The effectiveness of the tracking algorithms along with a case study is demonstrated in Section 4.

2 Deformable Models and Kinematics

To estimate the motion parameters of a subject's body parts, we follow the deformable model-based approach [23]. In that approach, the two-dimensional image data apply forces to a model. Based on these forces the model translates and rotates to minimize the discrepancy between its projection and the image data. The model $\mathbf{M}$ extracted during the model building phase consists of the number of the body parts n, their connectivity and their respective shape expressed in terms of model-centered coordinate frames ϕ_i $(i = 1, \ldots, n)$ and a noninertial coordinate frame Φ. The models used in this work are three-dimensional surface shape models. For simplicity, let ${}^{\Phi}\mathbf{x}$ denote the position of a point j on part i with respect to the frame Φ. Then its position, with respect to the camera coordinate system C, can be expressed as: ${}^{\Phi}\mathbf{x} = {}^{\Phi}\mathbf{t} + {}^{\Phi}_{C}\mathbf{R}\,{}^{C}\mathbf{x}$, where ${}^{\Phi}\mathbf{t}$ is the position of the origin O_c of the camera frame C with respect to the frame Φ, ${}^{\Phi}_{C}\mathbf{R}$ is the matrix that encapsulates the orientation of C with respect to Φ and ${}^{C}\mathbf{x} = (x, y, z)^{\top}$ is the position of the point j on part i with respect to the frame C. Under perspective projection, the point ${}^{C}\mathbf{x}$ projects into an image point ${}^{I}\mathbf{x} = ({}^{I}x, {}^{I}y)^{\top}$ according to the following equations: ${}^{I}x = \frac{x}{z}f, \quad {}^{I}y = \frac{y}{z}f$, where f is the focal length of the camera. By taking the time derivative we get ${}^{I}\dot{\mathbf{x}} = \mathbf{H}\,{}^{C}\dot{\mathbf{x}}$ where

$$\mathbf{H} = \begin{bmatrix} f/z & 0 & -(x/z^2)f \\ 0 & f/z & -(y/z^2)f \end{bmatrix}.$$

Finally, ${}^{I}\dot{\mathbf{x}} = \mathbf{H}({}^{\Phi}_{C}\mathbf{R}^{-1}\,{}^{\Phi}\dot{\mathbf{x}}) = \mathbf{H}\,{}^{\Phi}_{C}\mathbf{R}^{-1}(\mathbf{L}\dot{\mathbf{q}}) = \mathbf{L}_p\dot{\mathbf{q}}$ [17].

3 Human Motion Analysis

To capture the geometric and kinematic parameters of humans in a nonintrusive, fast, and robust fashion, we have constructed an experimental testbed (3D-studio) [17]. The imaging hardware of the 3D-studio consists of three calibrated grayscale cameras placed in mutually orthogonal configuration [17]. In particular, the input to the analysis part is a sequence of grayscale images taken from these cameras. The output is the three-dimensional positions and orientations of the subject's body parts at each time instant from which the trajectories, velocities, and accelerations can be computed. Since our method is model-based, a model of the subject being tracked will be used in tracking. Having a priori model of the subject will allow prediction of the occlusions among body parts.

Modeling Phase: The first phase of our system is the model building phase, that is, we first extract an accurate shape model of the body parts of the subject [19]. For the experiments described in this paper, the subject is asked to perform a set of motions that permit the acquisition of the anthropometric dimensions of the subject's upper and lower arms. By employing the Human Body Part Identification Algorithm [19], we extracted the three-dimensional shape of the upper and lower arm of the subject.

Motion Estimation: We begin by presenting first the principal steps in motion estimation. In the pseudo-code below, t_k denotes time, $startT$ and $endT$ are the starting and ending times, respectively, $\mathbf{u} = (\mathbf{q}^{\top}, \dot{\mathbf{q}}^{\top})^{\top}$ where $\mathbf{q}$ is the vector of generalized coordinates that describe the articulated model $\mathbf{M}$ of the subject with n parts. The elements of $\mathbf{q}$ are the positions and orientations of each modeled body part of the subject's body. The vector $\mathbf{q}$ along with the vector $\dot{\mathbf{q}}$ completely describe the shape and motion of the articulated model of the subject. The matrix $\mathbf{P}_{start}$ represents the uncertainty of the state and is part of the Kalman filter described later. The motion estimation process proceeds as follows:

procedure MotionEstimation($startT$, $endT$, $\mathbf{M}$, $\mathbf{u}_{start}$, $\mathbf{P}_{start}$)
$t_k \leftarrow startT$; $\hat{\mathbf{u}}(t_k|t_k) \leftarrow \mathbf{u}_{start}$; $\mathbf{P}(t_k|t_k) \leftarrow \mathbf{P}_{start}$;
while($t_k < endT$)
  $\{\hat{\mathbf{u}}(t_{k+1}|t_k), \mathbf{P}(t_{k+1}|t_k)\} \leftarrow$ **Predict**($\hat{\mathbf{u}}(t_k|t_k)$, $\mathbf{P}(t_k|t_k)$);
  SynthesizeAppearance($\hat{\mathbf{u}}(t_{k+1}|t_k)$);
  $\{\mathcal{BC}_1, \ldots, \mathcal{BC}_m\} \leftarrow$ **SelectCameras**($\hat{\mathbf{u}}(t_{k+1}|t_k)$, $\mathbf{M}$);

