Tracking End-Effectors for Marker-Less 3D Human Motion Estimation in Multi-view Image Sequences*

Wenzhong Wang[1], Zhaoqi Wang[2], Xiaoming Deng[3], and Bin Luo[1]

[1] Computer Science Department, Anhui University, China
[2] Institute of Computing Technology, Chinese Academy of Sciences
[3] Institute of Software, Chinese Academy of Sciences
{wenzhong,luobin}@ahu.edu.cn, zqwang@ict.ac.cn,
xiaoming@iscas.ac.cn

Abstract. We propose to track the end-effectors of human body, and use them as kinematic constraints for reliable marker-less 3D human motion tracking. In the presented approach, we track the end-effectors using particle filtering. The tracked results are then combined with image features for 3D full pose tracking. Experimental results verified that the inclusion of end-effectors' constraints improves the tracking performances.

Keywords: end-effectors, motion tracking, particle filtering.

1 Introduction

Estimation of 3D human motion from multi-view image sequences has been a very active research topic in the late decades [1]. The most successful approaches are those based on marker tracking, such as the VICON system. These marker-based methods have proven to be very accurate in estimating 3D body poses. However, these methods face several practical difficulties and inconveniences: markers attached to the subjects hinder their motion; erroneous marker reconstructions need to be manually fixed; et al. Highly accurate motion can hardly be obtained without skilled operators and intensive labors. The rather cumbersome processes of marker-based motion capture systems limit their applications. These drawbacks have led to the vast research on marker-less motion capture. Marker-less motion capture aims to estimate 3D human motion directly from image sequences with no attachments on the subjects. It proves to be very hard due to the image noises, motion variations, and high degrees of freedom (D.O.F.) in human motions.

Most of the published research on this problem can be categorized into two groups, namely, the bottom-up approaches and the top-down approaches [1].

The former ones estimate human motion directly from image features. These methods either assemble the local image features (such as joint locations, limb edges) to 3D poses or learn a map from image features to 3D poses. In the first case, it is hard to detect discriminating and unambiguous local features from the noisy images, and

* This work is partially supported by the NSFC (project No. 61005039).

M. Kurosu (Ed.): Human-Computer Interaction, Part V, HCII 2013, LNCS 8008, pp. 227–235, 2013.

the estimated poses are rather coarse. In the later case, the mappings from image features to 3D poses are multi-valued and it is unlikely to learn such mappings without plentiful training data. The reliance on training data makes these approaches only applicable on specific motions.

The generative methods, on the other hand, do not presume any pre-captured training data. These approaches utilize a 3D human body model, and minimize an error function which measures how well this model fits the images. Many different error functions have been devised; most of them are based on the residuals between the 2D projections of 3D body model and the observed image features (such as silhouettes and edges). Some other error functions represent the alignment error of the body model to the voxel reconstructions of multi-view body silhouettes. The optimal 3D pose is found by minimizing these functions. Due to image noise and ambiguities, these functions are very peaky, rendering a difficult optimization problem. Still further, the high D.O.Fs of human pose makes the optimization even harder. There are mainly two categories of methods to this optimization problem, the one based on local optimization and the other one based on stochastic optimization. Local optimization methods start from an initial guess and iteratively find a descent direction of the error function. These methods guarantee convergence to local minimums. Their performance relies heavily on the initial values. They can hardly recover from errors during tracking. The stochastic methods approximate the posterior of poses by a set of weighted samples and can thus represent multi-mode distributions. This representation power makes the recovery from tracking error probable. The downside of these methods is that the effective number of samples increases exponentially with the D.O.Fs. Another difficulty with these methods is that the high dimensional prior distribution of poses cannot be effectively modeled, and this may results in many wrong samples which deteriorate the optimization performance.

In this paper, we try to explore high level information to facilitate top-down motion tracking. Specifically, we propose to track human limbs (a.k.a. end-effectors) firstly. This limb information is then used in a stochastic optimization process yielding optimal poses.

The motivation of our approach is based on these facts: the end effectors can significantly narrow down the solution space by the rule of inverse kinematics (IK); and this will compensate the weakness of the widely used low-level image features for pose optimization.

