Complex Human Motion Estimation Using Visibility

Tomoyuki Mukasa*, Arata Miyamoto, Shohei Nobuhara, Atsuto Maki and Takashi Matsuyama
Graduate School of Informatics, Kyoto University,
Sakyo-ku, Kyoto, 606-8501, Japan
* mks@vision.kuee.kyoto-u.ac.jp

Abstract

This paper presents a novel algorithm for estimating complex human motion from 3D video. We base our algorithm on a model-based approach which uses a complete surface mesh of a 3D human model to be matched with 3D video data. This type of method usually works well against partly incomplete input data. However, it fails to estimate what we call "complex motion": where some parts of the body touch each other for a long period. It is because the touching deteriorates the visibility of the neighbouring surface, which causes matching failures.

In order to solve this problem, we introduce a "visibility" measure for each mesh vertex that represents how it is occluded or missed on the observed surface. Using the "visibility" we selectively suppress the outliers caused by low observability while traditional surface matching algorithms try to find corresponding area for the entire surface and cannot converge to the real posture by definition. Our algorithm shows improvements over naive surface matching algorithm on both synthesized and real 3D video.

1. Introduction

The representation of a moving articulated object such as a human body is important in many applications, for example technical analysis of human motion in sports and dance, or production of video content. Motion capture systems are popular and available for such purposes but they require special markers and suits, which is often a burden to the target. Moreover, it is not possible to capture a target's motion and natural appearance simultaneously when markers are used. For these reasons, *vision-based* human motion capture has been an active topic in computer vision in recent years [6].In this paper, in order to realize a scheme for representing the motion of a human body which can be approximated by an articulated rigid body without using any special markers and suits, we propose a *visibility-based* method for human motion estimation.

978-1-4244-2154-1/08/$25.00

1.1. Problem Definition

We use time series volume data as input which we compute from multiple view video (we call it reconstructed 3D shape) in advance since it can provide rich 3D information for accurately estimating human pose. We also utilize a manually constructed 3D human model, consisting of a complete surface and a kinematic model which are given a-priori. We then estimate target's motion by matching the reconstructed 3D shape and the 3D human model frame by frame. Thus, the problem of human motion estimation can be attributed to that of model fitting in which the kinematic model of the 3D human model is fitted to the input so as to minimize the difference between its surface and the reconstructed 3D shape.

1.2. Complexity of Human Motion

The reconstructed surface is usually incomplete due to visibility problem caused by self touching of the body. Low visibility thus deteriorates the matching between a reconstructed 3D shape and the 3D human model. Therefore we evaluate the complexity of human posture in terms of the degree of touching of target's body parts. We then define the complexity of human motion on the basis of complexity of posture while taking the duration into account.

Human motion can be classified into three types: *simple motion*, *semi-complex motion*, and *complex motion*. We here define these three motions as follows. simple motion refers a motion without any contacts beteen body parts. semi-complex motion involves *complex posture*, i.e. some body parts touch with each other. complex motion refers a motion in which the same complex posture lasts for a long period. Although conventional model-based approaches still work well for semi-complex motions, they cannot accurately estimate complex motion due to the cumulative matching failures.

In order to cope with complex motion as well as semi-complex motion, we introduce the visibility-based approach to the model fitting problem in the next section.

1.3. The Visibility-based Method

3D Surfaces Reconstructed from Multiple View Video The 3D shape of an object is represented by primitives, e.g. vertices on surface and their time-variant connectivity. Given the target is represented by a mesh, the mesh topology can generally change from frame to frame. Especially, when two body segments touch each other, the touching surfaces cannot be seen at all (Figure 1). We call this area *invisible surface*. Further, the neighbouring surfaces cannot be observed by some of the cameras (Figure 1). We refer to such regions of surfaces as less-visible.

We assume that less-visible surfaces on a reconstructed 3D shape always occur with invisible surfaces and thus cannot be fully fitted by the 3D human model. Less-visible surfaces are likely to be reconstructed with low accuracy since it is difficult to observe such a partial surface from many of the cameras due to occlusions caused by touching body parts. For these reasons, less-visible surface are not suitable to fit for the 3D human model, It is obvious that the more complex the target motion is, the more areas suffer from the problem due to potential risk of body touching causing reconstruction errors.

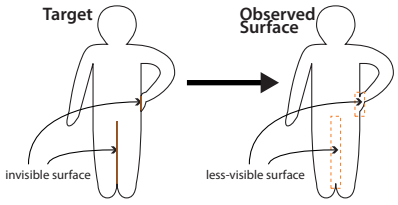

Figure 1: Invisible surface and less-visible surface.