1. Parts of this paper have appeared previously in [18], [20].

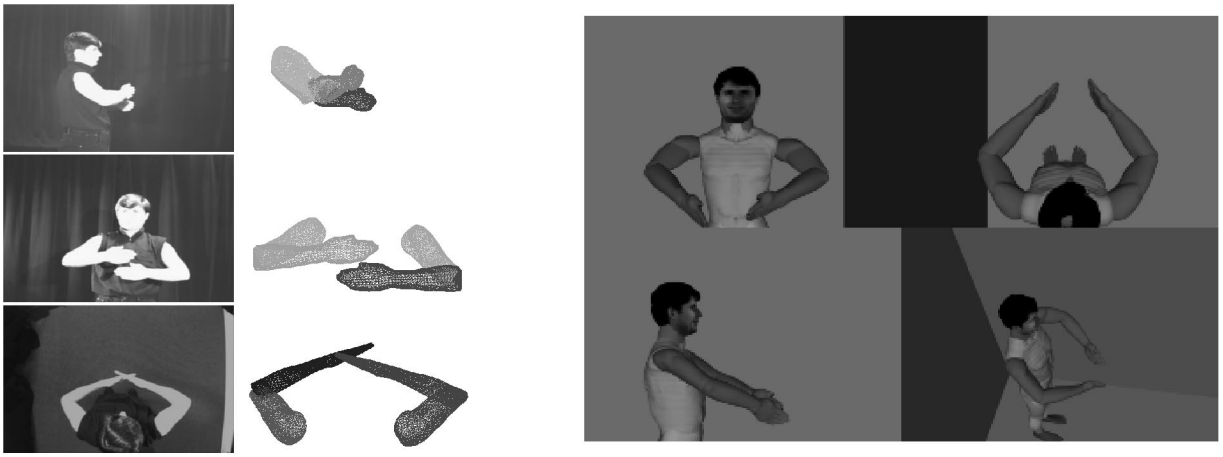

Fig. 1. (a) Side, front, and top views of the subject performing a complex two-arm motion. (b) Recovered three-dimensional pose of the subject's upper and lower arms. (c) The estimated motion parameters for the motion of the subject's arms have been applied to a *customized* graphical model of the subject (four different views).

MatchAndMeasure($\hat{\mathbf{u}}(t_{k+1}|t_k), \{\mathcal{BC}_1, \ldots, \mathcal{BC}_m\}$);
StateUpdate($\hat{\mathbf{u}}(t_{k+1}|t_k)$, $\mathbf{P}(t_k|t_k)$, $\mathbf{P}(t_{k+1}|t_k)$); $t_k \leftarrow t_k + 1$;

The procedure **Predict** takes into account states up to time t_k to make a prediction for the state of the model at time t_{k+1}. The procedure **SynthesizeAppearance** synthesizes the appearance of the graphical model for each camera while the procedure **MatchAndMeasure** establishes the discrepancy between the predicted and the actual appearance. Then, a new state for the model is estimated by the procedure **StateUpdate** so as to minimize these discrepancies. The process is repeated again for the following frames in the image sequence. These were the steps in previous motion estimation methods as well as in [26]. However, since we are using multiple cameras, one of the questions we are called upon to answer is how many views should we employ for tracking. Should we process the occluding contours from all the cameras or from only one? Therefore, we have added to the algorithm the **SelectCameras** procedure that (for each body part) selects the subset of cameras ($m \leq n$) that provide the most informative views for tracking.

Predicting the model's motion: We incorporate the dynamics of our model into a Kalman filter formulation by treating the differential equations of motion as the system model, with uncorrelated modeling error noise $\mathbf{w}(t)$, assumed to be a zero mean white noise process with known covariance i.e., $\mathbf{w}(t) \sim \mathbf{N}(\mathbf{0}, \mathbf{Q}(t))$. Let also the observation vector $\mathbf{z}(t)$ denote time-varying input data. The system model and the measurement model equations for the Extended Kalman Filter take the form: $\dot{\mathbf{u}}(t) = \mathbf{A}\ \mathbf{u}(t) + \mathbf{w}(t)$ and $\mathbf{z}(t) = \mathbf{g}(\mathbf{u}(t), t) + \mathbf{v}(t)$, where $\mathbf{u}(t) = (\dot{\mathbf{q}}(t), \mathbf{q}(t))^\top$ is the vector of state variables, $\mathbf{g}(\mathbf{u}(\mathbf{t}))$ is a nonlinear function which relates the input data to the model's state [17], and

$$\mathbf{A} = \begin{bmatrix} -\mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}.$$