2 Related Works

We briefly discuss some works on human motion tracking using end-effectors' constraints.

Ganapathi [2] et al. detects the head, hands and feet in depth image, and generates poses from these detections using inverse kinematics.

Pons-Moll [3] et al. attaches inertial sensors on the arms and tibias, and use orientation cues derived from these devices to sample particles from the manifold of valid poses.

In [4], Hauberg et al. propose to track human motion in the end-effectors space. In their approach, the full 3D poses are modeled as normal distributed around the mean

obtained from the end-effectors' goals using inverse kinematics. Samples are then drawn from this distribution and evaluated using stereo image data. The mean of these weighted samples is used as the pose estimation.

Andreas Baak [5] et al. estimates five feature points from depth images; these feature points are assumed to be the head, hands and feet. These points are then used to retrieve similar 3D poses from a pre-recorded motion database.

These works use depth image data or other devices to facilitate the human pose estimation. Our work differs from the above ones in that we do not rely on any supplementary devices to obtain the end-effectors positions. We'd rather estimate this information directly from multi-view image data.

3 Proposed Method

Our method consists of two interleaved processes: one for end-effectors tracking and the other for 3D pose tracking. Both of these processes are implemented using particle filters.

Let g_t ,x_t be the end-effectors goals and pose at time instance t, respectively. We denote the observed image at time t as z_t. We formulate the pose tracking problem in a Bayesian filtering framework [6] as depicted in Fig1.

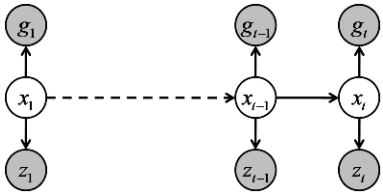

Fig. 1. Probabilistic graphical model for our pose estimation problem

In Bayesian filtering, we are interested in the posteriori of x_t, $P(x_t|G_t, Z_t)$:

$$P(x_t|G_t, Z_t) = k_t P(g_t, z_t|x_t) \int P(x_t|x_{t-1}) P(x_{t-1}|G_{t-1}, Z_{t-1}) d\, x_{t-1} \quad (1)$$

where we have defined $G_t = \{g_1, \cdots g_t\}$, $Z_t = \{z_1, \cdots z_t\}$ and k_t is a normalization constant independent of x_t.

3.1 Particle-Based Estimation of Human Pose

The posteriori in (1) is approximated as a set of weighted particles and estimated using particle filtering.

Since the end-effectors' goal g_t and image z_t are conditionally independent given the pose x_t, (1) can be written as:

$$P(x_t|G_t, Z_t) = k_t P(g_t|x_t) P(z_t|x_t) \int P(x_t|x_{t-1}) P(x_{t-1}|G_{t-1}, Z_{t-1}) d\, x_{t-1} \quad (2)$$

We now define the image observation model $P(z_t|x_t)$, forward kinematics model $P(g_t|x_t)$ and motion dynamical process $P(x_t|x_{t-1})$.

The Image Observation Model $P(z|x)$. We use a simple human body model shown in Fig2. The whole body is decomposed into 10 rigid parts, and each part is modeled as a circular truncated cone. This is a very coarse simplification of real human body. The benefit of such a model is that it is very easy to calculate its projection and check its limb occlusions.

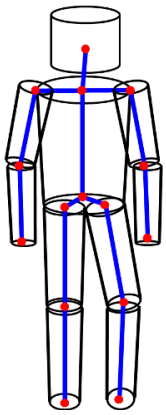

Fig. 2. 3d human body model

Having specified the 3D body model, we now define the image observation model as:

$$P(z|x) \propto \exp\left(-\frac{\epsilon^2(x)}{\sigma_z^2}\right) \tag{3}$$

where $\epsilon(x)$ is the discrepancy between the projection of body model and the observed image z.

We use edges as image features and define this error as:

$$\epsilon(x) = D^{M(x)} \cdot E^I + D^I \cdot E^{M(x)} \tag{4}$$

where $E^{M(x)}$ and $D^{M(x)}$are , respectively, edge map and chamfer distance map of the projected image of of body model in pose x, and E^I is the edge map of observed image z and D^I its distance map. $\cdot$ denotes pixel-wise product operation.