3D Human Model We deal with the human body as the target object in our research and assume that it can be approximated by an articulated rigid body, which is comprised of segmented parts and their time-invariant connectivity. That is, each segmented part is connected to each other at its end points by a joint .

As can be seen in Fig.1, a major difference between the assumed articulated rigid body and the reconstructed 3D shape typically arises at the time of touching between body parts. We can compare the 3D human model and the reconstructed 3D shape from this point of view, and find potential areas in the 3D human model that correspond to less-visible surfaces in the observed 3D model. This is possible by observing the 3D human model with virtual cameras that are placed in the same way as real cameras used in reconstructed 3D shape acquisition. We call these parts of surface *solitary*. Solitary surfaces that are identified by observing the 3D human model virtually are not suitable for fitting with the reconstructed 3D shape.

Visibility-based Model Fitting To estimate complex human motion by model fitting, we propose to consider not only the target's shape but also visibility.

As we already explained, it is difficult to correlate a reconstructed 3D shape with the 3D human model by naive fitting when the target is in a posture with touches between different body parts. It is because of changes of mesh topology on the reconstructed 3D shape which act on less-visible surface in the reconstructed 3D shape, and also on the solitary surface on the 3D human model.

To overcome this difficulty in correlation, we present a method to make both global topologies of the reconstructed 3D shape and the 3D human model similar, by excluding the less-visible surface in the reconstructed 3D shape and the solitary surface in the 3D human model in the fitting process. This topological similarity helps more accurate fitting by removing outliers.

In implementation, we evaluate both reconstructed 3D shape and 3D human model, find less-visible surface and solitary surface and exclude them from fitting. For this evaluation, we introduce a *visibility measure*. For each vertex, we count the number of cameras that are visible from its position, and normalize it by the total number of cameras. We call this value visibility, and evaluate appropriateness of vertices for the fitting process.

After the evaluation, we exclude less observable areas assuming that they are less-visible or solitary surface, and fit the remaining surfaces of 3D human model to that of the reconstructed 3D shape. We call this fitting method *visibility-based model fitting*. It can overcome not only touching surface problem but also reconstruction error problem when it is caused by the lack of adequate observing cameras. By applying this fitting to the entire time series, we can estimate complex human motion.

2. Related Work

2.1. Human Motion Estimation Methods

In this section, we classify related works on human motion estimation from two aspects, i.e. types of input data and the use of 3D human model.

Input Data Many types of input data are used for motion estimation. They can be classified to three levels of data type, monocular image data, multiple image data, and 3D shape data.

The system based on monocular image data [9] is easy to set up, but not desirable to acquire accurate estimation result. Using multiple images [5] we can expect higher capability than a monocular image based system, but cannot use full 3D shape information, 3D normal and curvature on surface for example. To overcome these difficulties, Ogawara

et al.[8] employed a time-series of full 3D shapes of the object estimated from multi-viewpoint videos.

However, a problem in using 3D shape data as input is that the accuracy of 3D shape reconstruction generally affects the final result of motion estimation. In order to handle the problem we introduce a visibility measure for motion estimation. By taking account the visibility measure, we can reduce the influence of reconstruction error since partial surfaces with low accuracy on the reconstructed 3D shape tend to be occluded from every camera. In general, it is desirable that a reconstructed mesh is coherent across the sequence. However, we used a frame-wise mesh reconstruction algorithm in this time because our experiments did not suffered by instability of the mesh reconstruction.

Model-based or Case-based Motion Estimation Methods Model-based approaches use a specific human model given a-priori to fit to observed data. On the contrary, the case-based approaches do not use such a human model but compare observed data and a database constructed beforehand. These approaches can work real time, but cannot cope with the postures that do not have correlated data in the database. To construct a comprehensive database which can cope with every possible posture is impossible.

We adopt the model-based approach because our purpose is to estimate complex human motion including poses which are unlikely to be covered by a database.

2.2. Surface Matching

Iterative Closest Point algorithm As already stated we match a 3D human model to the reconstructed 3D shape. In order to match these data to each other, we first need to find their correspondence for the entire surface. For this purpose, the ICP (Iterative Closest Point) algorithm [1, 2] is widely used. In ICP sets of corresponding vertices are determined based on their Euclidean distance. The algorithm first finds the nearest neighbouring vertex on one side of the mesh for each vertex on another side, then, update model parameters to minimize the sum of distance between each corresponding set of vertices. After that, the process is executed iteratively until the error function converges.