The vector $\mathbf{v}(t)$ represents the uncorrelated measurement error, as a zero mean white noise process with known covariance i.e., $\mathbf{v}(t) \sim \mathbf{N}(\mathbf{0}, \mathbf{V}(t))$. For initial conditions $\mathbf{u}(0) \sim \mathbf{N}(\hat{\mathbf{u}}_0, \mathbf{P}_0)$ and for uncorrelated system and measurement noises (i.e., $E[\mathbf{w}(t)\mathbf{v}(\tau)^\top] = 0$), the state estimation equation is given by:

$$\begin{aligned}\dot{\mathbf{P}}(t) =& \mathbf{A}\ \mathbf{P}(t) + \mathbf{P}(t)\ \mathbf{A}^\top + \mathbf{Q}(t) \\ & - \mathbf{P}(t)\ \mathbf{G}^\top(\hat{\mathbf{u}}(t), t)\ \mathbf{V}^{-1}(t)\ \mathbf{G}(\hat{\mathbf{u}}(t), t)\ \mathbf{P}(t),\end{aligned}$$

where

$$\mathbf{G}(\hat{\mathbf{u}}(t), t) = \left.\frac{\partial \mathbf{g}(\mathbf{u}(t), t)}{\partial \mathbf{u}(t)}\right|_{\mathbf{u}(t)=\hat{\mathbf{u}}(t)}.$$

The entries of matrix $\mathbf{G}$ consist of evaluations of the Jacobian matrix $\mathbf{L}_p$ at the various locations of the projected model points.

Synthesizing the appearance of the model: Based on the predicted position of the shape model, the algorithm synthesizes the appearance of the model to the different cameras. Therefore, the projection of the model nodes (for every part) to the image plane of each camera is computed.

Ordering Phase—Camera Selection: Our goal is to track (using the recovered shape model for the subject) the three-dimensional position and orientation of the subject's body parts. To alleviate the problem of degenerate views and severe occlusion from a specific view, we employ three calibrated cameras placed in a mutually orthogonal configuration. Specifically, we develop a formal methodology to track the motion of human body parts from three sets (or n sets in general) of projected contour sequences, each taken from a different camera. Should we process the occluding contours from all the cameras or from only one? Using portions from all three of them simultaneously to compute forces on a particular part can result into an incorrect solution due to incorrect association of contour points to model points. Therefore, at every step of the computation, we decide which cameras to use by employing the procedure below.

The procedure **ComputePartVisibility** computes the visibility of a subject's body part from a particular camera and the procedure **ComputePredictedMotionObservability** computes the observability of its motion w.r.t the particular camera. For a part i and a camera $\mathrm{C}_k, k \in \{1, 2, 3\}$, we define a visibility index $\mathcal{VI}_{i,j}$ and an observability index $\mathcal{OI}_{i,j}$.

Part Visibility Criterion: We define the visibility index $\mathcal{VI}_{i,j}$, as the ratio of the area of the visible projection of the part i to the image plane of camera C_k to its projected area when we do not take into account the occlusions. According to our definition, a part i is considered visible from camera k if $\mathcal{VI}_{i,k} \geq 0.2$. This criterion is implemented in real time using the hardware of a Silicon Graphics workstation. The visibility index is a function of the shape and pose of the object, the pose of the camera, and the pose of other objects in the scene. To motivate the use of this index, we have performed a simulation whose purpose was to demonstrate the change to the visibility of an object (due to occlusions) as a camera moves around it in a unit sphere (the position of the camera is

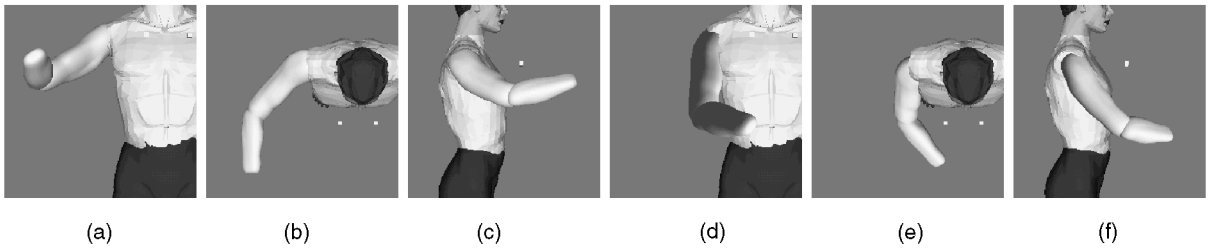

Fig. 2. (a), (b), and (c) Starting and (d), (e), and (f) ending position of the right upper arm and forearm of a graphical model.

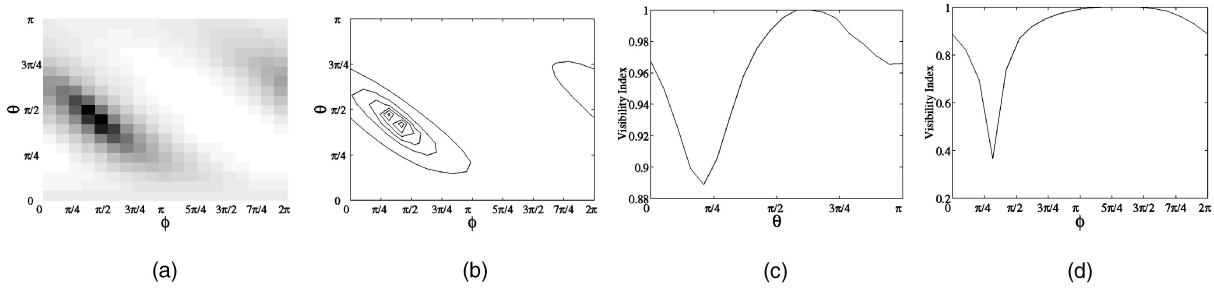

Fig. 3. (a) Intensity map and (b) iso-contour plot of the variation of the visibility index for the right upper arm of the graphical model. (c) Variation of the visibility index along θ for $\phi = 0$. (d) Variation of the visibility index along ϕ for $\theta = 0$.

