Integrating Component Cues for Human Pose Tracking

Mun Wai Lee and Ramakant Nevatia

Institute for Robotics and Intelligent Systems
University of Southern California
Los Angeles, CA 90089-0273, USA
{munlee, nevatia}@usc.edu

Abstract

Tracking human body pose in monocular video in the presence of image noise, imperfect foreground extraction and partial occlusion of the human body is important for many video analysis applications. Human pose tracking can be made more robust by integrating the detection of components such as face and limbs. We proposed an approach based on data-driven Markov chain Monte Carlo (DD-MCMC) where component detection results are used to generate state proposals for pose estimation and initialization. Experimental results on a realistic indoor video sequence show that the method is able to track a person during turning and sitting movements.

1 Introduction

Human body pose tracking is important for surveillance and video analysis, especially for understanding human activities and events. Our aim is to recover body and limb positions and orientations, *i.e.* their poses, in 3-D from monocular video sequences without any special markers placed on the human body or clothing. This requires segmentation of human body from the surround and other moving objects and inference of 3-D from 2-D observations where not all the limbs and body joints may be visible in each frame and image feature estimates are likely to contain noise and outliers.

We model human body as an articulated object with 14 joints and 31 degrees of motion freedom. We use a generative approach to estimate these parameters; thus efficient search is a key issue. In a sequence, we can use the previous state and a dynamic model to reduce the search space for the current state. However, we must estimate the initial state and re-initialize when tracking becomes unreliable. Therefore a robust pose estimation scheme has to be augmented with data-driven mechanism to detect body parts for initialization and this is the main focus of our work.

Figure 1 shows the overall configuration of our system. First, we extract motion foreground (by background subtraction assuming a stationary camera), detect image edges and perform color segmentation using the mean shift technique. These features are used to make estimates for body components such as the face, head-shoulder contours and limbs; probabilities of their detection are stored in *proposal maps* which are used to guide the search for body pose estimation. The search is conducted by using the data-driven Markov chain Monte Carlo (DD-MCMC) approach [16]. Given a state candidate, we evaluate it by generating a synthesized human (we assume an orthographic approximation) and comparing it to the input image. This generative approach allows us to consider highly nonlinear constraints such as non-self-penetration.

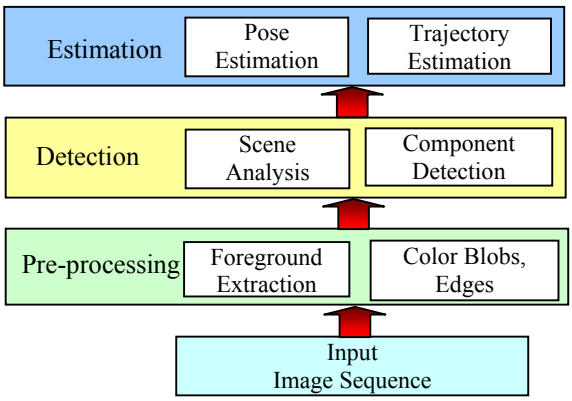


Figure 1. Overview of Approach

For tracking dynamic pose, multiple Markov chains are used, one for each frame, and the estimation in adjacent frames influence the Markov transition based on a human dynamic model. Dynamic programming is used to extract the optimal trajectory of the human movement.

We have used a similar approach previously for human pose estimation in single images and video [1] [2] for people who are mainly in upright and frontal poses. This work extends the method to dynamic pose estimation in much more difficult scenarios including turning and sitting movements as well as some interactions between the person and a moving scene object (a door). This requires substantial new capabilities for detection of body components, estimation of scene object motion and composite *proposal maps*.

1.1 Related Work

There has been substantial work on estimating the 2D human pose [9][18][10] and in the use of range data from stereo [4]. Estimating 3D pose is more challenging as

some degrees of motion freedom are not observed and it is difficult to find a mapping from observations to state parameters. Therefore, a generative approach is commonly used because it is easier to evaluate a state candidate by synthesizing the human appearance. Monte Carlo techniques are often used to generate plausible candidates for evaluation.

In [5], *particle filtering* is used for 3D pose tracking with multiple cameras by approximating the state posterior distribution with a set of samples. It is however difficult to extend this to tracking monocular view because of the significant ambiguity in depth. In [12], a *mixture density propagation* approach is used to overcome the depth ambiguities of articulated joints seen in monocular view. This is later improved with a *mixture smoother* [13]. Nonetheless, the issue of pose initialization is not addressed.