Generally, it is not guaranteed that the nearest neighbouring pair of vertices are a genuinely corresponding pair. However, if the difference of the entire shape of matching meshes is small enough, the model parameter will be optimized when an appropriate initialization is available. We introduce a new approach based on the ICP algorithm for matching from frame to frame, assuming that the difference between temporally neighbouring meshes is relatively small, i.e., the temporal sampling rate of the input volume data is sufficiently high to follow the rapid motion of the target object.

Elimination of Erroneous Matching Many erroneous pairs are included in the pairs of corresponding vertices that are simply obtained from the Euclidean distance. Therefore many improvements for the ICP algorithm have been proposed [10, 3, 4]. They utilize attributes of each vertex or shape characteristics of the mesh to remove erroneous pairs. However, these parameters are not always reliable because they are often affected strongly by the results of 3D shape reconstruction. Moreover, conventional approaches which use only these parameters cannot cope with the case that partial surfaces to be matched are invisible, e.g. when body parts are touching.

In order to overcome this problem, we utilize the target's shape and the camera parameters which are available in our case, and thereby evaluate the visibility measure at each vertex to remove erroneous pairs. The following are the major factors which may cause such errors at each mesh vertex.

1. Noise in 3D shape reconstruction due to insufficiency of cameras visible from the vertex.

2. Lack of vertices in the reconstructed 3D shape in comparison to the 3D human model caused on touching body parts.

All of the factors can be evaluated with the visibility measure on each mesh vertex.

In the reconstructed 3D shape, low observability implies that the vertex is unreliable for matching with since the area in which the vertex parts suffers from reconstruction noise. Such areas are often adjacent to touching area. We label the area which consists of such vertices as less-visible surface. For 3D human model, on the other hand, low observability area has a risk to lose correspondence with the reconstructed 3D shape because of its lack of counterparts and noise in reconstruction process. We call such area as solitary surface.

Hence, we exclude the area with low observability in both reconstructed 3D shape and the 3D human model in the matching process based on the ICP algorithm.

3. Motion Estimation from Observed 3D Shape Sequence

3.1. Algorithm Overview

We denote a sequence of reconstructed 3D shape of the target and a motion parameter, the result of the motion estimation, as $S_T[t]$ and $\mathbf{p}[t](1 \leq t \leq N_t)$, respectively. In these notation, t is time, N_t is the number of frames for which reconstructed 3D shapes are provided, T stands for target, and $\mathbf{p}[t]$ is a posture parameter that is estimated for the reconstructed 3D shape $S_T[t]$ by minimizing the error function which we detail later.

We select the initial pose from arbitrary poses in input data, and call its time as the initial frame. We use $S_T[1]$

computed in the initial frame as the surface of 3D human model S_M while M stands for model (we call this "model mesh"). To initialize the 3D human model, we set the kinematic model manually to model mesh, and construct a "skin-bone model". After the initialization, we fit the 3D human model to the reconstructed 3D shape, $S_T[t](2 \leq t \leq N_t)$, from frame to frame and estimate $\mathbf{p}[t](2 \leq t \leq N_t)$ by minimizing error function, $E(\mathbf{p}; S_M(\mathbf{p}), S_T[t])$, which evaluates the correspondence between model mesh, $S_M(\mathbf{p})$, and reconstructed 3D shape, $S_T[t]$. In this error function, we evaluate the visibility measure for each vertex. For convenience, we occasionally represent the error function by $E(\mathbf{p})$.

The posture estimation process in each frame is summarized as in the following chart.

INPUT $S_T[t]$:reconstructed 3D shape at time t

$\mathbf{p}_{t-1}$:result of posture estimation at time $t-1$

PRIOR KNOWLEDGE $S_M(\mathbf{p})$:skin-bone model

OUTPUT $\mathbf{p}_t$:posture at time t

step.1 $\mathbf{p} \leftarrow \mathbf{p}_{t-1}$
// initialize $\mathbf{p}$ by $\mathbf{p}_{t-1}$

step.2 $\mathbf{p} \leftarrow \arg\min_{\mathbf{p}} E(\mathbf{p}; S_M(\mathbf{p}), S_T[t])$
// update $\mathbf{p}$ by minimizing the error E between model mesh $S_M(\mathbf{p})$ and reconstructed 3D shape $S_T[t]$ based on the Levenburg-Marquardt method.

step.A $i \leftarrow 1$

step.B calculate new value of $\mathbf{p}'$ based on error E.

step.C when any of the following conditions are satisfied, set $\mathbf{p} \leftarrow \mathbf{p}'$ and go to step.3, otherwise, set $\mathbf{p} \leftarrow \mathbf{p}'$, $i \leftarrow i+1$, and go back to step.A.