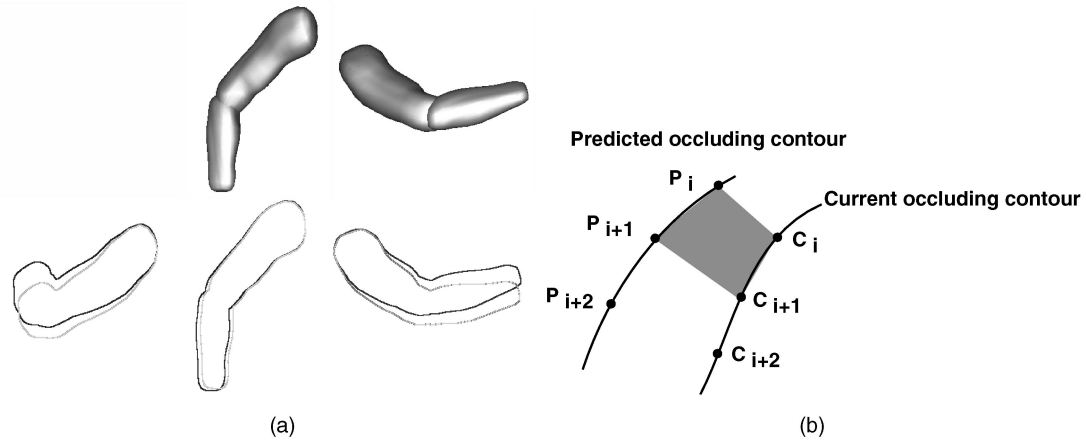

Fig. 4. (a) The same motion in 3D induces different changes in the occluding contours from each of the three cameras. (b) Notation pertaining to the computation of the observability index.

determined by the spherical coordinates ϕ and θ). For example, imagine a sphere enclosing the object and a camera positioned at any point of this sphere looking at the object. Using the EAI Jack® software we have generated synthetic images of an arm at a specific pose (Fig. 2) for different positions of camera around it. Figs. 3a, 3b, 3c, and 3d depict as an intensity value, the value of the visibility index as the camera sweeps the sphere around the graphical model of the human. The white areas correspond to positions of the camera from which the arm is not occluded by other objects present in the scene. It is hardly a surprise that there is a wide variation in the visibility index. Our motivation is just to reinforce that there are zones of high visibility and zones of low visibility. Therefore, if one has the opportunity to choose a camera from a set of cameras, it is relevant to chose the cameras that meet the visibility criteria.

Observability Criterion: The second criterion for view ordering refers to the observability of motion. Let's assume that we observe the movement of the upper arm and the forearm of a graphical model using three cameras. Our algorithm is based on occluding contour information. The occluding contour prior to movement is indicated with purple color and the occluding contour after the movement is indicated with green color. Fig. 4a depicts that the same motion in 3D induces different changes in the occluding contours from the three cameras. This change is a function of the shape and pose of the object, its motion, occlusions, and the pose of the observer. Our objective is to select the subset of cameras in which large change occurs. Once a part is considered visible from a particular camera, we further check if the camera's viewpoint is degenerate given the predicted motion (based on the Kalman Filter) of the part. For every node C_i at the current occluding contour of the model, we determine the nearest point P_i to the predicted occluding contour and we compute the area of the polygon with vertices C_i, C_{i+1}, P_i, and P_{i+1} (Fig. 4b). The sum of these areas is the observability index and summarizes the changes in the apparent contour of an object for a given motion.

To motivate the use of the observability index, we recorded the predicted motion observability index as the graphical model performed the motion detailed in [17]. Figs. 5a, 5b, 5c, and 5d depict, as an intensity value, the value of the observability index as a function of the camera position. The white areas correspond to positions of the camera to which the specific motion for the arm induces the largest change in image coordinates. In addition, this map reflects the axes of symmetry of the object. It gives the axes around which, when the object moves, its apparent contour does not change.

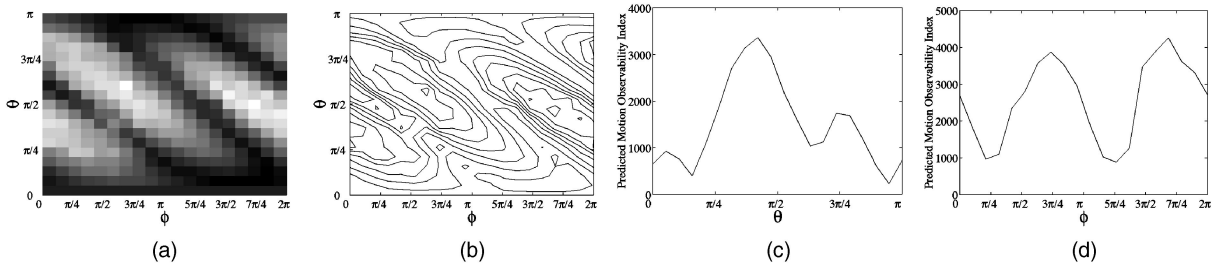

Fig. 5. (a) Intensity map, and (b) iso-contour plot of the variation of the predicted motion observability index for the right upper arm of the graphical model. (c) Variation of the observability index along θ for $\phi = 0$. (d) Variation of the observability index along ϕ for $\theta = 0$.