Recently, a *non-parametric belief propagation* (NBP) method has been used for pose estimation, in which the inference on each body component is propagated along a belief network [11].

To deal with the high dimensional state space and pose ambiguities, some previous efforts assume a low-dimensional dynamic space, (e.g. walking human) [3]. This scheme often results in good tracking in the studied movement, but less applicable for uncontrolled human motion. Recently, the use of bottom-up local parts detection is becoming a popular approach [8][10] to bootstrap the pose estimation. In addition, global shape-based features are used in [17] mainly for people walking in a frontal-parallel manner.

2 Pose Estimation Framework

In this section, we describe the key components of our estimation framework including the human model, the observation, and the formulation of the prior distribution and likelihood function.

2.1 Human Body Model

We use an articulated limb model of human body that defines the pose parameters as consisting the torso position and orientation, and various body joint angles. Additional latent parameters that describe the shape of the torso and limbs and the clothing type are also included to synthesize the human appearance more suitably for pose evaluation, as in [1].

2.2 State Estimation

Pose estimation is formulated as the problem of estimating the state of a system. State for a sequence with T frames is represented by $\{X_1, X_2, ..., X_T\}$, where X_t represents the state at the t^{th} frame. The state includes the pose parameters, the shape parameters and the clothing parameters.

The observed shape of a moving person tends to change due to clothing and posture. Therefore, the shape parameters are dynamic to allow deformation so that the synthesized human could align with the input more accurately. The observed images, denoted as $\{I_1, I_2, ..., I_T\}$, and are assumed to be conditionally dependent on the current states only.

2.3 Prior Distribution

The prior distribution of the state, denoted by $p(X_1, X_2, ..., X_T)$, can be decomposed into a series of potential functions corresponding to the pairs of adjacent states in the graphical model.

$$p(X_1, X_2, ..., X_T) = \frac{1}{z}\prod_{t=1}^{T-1}\psi(X_t, X_{t+1}),$$

where Z is a normalization constant. Each potential function can be further decomposed into a prior distribution of a state, and a conditional distribution:

$$\psi(X_t, X_{t+1}) = p(X_t)p(X_{t+1} \mid X_t).$$

The prior distribution is learned from a training set of human poses in static image and sets of motion capture data. The conditional distribution is based on a zero$^{\text{th}}$ - order dynamic model and is approximated by a normal distribution.

$$p(X_{t+1} \mid X_t) \approx N(X_{t+1} - X_t, \Sigma), \quad (1)$$

where Σ is the covariance matrix of the dynamic model and is learned from motion capture data.

2.4 Likelihood Function

A state candidate X_t is evaluated by a likelihood function denoted by $p(I_t \mid X_t)$. We formulate the image likelihood function as consisting of four components, based on (*i*) region coherency, (*ii*) color dissimilarity with background, (*iii*) skin color and (*iv*) foreground matching, respectively. These likelihood measures are described in detail in [1].

3 Component Detection and Scene Analysis

Component detection has become a popular approach to jump-start or bootstrap pose estimation [8][9][11] and is essential for applications that require automatic initialization. In our work, component detectors are used to define proposals for generating pose candidates during the Markov chain search. Some of these techniques have been described in our previous work [1] mainly for frontal upright poses. In this section, we describe additional techniques that enable the tracking of more complex poses. These include an integrated head detector and a body orientation estimator. We also describe a scene analyzing tool that estimates the state of a door in an indoor scene.

3.1 Integrated Head Detector

Our head detector uses multiple cues to generate weighted candidates of head position used for data-driven proposal.

Figure 2 illustrates the detection process. Candidates for head are first detected by one of two methods: (i) cascade classifier for face detection proposed by Viola and Jones [15], and (ii) detection of blobs with skin and hair colors. This set of candidates is then pruned using the extracted foreground. At the remaining candidates, we search for the contour of the head-and-shoulder using a probabilistic shape model and use the result to further reweigh the candidates. A candidate's weight, w_{Face}, is given by:

$$w_{Face} = K \times w_{Shape} \times w_{Foreground} \times w_{Detector},$$

where K is a constant. The shape-based weight is given by

$$w_{Shape} = \exp(-e^2_{shape} / \sigma^2_{shape}),$$

e_{shape} is the error of contour matching. The foreground-based weight is given by