- $|\mathbf{p}' - \mathbf{p}| < \epsilon_p$ // increment of posture parameter falls below a threshold ϵ_p.
- $E(\mathbf{p}) - E(\mathbf{p}') < \epsilon_e$ // diminution of E falls below a threshold ϵ_e.
- $i = i_{max}$ // the number of iteration exceeds a threshold i_{max}.

step.3 $\mathbf{p}_t \leftarrow \mathbf{p}$
// take the updated parameter as the resulting estimate of posture at time t.

3.2. Mesh Correspondence

Using the framework of the ICP we represent each pair of corresponding vertices by $k = (v_m^k, v_t^k)$ and then define the correspondence of S_M and S_T by the set K as:

$$k = (v_m^k, v_t^k) \in K \quad \text{where } v_m^k \in V_M, v_t^k \in V_T \tag{1}$$

Note that V_M and V_T denotes set of vertices in model mesh and observed mesh respectively. We define the correspondences based on their Euclidean distance and represent them in a binary graph as shown as left one in Figure 2. As

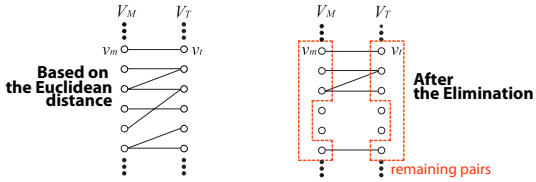

Figure 2: Correspondences between meshes.

we explained earlier, erroneous pairs of correspondence can occur where either vertex, i.e. either in the reconstructed 3D shape or in 3D human model, has a low observability value, being in the less-visible surface or in the solitary surface. We thus remove them based on the visibility measure (see right graph in Figure 2). Such a correspondence can be defined in two different ways as:

$$K_M(\mathbf{p}) = \{(v_m, v_t^m) | v_m \in V_M, v_t^m = \arg\min_{v_t \in V_T} d(v_m, v_t)\} \tag{2}$$

$$K_T(\mathbf{p}) = \{(v_m^t, v_t) | v_t \in V_T, v_m^t = \arg\min_{v_m \in V_M} d(v_m, v_t)\} \tag{3}$$

We use the sum of K_M and K_T as the entire set of correspondences.

$$K(\mathbf{p}) = \{K_M(\mathbf{p}) \cup K_T(\mathbf{p})\} \tag{4}$$

3.3. Visibility Measure as an Error Function

We find the posture, $\mathbf{p}$, which consists of rotation angles of the joints, by an optimization based on the Levenburg-Marquardt method. We first define a hypothetical correspondence set, $K(\mathbf{p})$, as explained before, and then revise $\mathbf{p}$ to minimize the error which we computed by function (5) where E_M and E_T are cost functions for each set of correspondences, K_M and K_T. We carry out this process iteratively. In each iteration we evaluate the visibility of vertices by an error function.

$$E(\mathbf{p}; S_M(\mathbf{p}), S_T) = \sum_{k_m(\mathbf{p}) \in K_M(\mathbf{p})} E_M(k_m(\mathbf{p})) + \sum_{k_t(\mathbf{p}) \in K_T(\mathbf{p})} E_T(k_t(\mathbf{p})) \tag{5}$$

In each frame we use the posture parameter derived in the previous frame for an initial value of $\mathbf{p}$.

The factors of our function, E_M and E_T, are based on the visibility measure, ρ_m and ρ_t. ρ_m evaluates how a vertex v_m is likely to have a corresponding vertex on the surface of a reconstructed 3D shape. ρ_t evaluates how accu-

rately the area where vertex v_t exists is likely to be reconstructed. Both of them are measured in terms of the numbers of cameras that are visible from the vertex and normalized in the range $[0, 1]$.

Figure 3 illustrates the visibility measure, ρ_m and ρ_t, on a simple synthesized data which is an articulated body with two joints. Left image in Figure 3 shows the distribution of visibility on the observed mesh and right one shows that on the model mesh. Less-Visible surfaces are colored by red. We exclude the red colored area from both observed and model mesh in the fitting. By using the visibility measures, our error functions, E_M and E_T, are formulated as equations (6) and (7) where $k(\mathbf{p}) = (v_m^k(\mathbf{p}), v_t^k)$ denotes a correspondence pair in posture $\mathbf{p}$.