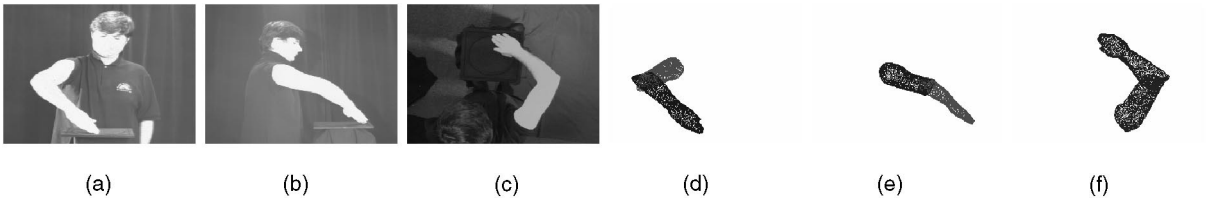

Fig. 6. Tracking of the subject's arm as he slides the tip of his right hand along a circular groove engraved in a glass plate. (d), (e), and (f) Estimated pose of the arm from the three cameras. (a), (b), and (c) of the 3D studio.

Matching and Measurement Phase: Having selected the set of cameras that provide the most information, we employ information from the occluding contours to provide constraints to the estimation of the rotational and translational motion of each of the parts. In physics-based tracking techniques, image data points apply forces to the deformable model. Since each point on the occluding contour is the projection of a 3D model point, the algorithm has to find a match between the points on the occluding contour and the points on the model. In other words, the algorithm has to select for each image data point, the model node it should apply forces to. To achieve this matching between an occluding contour point and a model point, we invoke a theorem from geometry that relates points on the occluding contour of an object to points on the surface of the object itself (see also [9]). More specifically, if ${}^{\Phi}\mathbf{x}_{i,j}$ is a point on the surface of a part whose projection point, ${}^{\mathrm{I}}\mathbf{x}_{i,j}$ on the image plane, lies on the occluding contour of the part, then the part's normal $\mathbf{n}$ at point ${}^{\Phi}\mathbf{x}_{i,j}$ is parallel to the normal vector $\mathbf{n}'$ of the plane defined by the origin of the camera coordinate system, the point ${}^{\mathrm{I}}\mathbf{x}_{i,j}$, and the tangent of the occluding contour at point ${}^{\mathrm{I}}\mathbf{x}_{i,j}$, that is $\mathbf{n}.\mathbf{n}' = 1$. Based on the above theorem, we can establish correspondences between the occluding contour of a part and its model's projection.

Since we estimate the object's parts and their respective shape prior to tracking, we can compute the normal $\mathbf{n}_{i,j}$ (for simplicity $\mathbf{n}_j$) at the each node j of part i. Also, to characterize the variation of the normal over each part's model, we compute for each node m of the part's model the quantity $\epsilon_m = \min_{k \in \mathcal{N}_i}(\mathbf{n}_m.\mathbf{n}_k)$, where $\mathcal{N}_i$ is the set of nodes neighboring node m. Therefore, the variation of each part's i normal is $\epsilon_i = \sum_{m=1}^{n} \frac{1}{n}\epsilon_m$, where n is the number of the nodes of part i. Specifically, assuming that the initial pose of the parts is known, for every occluding contour, we employ the algorithm below:

procedure MatchAndMeasure($\hat{\mathbf{u}}(t_{k+1}|t_k), \{\mathcal{BC}_1, \ldots, \mathcal{BC}_n\}$);
for j in $\{\{\mathcal{BC}_1, \ldots, \mathcal{BC}_n\}$
  Fit2DDeformableModel(j);**MatchPoints**(j,$\hat{\mathbf{u}}(t_{k+1}|t_k)$;
  MeasureDifference($\hat{\mathbf{u}}(t_{k+1}|t_k)$;

The procedure **Fit2DDeformableModel** fits a two-dimensional deformable model to the occluding contour in camera j to obtain a smooth differentiable model of the curve [23]. At every point $\mathbf{z}$ of the occluding contour, we compute the tangent vector ${}^{\mathrm{I}}\mathbf{t}_z$ and the vector ${}^{\mathrm{C}}\mathbf{k}_z$, which is normal to the plane defined by the origin of the camera coordinate system C, point $\mathbf{z}$ and ${}^{\mathrm{C}}\mathbf{t}_z$. The procedure **MatchPoints** computes the correspondence between nodes on the model and points on the occluding contour. For every point $\mathbf{z}$ on the occluding contour, it determines the set of the tessellated shape model nodes $\mathbf{S}_i$ for the part i whose normal ${}^{\phi_i}\mathbf{n}_{i,j}$ (expressed in the model frame ϕ_i) satisfies the relationship: ${}^{\phi_i}\mathbf{n}_{i,j}.{}^{\phi_i}\mathbf{k}_z \geq \epsilon_i$. The point $\mathbf{z}$ of the occluding contour will apply forces to the model node j, which is a member of the set $\mathbf{S}_i$ and whose Euclidean distance from the node is the smallest. The procedure **MeasureDifference** measures the inconsistencies between the synthesized appearance of the model and the actual one. In physics-based tracking terms, it measures the forces that the image points apply to the model. The force that the point $\mathbf{z}$ applies to the node j has two components: $\mathbf{f}_{\mathbf{z}}^{im}(t_{k+1}) = (\mathbf{z} - {}^{\mathrm{I}}\mathbf{x}_{i,j})$ and $\mathbf{f}_{\mathbf{z}}^{3D}(t_{k+1}) = {}^{\mathrm{C}}\mathbf{n}_{i,j} - {}^{\mathrm{C}}\mathbf{k}_z$. The first component measures the difference between the position of the matching points and the second component measures the difference in their normals. After we compute the force assignments, we estimate the new pose parameters of the part based on the extended Kalman filter. The estimation of the motion parameters is achieved by numerically integrating through time the state estimation equation.