The Forward Kinematics Model $P(g|x)$. The end-effectors' positions are deterministic functions of full pose x: $g = f(x)$. However, in order to account for the errors in our estimations of g, we model the kinematics constraint in a probabilistic way:

$$P(g|x) = N\big(f(x), \Sigma_g\big) \tag{5}$$

The Dynamical Process $P(x_t|x_{t-1})$. Each body pose is composed of 36 degrees of freedom: $x = (r, o, \theta)$, where $r \in R^3$ is the root position, $o = \big(o_x, o_y, o_z\big)$is the root orientation, and $\theta = (\theta_1, \cdots \theta_{30})$ is 30 joint angles.

Since r, o, θ are independent of each other. We factorize $P(x_t|x_{t-1})$ into three parts:

$$P(x_t|x_{t-1}) = P(x_t|x_{t-1})P(o_t|o_{t-1})P(\theta_t|\theta_{t-1}) \tag{6}$$

The parameters are learned from a prerecorded human motion database.

In our implementation, we firstly retarget the training joint angles $\theta_j$to the body model of the subject to be tracked, and then extracted all the joint positions $y_j \in R^D$. Other than learn a probabilistic model in the angular space (which is not a vector space), we learn the dynamical process in the joint position space.

The original high dimensional position states y_j's are projected into a lower dimensional space using PCA:

$$y_j = \bar{y} + W\tilde{y}_j$$

where $\bar{y}$ is the mean state of training data $\{y_j\}_{j=1}^n$, $W \in R^{D\times K}$is the projection matrix consisting of the first K principle components. In our experiments, $K = 6$. So the high dimensional state θ_j is replaced with a low-dim vector $\tilde{y}_j \in R^6$.

We will track the poses in the joint space of r, o and $\tilde{y}$. The dynamical process in (6) now becomes

$$P(\boldsymbol{x_t}|\boldsymbol{x_{t-1}}) = P(\boldsymbol{r_t}|\boldsymbol{r_{t-1}})P(\boldsymbol{o_t}|\boldsymbol{o_{t-1}})P(\tilde{\boldsymbol{y}}_\boldsymbol{t}|\tilde{\boldsymbol{y}}_\boldsymbol{t-1}) \tag{7}$$

The three components in (7) are modeled as first order Gaussian auto-regression processes:

$$\begin{cases} P(\boldsymbol{r_t}|\boldsymbol{r_{t-1}}) = N(\boldsymbol{r_{t-1}}, \boldsymbol{\Sigma_r}) \\ P(\boldsymbol{o_t}|\boldsymbol{o_{t-1}}) = N(\boldsymbol{o_{t-1}}, \boldsymbol{\Sigma_o}) \\ P(\tilde{y}_t|\tilde{y}_{t-1}) = N(\tilde{y}_{t-1}, \boldsymbol{\Sigma_y}) \end{cases} \tag{8}$$

The covariance matrices $\boldsymbol{\Sigma_r}, \boldsymbol{\Sigma_o}, \boldsymbol{\Sigma_y}$ are estimated from training data.

3.2 Tracking the States of End-Effectors

The above statements presume the end-effectors' positions are known. In this subsection, we present an approach for tracking these positions.

The end-effectors we used are the four lower limbs (two tibias and two forearms) of the human body. The state of each end-effector is described by its position and orientation in the world coordinate system.

The state of the four end-effectors is given by $g = \{p_i, \alpha_i, \beta_i\}_{i=1}^4$. Where $p_i \in R^3$ is the position of the proximal end of limb i, and (α_i, β_i) is the pointing direction of limb i (in a cylindrical parameterization).