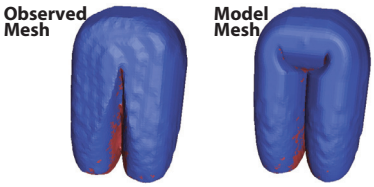

Figure 3: Distribution of visibility on a surface.

$$E_M(k(\mathbf{p})) = \rho_m^k(\mathbf{p})\{v_m^k(\mathbf{p}) - v_t^k\}^2 \frac{N_M}{\sum_{v_m}\rho_m(\mathbf{p})} + \zeta_m^k \left|\mathrm{sd}(v_m^k)\right| \quad (6)$$

$$E_T(k(\mathbf{p})) = \rho_t^k(\mathbf{p})\{v_m^k(\mathbf{p}) - v_t^k\}^2 \frac{N_T}{\sum_{v_t}\rho_t(\mathbf{p})} \quad (7)$$

The first term in (6) is a squared distance between corresponding vertices weighted by $\rho_m^k(\mathbf{p})$ where N_M is a number of vertices on model mesh, and $\sum_{v_m}\rho_m(\mathbf{p})$ is the sum of the visibility measures. The second term in (6) represents the cost for evaluating self collision of model mesh at vertex v_m^k where

$$\mathrm{sd}(v_m) = \begin{cases} |\mathbf{v}_m^c - \mathbf{v}_m| & \text{if } (\mathbf{v}_m^c - \mathbf{v}_m)\cdot\mathbf{n}_m^c \le 0 \\ -|\mathbf{v}_m^c - \mathbf{v}_m| & \text{otherwise.} \end{cases} \quad (8)$$

$$\zeta_m = \mathrm{sigmoid}\left(-\alpha\left\{\mathrm{sd}\left(v_m\right) + \tau_\zeta\right\}\right) \quad (9)$$

4. Experiments

We demonstrate the advantage of our approach in comparison to naive ICP algorithms using synthesized and real 3D video sequence. In this experiment, we use 15 XGA cameras surrounding a target in motion to acquire multiple view video for reconstruction of 3D shape of the target object. All cameras are calibrated in advance. The reconstructed 3D shape consists of about 14,500 vertices and 29,000 triangles. The processing time per frame is about 2 minutes by a PC (Xeon 3.0GHz).

4.1. Evaluation Using Synthesized Data

We have synthesized a simple shape model and given it a motion manually for the three cases, simple motion, semi-complex motion, and complex motion (see Figure 4). We virtually captured this data by multiple cameras, and reconstructed the shape by employing a visual-hull based method[7] for recovering the surface. We use the computed sequence of 3D shape as an input to our algorithm (and naive ICP algorithm for comparison), and evaluate the accuracy by comparing the resulting motion estimation with the ground truth which we give manually.

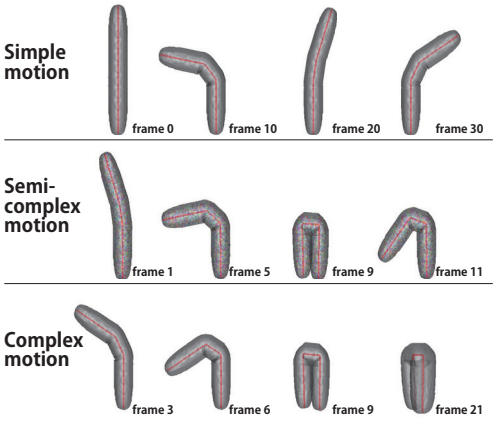

Figure 4: Input sequence and the estimation results.

Evaluation Method We evaluated the following three values that represent the error between estimated joint positions and the ground truth,

$$e_{mean} = \frac{\sum_{j=1}^{N_{joint}} |\mathbf{q}_j - \mathbf{p}_j|}{N_{Joint}} \quad (10)$$

$$e_{min} = \arg\min_j |\mathbf{q}_j - \mathbf{p}_j| \quad (11)$$

$$e_{max} = \arg\max_j |\mathbf{q}_j - \mathbf{p}_j| \quad (12)$$

where $\mathbf{p}_j$ and $\mathbf{q}_j$ are estimated position and the ground truth of joint j, respectively, where N_{joint} is the number of joints.