4 Results

In all the experiments, the region of interest has been obtained by subtracting the current image from the known background and the outlines have been obtained by applying a variation of the Canny edge detector to the input image sequence.

To demonstrate the performance of our algorithm, we have performed a number of experiments (using both synthetic and real data), which are fully described in [17]. In this paper, due to lack of space, we present the results from three experiments only. The first experiment illustrates the robustness to occlusions. During the modeling phase, the subject did not flex his wrists, therefore the estimated models for the forearms (Fig. 1b) include the wrists. Fig. 1a depicts sample input images from the three cameras when the subject was asked to move his arms in 3D. Fig. 1b depicts the estimated 3D pose of the subject's parts of the arm. As an example related to the selection of cameras, for the frame depicted in Fig. 1a and for the lower right arm the cameras were ordered as follows: top, side, and front camera. Fig. 1c shows four views for three frames of an animation which was created by applying the estimated motion parameters to a customized graphical model of the subject. The second experiment was designed to assess the accuracy and robustness of our tracking scheme. The experimental protocol was the following: a subject was asked to slide the tip of his right hand along a circular groove 5mm $\pm$ 0.5mm wide,

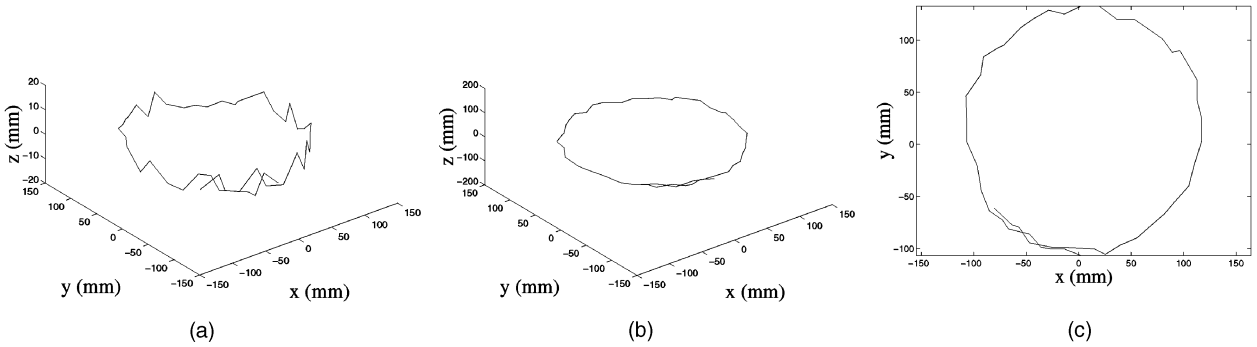

Fig. 7. Three views of the estimated trajectory of the tip of the right hand of the subject.

engraved in a glass plate, thus following a circular trajectory of radius 114.3mm ± 0.5mm (Figs. 6a, 6b, and 6c). After tracking (Figs. 6d, 6e, and 6f) and trajectory reconstruction, an error analysis was performed, which is presented in [17]. Even by taking into account human imprecision while following the groove, the mean error between the ideal and the recovered trajectory of the finger tip is 1.03cm (with std 4.7mm) for observation distance of 2.5m. Thus, the error is 0.4 percent. Fig. 7 depicts the recovered path from several views. In Fig. 7b all the axes have been drawn at the same scale. However, in Fig. 7a the Z axis is drawn at a smaller scale to depict the variation of the path. Fig. 7(c) depicts the path in the transverse plane.

Case Study: As a case study, we utilize this framework for the generation of personalized rehabilitation aids and present results on how head motion analysis is applied to the customized design of a head-controlled feeding device. This work was part of the UPGRADE project being conducted at the GRASP Lab, in collaboration with the AI DuPont Institute. The goal of the project was to merge several state-of-the-art technologies for the rapid prototyping of rehabilitation aids *customized* for a specific physically-challenged user [21].

We have chosen to investigate how a vision-based motion capture and animation system can be used for the acquisition of kinematic and geometric data from motion impaired users, in order to provide measurements to rehabilitation device designers. Accurate motion tracking would be especially welcome since most medical research has focused on very specific data such as force measurements and range of motion information. Because each person presents a unique neuro-physiological picture, it is essential to involve the user in a *customized* design process. While simulations using ideal trajectories can serve as a reasonable base for early prototype design, only real data obtained by observing actual patients constitute sensible inputs to the customizing process of a one-of-a-kind rehabilitation device.