$$w_{Foreground} = \exp(-R^2_{Bg} / \sigma^2_{Fg}),$$

where R_{Bg} is the ratio of face region pixels that are classified as background. The detector-based weight, $w_{Detector}$, depends on the type of detector that generates the candidate and is given by

$$w_{Detector} = \begin{cases} P_{TP,AdaBoost} & \text{for AdaBoost candidates} \\ P_{Skin} \times P_{TP,Skin} & \text{for skin - ellipse candidates} \end{cases},$$

where $P_{TP,AdaBoost}$ and $P_{TP,Skin}$ are the true positive rate of the two detectors, and P_{Skin} is the probability of skin-color of the face region, computed by a histogram-based skin-color model.

From a different set of training images, the observation model parameters σ^2_{shape}, σ^2_{Fg}, $P_{TP,AdaBoost}$ and $P_{TP,Skin}$ were learned. By using multiple cues, the detector has a high detection rate and an acceptable false alarm rate for subsequent processing.

3.2 Torso Detection

The torso, being the root node of the body kinematic tree, is important and good initial estimates of the torso will improve the efficiency of the Markov chain search.

The detected face is used to bootstrap the detection and tracking of the torso. We use a face-body tracker [1] that utilizes the specific human face-body structure and the consistent appearance of the torso to track the face and torso simultaneously using the meanshift technique (see Figure 4). This technique is used to provide evidence about the position and the slant of the torso in the image.

In addition, the pan angle of the torso can be estimated in two ways. First, the AdaBoost technique detects faces of 3 different views (frontal, full left/right profiles). This provides coarse but reliable estimates of torso pan angle. Second, the direction of image flow also provides evidence on the body orientation. This approach has been used effectively in [18] for tracking moving people.

3.3 Door Status Estimation

A video may contain moving components of the scene, such as a door. For surveillance applications, scene information about these objects can be obtained and included in generative modeling.

We describe modeling of a door in the scene of our test video. Training images provide information about the door position and color. The scene background appearance behind the door is also learned. When the door moves, it creates a foreground layer. A Bayesian classifier is used to classify these pixels as door or non-door using the learned color distributions. The classified door pixels are then used to determine the image width of the door (Figure 3).

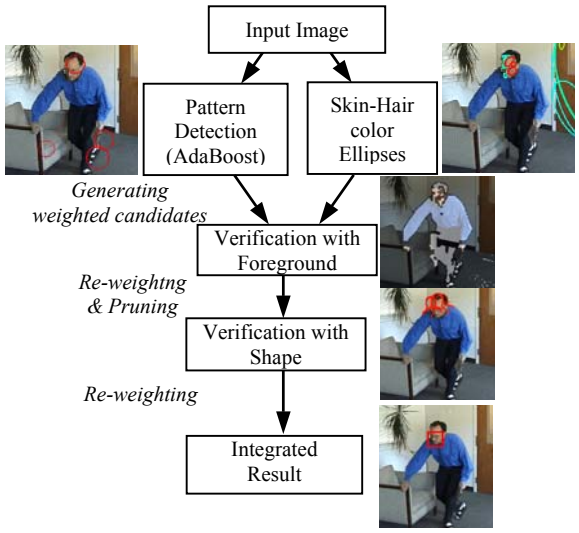


Figure 2. Head Detection.

Figure 3. Door Status Estimation. (a) and (b) show the 2 training images and manually defined masks of the door. Human regions were removed from these images prior to training. (c) and (d) show the test results of two frames: the classified door pixels are highlighted in green in (c), and the estimated door edges are shown in (d).

4 Proposal Mechanisms

Different proposal mechanisms are used for the Markov chain transitions. We follow the procedure described in our earlier paper [1] and provide only a brief summary here for completeness.

4.1 Proposal Function

The MCMC uses a proposal function to generate state candidates. In theory, one can generate a candidate for the whole sequence of states $\{X_1, X_2, ..., X_T\}$, (e.g. [7],) but such schemes have high computation complexity and difficult to implement. Instead, we focus on the proposal function for a state X_t at a time.

A state candidate, X_t^*, is generated by three types of evidence:

1. The estimation of previous state, X_{t-1}, can be propagated using a human dynamic model to generate candidates for the current state.