Results The results of experiments for the above three cases are in Figure 5. In each figure, the lines denote error function e_{mean} of the proposed method, naive ICP algorithm and touching degree which implies complexity of the target's posture. The top end and lower end of lines for error functions in each frame denotes e_{min} and e_{max}, respectively. Touching degree is the fraction of vertices on invisible surface and entire vertices in the 3D shape. e_{mean} of the proposed method remains much lower within a small range of deviation than that of naive ICP algorithm for all

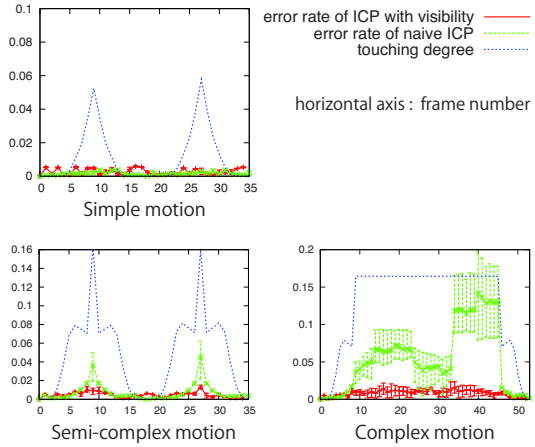

Figure 5: Evaluations.

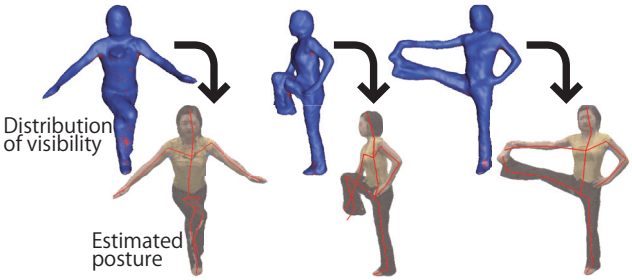

Figure 6: Estimation for Yoga sequence.

these cases with different complexity of motion. This shows that the target's motion is estimated more accurately and stably by the proposed method during complex motion in which the posture is complex with high touching degree while the conventional ICP algorithm failed due to the cumulative matching failures.

4.2. Experiments Using Reconstructed Data

We also made experiments with real data. We use sequences of reconstructed 3D shape of a target that is practicing yoga. We show the results of estimation in Figure 6. Red colored skeleton denotes estimated posture for each frame. We confirmed that our method can work on long sequences that consist of many types of postures including complex ones that involve an arbitrary number of collisions between body parts.

5. Conclusion

In this paper, we have proposed a new method for estimating complex human motion including touchings of body parts from reconstructed 3D shape. To cope with these touchings and noise in the process of reconstruction process, we introduced the visibility measure for model fitting process for posture estimation. We examined the proposed method with experiments, using synthesized and real data, and found that our approach outperforms the conventional method with naive ICP algorithm. To examine the advantage of our algorithm more, our future work includes comparisons against other improved ICPs and quantitative evaluations on real data sets.

Acknowledgements

This research was supported by "Development of High Fidelity Digitization Software for Large-Scale and Intangible Cultural Assets" project and GCOE program: "Informatics Education and Research Center for Knowledge-Circulating Society" (MEXT of Japan), and "Foundation of Technology Supporting the Creation of Digital Media Contents" project (CREST, JST). We would like to thank Dr. Lyndon Hill for his valuable writing corrections.

References

[1] P. Besl and N. McKay. A method for registration of 3-D shapes. *Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 3

[2] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992. 3

[3] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003. 3

[4] J. Feldmar and N. Ayache. Rigid, affine and locally affine registration of free-form surfaces. *International Journal of Computer Vision*, 18(2):99–119, 1996. 3

[5] D. Gavrila and L. Davis. Tracking humans in action: A 3d model-based approach. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 730–80, 1996. 2

[6] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006. 1

[7] S. Nobuhara and T. Matsuyama. Heterogeneous deformation model for 3d shape and motion recovery from multi-viewpoint images. *3D Data Processing, Visualization and Transmission*, pages 566–573, 2004. 5

[8] K. Ogawara, X. Li, and K. Ikeuchi. Marker-less human motion estimation using articulated deformable model. *Proc. IEEE Int. Conf. on Robotics and Automation*, 2007. 3

[9] J. M. Rehg and T. Kanade. Model-based tracking of selfoccluding articulated objects. *IEEE Int Conf. on Computer Vision*, pages 612–617, 1995. 2

[10] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994. 3