The state g is estimated using particle filtering. We now define the dynamical process of g and the observation process of image z given g as follows:

The dynamical process of g, $P(g_t|g_{t-1})$, is factorized into three processes, $P(g_t|g_{t-1}) = P(p_t|p_{t-1})P(\alpha_t|\alpha_{t-1})P(\beta_t|\beta_{t-1})$ and defined as below:

$$\begin{cases} P(p_t|p_{t-1}) = N(p_{t-1}, \Sigma_p) \\ P(\alpha_t|\alpha_{t-1}) = N(\alpha_{t-1}, \sigma_\alpha) \\ P(\beta_t|\beta_{t-1}) = N(\beta_{t-1}, \sigma_g) \end{cases} \tag{9}$$

The observation process is defined as:

$$P(z|g) \propto exp\left(-\alpha_a d_a(g,z) - (1-\alpha_a) d_c(g,z)\right) \tag{10}$$

where $d_a(g,z)$ and $d_c(g,z)$ are two energy terms representing the mismatch of the limb projections with the image z.

The appearances of lower limbs are represented using eigen-templates [7]. We manually labeled about 30 image patches that corresponds to each limb, and project the scale normalized patches onto a low dimensional space using PCA[7], yielding eigen-template representations of each limb. Since the left and right limbs are very similar in appearance, they share the same eigen-templates model. Fig3 shows the learned eigen-templates of forearm. The first panel shows the mean patch and the rests are the first 15 principle components.

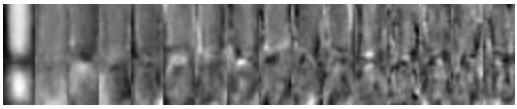

Fig. 3. Eigen-templates of forearm

The appearance energy $d_a(g,z)$ is calculated in the low dimensional space. The image area covered by the limb projection is extracted, rotated and rescaled, and then is projected onto the low dimensional space spanned by the eigen-templates. $d_a(g,z)$ is then calculated as done in [7].

The edge energy $d_c(g,z)$ is calculated as the edge mismatch between image edges and projected limb contours, defined as equation (4).

Occlusion Handling. One must be careful when evaluating the particles because the limbs may be occluded by the body. So we introduce the visibility parameter for each limb in each camera view.

In order to predict the limb visibility, we project the estimated pose in previous frame onto each view, and calculate the visible area of each limb. We then project a single limb onto each view, and calculate the area it covers. Wu use the ratio between these two areas as the visibility measurement.

The visibility of limb i in camera view k, $v_{i,k}$, is defined as

$$v_{i,k} = \frac{A_{i,k}}{A_{i,0}}$$

where $A_{i,k}$ is the visible projection of limb i in camera view k in previous pose, and $A_{i,0}$ is the area covered by limb i in camera view c. Fig4 illustrates the projections of left forearm, the white area in the left and right images are $A_{i,k}$ and $A_{i,0}$, respectively. Note that in the left image, the left forearm is partially occluded by the rest of the body (red area).

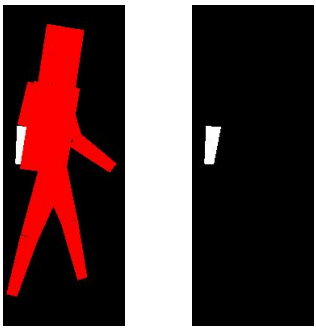

Fig. 4. Calculating limb visibility

Having determined the visibility of each limb, we can accumulate the energy terms in different camera views:

$$\begin{cases} d_a(g_i, z) = \sum_{k=1}^{C} v_{i,k} d_a\left(g_i^k, z^k\right) \\ d_c(g_i, z) = \sum_{k=1}^{C} v_{i,k} d_c\left(g_i^k, z^k\right) \end{cases} \tag{11}$$

where g_i^k is the projection of g_i onto image k, and z^k is the image in camera view k.

4 Experiments and Discussions

We've tested the above method on the HumanEvaI [8] dataset. In order to illustrate the improvement made by the end-effectors' constraints, we also run the tests without using these constraints (i.e. traditional particle filtering). All other parameters are the same for these tests.

We perform experiments using walking videos of subject S1 in cameras C1, C2, and C3. The parameters of dynamical processes (eqn. 8, eqn. 9) are learned from the other 3D motion data of S1. We've tried to estimate these parameters from motion data of other subjects, but found that the tracking results significantly deteriorated. This indicates that the motion model is crucial for robust tracking.