2. The candidates can be generated from the current observation, I_t. This is an adaptation of a bottom-up data-driven approach [16] that has now been used for a number of computer vision tasks [14][18]. We shall discuss this in more details in Section 4.2.

3. Using a backward-propagation approach, the next state estimation, X_{t+1}, can also be used to generate candidates for the current state. We note that the backward-propagating component, $q(X_t^* \mid X_{t+1})$, is optional, depending on application requirement and constraint.

The proposal distribution is denoted by $q(X_t^* \mid X_{t-1}, I_t, X_{t+1}, X_t')$, where X_t' is the current Markov chain state. For simplicity, the distribution can be decomposed into its components:

$$q(X_t^* \mid X_{t-1}, I_t, X_{t+1}, X_t') = \alpha_1 q(X_t^* \mid X_{t-1}) + \alpha_2 q(X_t^* \mid I_t) + \alpha_3 q(X_t^* \mid X_{t+1}) + \alpha_4 q(X_t^* \mid X_t')$$

where $\alpha_1, \alpha_2, \alpha_3$, and α_4 are the mixing ratios for the different components. The last component, $q(X_t^* \mid X_t')$, represents a proposal distribution derived from the current Markov state. It is implemented to involve both the *random-walk sampler* [6], and the *flip kinematic jump* [12] which is specifically designed to explore the depth space [1].

4.2 Proposals from Observation

We use observations in the current frame to generate proposals for the state, $q(X_t^* \mid I_t)$. The observations include face detection, head-shoulder contour matching, skin blob extraction, torso and foot detection [1].

(a) (b) (c) (d) (e)

Figure 4. Detection modules. (a) Face pattern, (b) head-shoulder contour, (c) Skin color blobs (for face and hand), (d) face-body tracker, and (e) foot candidates.

The evidence provided by these observations can be integrated and represented in *proposal maps*. There is one map for each body joint (see Figure 5). In each map, the value at each pixel position represents the importance sampling probability of the corresponding joint's image position. The maps are used to generate pose candidates in a component-based Metropolis-Hastings approach. This technique is motivated by the data-driven MCMC framework [16][18]. In [1], it is shown how this framework can be adapted for estimating 3D kinematics parameters by constructing reversible jumps using the proposal maps and inverse kinematics computation, as illustrated in Figure 6, using the left foot as example.

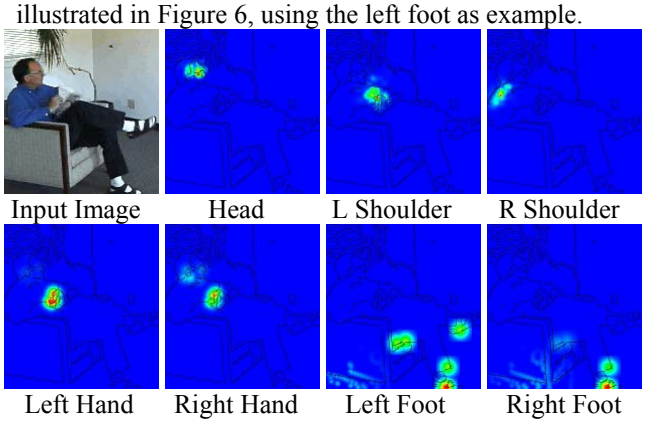
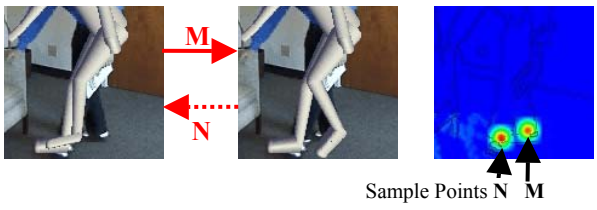


Figure 5. Proposal Maps (shown in pseudo-colors).



(a) (b) (c)

Figure 6. Data Driven Proposal. (a) Initial lower body pose, (b) updated pose, if point M is sampled from the proposal map of the left foot, as shown in (c). The reverse jump would occur if point N is sampled.

4.3 Dynamic Proposals

Dynamic proposal mechanism involves generating a state candidate for the current frame, X_t^*, either from the

estimates in the previous frame, X_{t-1}, or in the next frame, X_{t+1}. For discussion, we focus on the former.