We considered as a testbed the device showed in Fig. 8b, where the three-dimensional head and neck motions of a quadriplegic patient can control a feeding utensil via the extension and retraction of cables (the cables and the attachment to the patient's head are not shown for simplicity). In this case, our goal is to measure how, and to what extent, a given patient can perform the motion controlling the feeding apparatus. The subject is being observed by the set of three CCD cameras of the 3D studio. A deformable model of the subject's head is automatically fitted to the first image of the sequence and then tracked. Fig. 8a shows the trajectory followed by a point on the chin of our test subject. Note that the motion is much more complex than the simple arc one would expect. The kinematic data thus acquired is used as input to a virtual prototyping module, where a tentative design for the feeding device can be simulated and refined. In particular, an optimization procedure was used to compute the optimal mechanism dimensions and the motion coupling required to generate a desired spoon motion based on the recovered user-specific 3D head trajectory [21]. Eventually, a parameterized CAD model of the mechanism can be adjusted to the patient's specific needs and manufactured accordingly (see Fig. 8c). It is crucial to note that our framework is general and equally allows for other type of input motions (e.g., arm, leg, ...) to be tracked and used for other instances of customized manufacturing, such as different types of rehabilitation aids or more generally other man/machine interfaces.

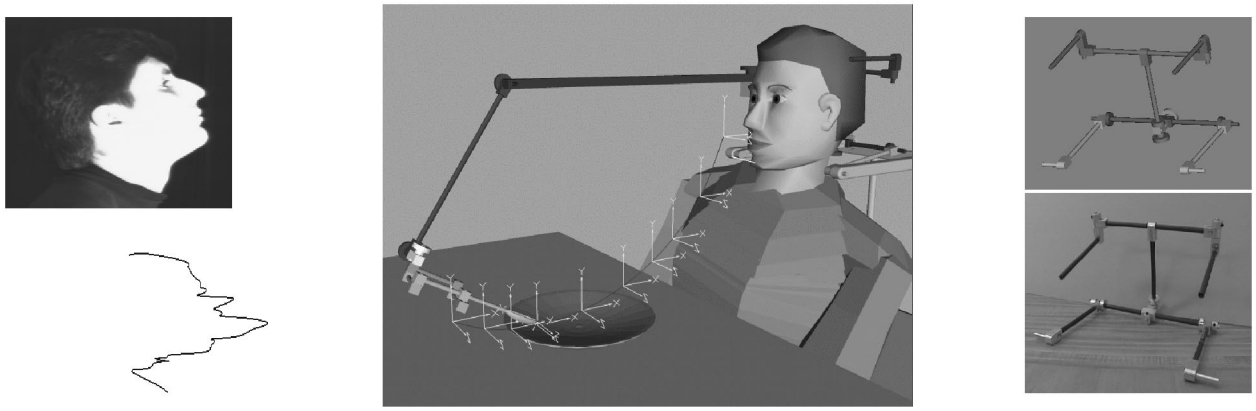

Fig. 8. Application to rehabilitation aid design. (a) Side view of the subject performing the head motion and the resulting trajectory of a point on the chin as recovered by the tracking algorithm. (b) Virtual optimization of an experimental feeding mechanism. (c) Pro Engineer© model for the part of the mechanism coupled to the patient's head and actual early prototype [21].

5 Conclusion

In this paper, we presented a mathematical formulation and implemented a system capable of accurate human motion capture. We provided an analysis of the criteria that allow the selection of the subset of cameras that provide the most information for tracking. In addition, we presented a detailed performance analysis of the motion tracking technique for the case of tracking upper body extremities. Our experiments have demonstrated that we can indeed accurately estimate the motion parameters of a moving subject and we can create animation sequences through the vision-based analysis of the video input.

References

[1] J.K. Aggarwal and Q. Cai, "Human Motion Analysis: A Review," *Proc. IEEE Nonrigid and Articulated Motion Workshop,* pp. 90–102, June 1997.

[2] A. Azarbayejani, C. Wren, and A. Pentland, "Real-Time 3-D Tracking of the Human Body," *Proc. IMAGE'COM 96,* May 1996.

[3] C. Barrón and I.A. Kakadiaris, "Estimating Anthropometry and Pose from a Single Image," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 669–676, June 2000.

[4] C. Bregler and J. Malik, "Tracking People with Twists and Exponential Maps," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 8–15, June 1998.

[5] T.J. Cham and J. Rehg, "A Multiple Hypothesis Approach to Figure Tracking," *Proc. Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 239–245, June 1999.

[6] Q. Delamarre and O. Faugeras, "3D Articulated Models and Multi-View Tracking with Silhouettes," *Proc. 17th Int'l Conf. Computer Vision,* pp. 716–721, Sept. 1999.

[7] J. Deutscher, A. Blake, and I. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 126–133, June 2000.

[8] I. Douros, L. Dekker, and B.F. Buxton, "An Improved Algorithm for Reconstruction of the Surface of the Human Body from the 3D Scanner Data using Local B-Spline Parches," *Proc. IEEE Int'l Workshop Modeling People,* pp. 29–36, Sept. 1999.

[9] J. Feldmar, N. Ayache, and F. Betting, "3D-2D Projective Registration of Free-Form Curves and Surfaces," *Proc. Fifth Int'l Conf. Computer Vision,* pp. 549–556, June 1995.