The per-joint position errors are shown in Fig5. We've achieved better results than state-of-the-art method (particle filtering). We attribute the performance improvement to the adoption of the end-effectors' constraints. The end-effectors' information can significantly reduce the ambiguities in images, and bias the search of optimal pose towards those satisfying end-effectors' constraints.

The tracking results for end-effectors are shown in Fig6. We've found that tracking performance highly relies on the motion model (eqn.9) and the limbs' visibility. The end-effectors can hardly be tracked reliably with less than three cameras. In our experiments, each limb can be clearly observed from at least two cameras. Due to the ambiguities in between the left and right limbs, the tracker may be confused by the similar image data. This would happen when the two tibias cross over each other.

We show motion tracking results in Fig7. These results are visually appealing. It proves very difficult for reliable tracking when there are severe occlusions. In these cases, the image data are ambiguous and the tracker will fail without precise motion models.

In conclusion, we've verified that end-effectors constraints could be helpful for reliable tracking.

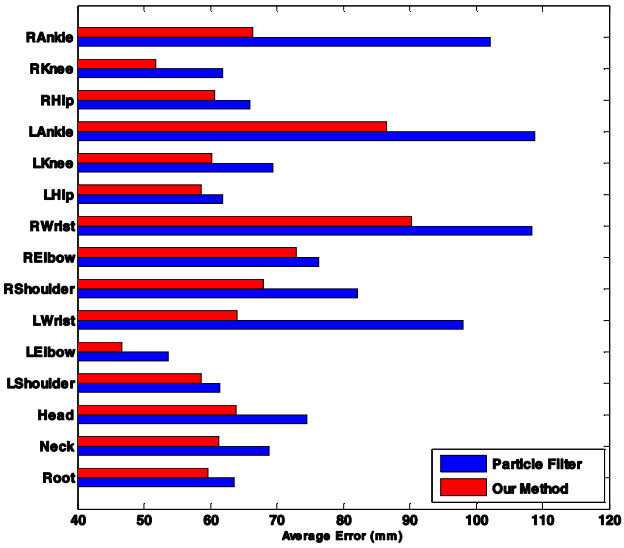

Fig. 5. Averaged joint error

Fig. 6. Tracked end-effectors in C1,C2 and C3

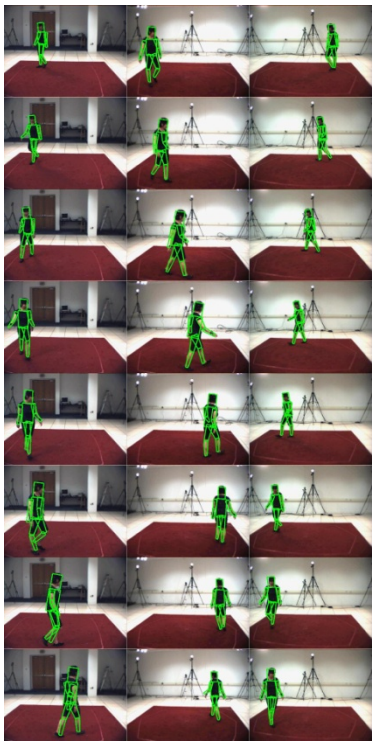

Fig. 7. Human pose tracking results in C1,C2 and C3

References

1. Moeslund, T.B., et al. (eds.): Visual analysis of humans: looking at people. Springer (2011)
2. Ganapathi, V., et al.: Real time motion capture using a single time-of-flight camera. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2010)
3. Pons-Moll, G., et al.: Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In: IEEE International Conference on Computer Vision, ICCV. IEEE (2011)
4. Hauberg, S., Pedersen, K.S.: Predicting articulated human motion from spatial processes. International Journal of Computer Vision 94(3), 317–334 (2011)
5. Baak, A., et al.: A data-driven approach for real-time full body pose reconstruction from a depth camera. In: IEEE International Conference on Computer Vision (ICCV). IEEE (2011)
6. Isard, M., Blake, A.: Condensation—conditional density propagation for visual tracking. International Journal of Computer Vision 29(1), 5–28 (1998)
7. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(7), 696–710 (1997)
8. Sigal, L., Balan, A.O., Black, M.J.: Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. International Journal of Computer Vision 87(1), 4–27 (2010)