The state estimation in the previous frame is represented by a set of state samples $\{X_{t-1}^1, X_{t-1}^2, ...\}$ generated by the Markov chain search. These samples are clustered to form a compact set of representative samples $\{X_{t-1}^{(1)}, X_{t-1}^{(2)}, ..., X_{t-1}^{(N)}\}$, where N is the number of mixture components [1][2]. These components are weighted according to their cluster sizes.

To generate a candidate for the current frame, a sample $X_{t-1}^{(*)}$ is selected from the set of mixture components in the previous frame $\{X_{t-1}^{(1)}, X_{t-1}^{(2)}, ..., X_{t-1}^{(N)}\}$ based on their normalized weights $\{w_{t-1}^{(1)}, w_{t-1}^{(2)}, ..., w_{t-1}^{(N)}\}$. Using a zeroth-order dynamic model, the state candidate is generated by sampling a normal distribution centered at $X_{t-1}^{(*)}$:

$$X_t^* \sim N(X_{t-1}^{(*)}, \Sigma),$$

where Σ is the covariance matrix of the dynamic model, as described in Equation (1) of Section 2.3. The dynamic proposal distribution is given by:

$$q(X_t^* \mid X_{t-1}) = \sum_1^N w_{t-1}^{(i)} N(X_{t-1}^{(i)}, \Sigma).$$

4.4 Extension to scene with two people and a moving scene object

In this section, we describe the extension to tracking of two people and the scene includes a door. In the video used, the people are well separated. We have not tested on videos where the people overlap each other.

Apportioned Proposal Maps

When there are two people in the scene, the proposal maps described earlier, encapsulate the evidence on *both* person. While these maps could still be used directly, it is inefficient as half of the data-driven candidates would be incorrect.

The proposal maps can be *apportioned* into two sets of maps, one for each person. This will reduce the search space for each person and improve efficiency when the two are separated, (Figure 7).

This approach requires an initial inference about the existence of two people and coarse estimates of their positions. This can be achieved by foreground blob analysis and we used a simplified version of a generative model-based approach described in [18]. This scheme is applicable even when the two people overlap partially in the image, but not completely.

We denote $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ as the estimates of the person's image positions, and $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ as the covariance matrices of measurement noise. For each pixel position, denoted as $\mathbf{x}$, we assigned the apportioned proposal maps, $m_1(\mathbf{x})$ and $m_2(\mathbf{x})$, as follows:

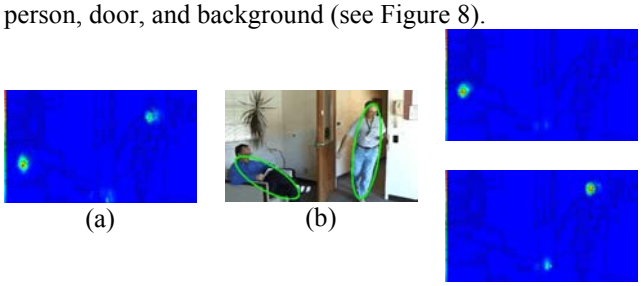
$$m_1(\mathbf{x}) = \frac{N(\mathbf{x} - \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)}{N(\mathbf{x} - \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) + N(\mathbf{x} - \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)} m_{all}(\mathbf{x}),$$

and

$$m_2(\mathbf{x}) = m_{all}(\mathbf{x}) - m_1(\mathbf{x}),$$

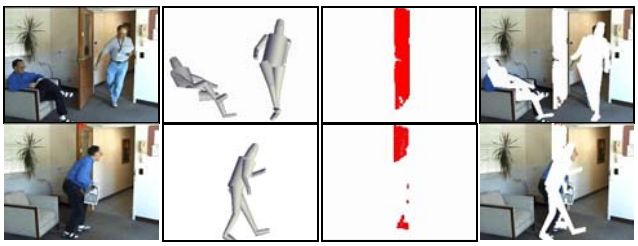
where $m_{all}(\mathbf{x})$ is the original map, and $N(\cdot,\cdot)$ is the 2D normal distribution. We then normalized $m_1(\mathbf{x})$ and $m_2(\mathbf{x})$.

MCMC is used to estimate the poses of the two persons jointly. For data-driven proposals, the state of each person is updated separately using the corresponding apportioned proposal maps. The evaluation is carried out jointly. The likelihood function described earlier can be easily extended to multiple people, it requires the assignment of pixels to one of four layers: 1st person, 2nd person, door, and background (see Figure 8).