[10] W. Frey, M. Zyda, R. McGhee, and W. Cockayne, "Off-The-Shelf, Real-Time, Human Body Motion Capture for Synthetic Environments," Technical Report NPSCS-96-003, Computer Science Dept., Naval Postgraduate School, Monterey, Calif., June 1996.

[11] D.M. Gavrila, "The Visual Analysis of Human Movement: A Survey," *Computer Vision and Image Understanding,* vol. 73, no. 1, Jan. 1999.

[12] D.M. Gavrila and L.S. Davis, "3-D Model-Based Tracking of Humans in Action: A Multi-View Approach," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 73–80, June 1996.

[13] L. Goncalves, E. DiBernardom, E. Ursella, and P. Perona, "Monocular Tracking of The Human Arm in 3D," *Proc. Fifth Int'l Conf. Computer Vision,* pp. 764–770, June 1995.

[14] J. Gu, T. Chang, I. Mak, S. Gopalsamy, H.C. Shen, and M.F. Yuen, "A 3D Reconstruction System for Human Body Modeling," *Proc. CAPTECH '98,* N. Magnenat-Thalmann and D. Thalmann, eds., pp. 229–241, Nov. 1998.

[15] A. Hilton, "Towards Model-Based Capture of a Person's Shape, Appearance and Motion," *Proc. IEEE Int'l Workshop Modeling People,* pp. 37–44, Sept. 1999.

[16] S. Iwasawa, J. Ohya, K. Takahashi, T. Sakaguchi, S. Kawato, K. Ebihara, and S. Morishima, "Real-Time, 3D Estimation of Human Body Postures from Trinocular Images," *Proc. IEEE Int'l Workshop Modeling People,* pp. 3–10, Sept. 1999.

[17] I.A. Kakadiaris, "Motion-Based Part Segmentation, Shape and Motion Estimation of Multi-Part Objects: Application to Human Body Tracking," PhD dissertation, Dept. of Computer Science, Univ. of Pennsylvania, Philadelphia, Penn., Oct. 1996.

[18] I.A. Kakadiaris and D. Metaxas, "Model-Based Estimation of 3D Human Motion with Occlusion Based on Active Multi-Viewpoint Selection," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 81–87, June 1996.

[19] I.A. Kakadiaris and D. Metaxas, "3D Human Body Model Acquisition from Multiple Views," *Int'l J. Computer Vision,* vol. 30, no. 3, pp. 191–218, 1998.

[20] I.A. Kakadiaris and D. Metaxas, "Vision-Based Animation of Digital Humans," *Proc. Computer Animation '98 Conf.,* pp. 144–152, June 1998.

[21] V. Kumar, R. Bajcsy, W. Harwin, and P. Harker, "Rapid Design and Prototyping of Customized Rehabilitation Aids," *Comm. ACM,* vol. 39, no. 2, pp. 55–61, Feb. 1996.

[22] H.J. Lee and Z. Chen, "Determination of 3D Human Body Postures from a Single View," *Computer Vision, Graphics, and Image Processing,* vol. 30, pp. 148–168, May 1985.

[23] D. Metaxas and D. Terzopoulos, "Shape and Nonrigid Motion Estimation Through Physics-Based Synthesis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 6, pp. 580–591, June 1993.

[24] D.D. Morris, J.M. Rehg, "Singularity Analysis for Articulated Object Tracking," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 289–296, June 1998.

[25] D. Ormoneit, H. Sidenbladh, M.J. Black, T. Hastie, and D.J. Fleet, "Learning and Tracking Human Motion using Functional Analysis," *Proc. IEEE Workshop Human Modeling, Analysis and Synthesis,* pp. 2–9, June 2000.

[26] J. O'Rourke and N.I. Badler, "Model-Based Image Analysis of Human Motion using Constraint Propagation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 2, no. 6, pp. 522–536, June 1980.

[27] A. Pentland, "Machine Understanding of Human Action," *Proc. Seventh Int'l Forum on Frontier of Telecommunication Technology,* Nov. 1995.

[28] R. Plankers, P. Fua, and N. D'Apuzzo, "Automated Body Modeling from Video Sequences," *Proc. IEEE Int'l Workshop Modeling People,* pp. 45–52, Sept. 1999.

[29] J.M. Rehg and T. Kanade, "Model-Based Tracking of Self-Occluding Articulated Objects," *Proc. Fifth Int'l Conf. Computer Vision,* pp. 612–617, June 1995.

[30] R. Sharma, T.S. Huang, V.I. Pavlovic, "A Multimodal Framework for Interacting with Virtual Environments," *Human Interaction with Complex Systems,* C.A. Ntuen, E.H. Park, and J.H. Kim, eds., 1996.

[31] S. Wachter and H.-H. Nagel, "Tracking of Persons in Monocular Image Sequences," *Proc. IEEE Nonrigid and Articulated Motion Workshop,* pp. 2–9, June 1997.

[32] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *Proc. Second Int'l Conf. Automatic Face and Gesture Recognition,* pp. 51–56, Apr. 1998.

[33] M. Yamamoto, A. Sato, and S. Kawada, "Icremental Tracking of Human Actions from Multiple Views," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 2–7, June 1998.

[34] M. Yamamoto and K. Yagishita, "Scene Constraints-Aided Tracking of Human Body," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 151–156, June 2000.