(a) (b) (c)

Figure 7. Apportioned Proposal Maps. (a) Original Map for head $m_{all}(\mathbf{x})$, (b) estimates of positions, (c) generated maps, $m_1(\mathbf{x})$ and $m_2(\mathbf{x})$. This figure illustrates using the maps for head only.



(a) (b) (c) (d)

Figure 8. Inferring different layers of the scene. (a) input images, (b) synthesized person with estimated pose, (c) foreground pixels classified as door, and (d) remaining pixels assigned as background.

4.5 Extracting Pose Trajectory

The previous section describes the state estimation for each frame. The set of generated Markov samples can be represented compactly using a mixture model as described in Section 4.3. Using dynamic programming, the trajectory of the human body can be recovered by "traversing" along the sequence and selecting a set of poses from these mixture components as in [2]; a similar procedure can also be found in [13].

5 Experimental Results

In this section, we describe the experimental setup and the result.

5.1 Setup

We used one of the SRI video sequences for evaluating the proposed technique. We annotated the video manually to aid in evaluation by locating the image positions of the body joints. The depths of these joints, relative to the hip, were also estimated. The annotation data are used for evaluation only and not for training. Some prior data of the scene is provided to the system: two sets of image and mask of the door (shown in Figure 3(a)-(b)), and a mask indicating an occluding layer of an armchair.

5.2 Pose Tracking Results

The results of pose estimation are shown in Figure 10 and Figure 11. The tracking started after the human fully entered the scene. The initialization is automatic. The shape of the human model was initialized as the mean shape.

As the results show, the proposed method is able to initialize and track the human poses robustly. Estimation of the salient components including the face, torso and hands are fairly accurate. These help to boost the estimation of the other joints that are either less salient (*e.g.* elbows) or are temporarily occluded.

Some instances of failure are observed and specific examples are described in the captions of Figure 10 and Figure 11. Nonetheless, the quality of tracking is good considering that we are using monocular video and some joints are temporarily occluded.

5.3 Error Measures

For evaluation, we compare the estimated joint position with the annotated data using the tool described in [2]. In the t^{th} frame, we compute the 3D Euclidean distance error (in pixels) for the j^{th} joint, denoted by e_t^j. A weighted average error, denoted by E_t, is defined by:

$$E_t = \left[\sum_{j=1}^{M} w_j e_t^j \Big/ \sum_{j=1}^{M} w_j \right],$$

where M is the number of joints used for evaluation and $w_1 \ldots w_M$ are the weights. The weights are chosen to approximate the relative size of the corresponding body parts, and the values are: 1.0 for torso and neck; 0.6 for shoulders, elbows and knees; 0.4 for wrists and ankles; 0.3 for head; and 0.2 for hand-tips and foot-tips. For the first person, evaluation is performed for Frame 72-230, and Frame 330-425. For the second person, it is for Frame 344-364. The evaluation is based on all the above-mentioned joints even though some joints are temporarily occluded. For this evaluation, we have ignored Frame 231-329 during which there is little movement in the scene.

The error at each frame is plotted in Figure 9. (We call the testing of our proposed method **Trial A**). The average error for the sequence is given in Table 1. Table 2 shows a decomposition of the error into image and depth distances, and into upper and lower body. The result shows higher error for depth and for the lower body. As a reference, the person image height is about 200 pixels; a pixel approximately represents 0.9cm.

For comparison, we also conducted two other trials. In **Trial B**, the back-propagation proposal was not used, and in **Trial C**, neither the forward nor the backward propagation proposals were used. (In all the trials, dynamic programming was used to extract the optimal trajectories). This analysis provides an insight about the contributions by the different proposal mechanisms.

The results show that in some frames, single image analysis with DD-MCMC is capable of obtaining good pose estimation, and these are the frames where error measures are comparable in all the three trials. In other frames, especially when there are self-occlusions, the errors for Trial A would be lower for Trial B and C. This demonstrates that the method utilizes current observation as well as estimates in neighboring frames to achieve more robust pose tracking.

Computation. The experiment was performed on a 2.8GHz machine. For each frame, 2500 Markov state samples were generated, which took on average 8 minutes. The system is not optimized and future effort will be directed to improve the efficiency.

Trials		Weighted Ave. Error (pixels)
A	*Proposed Method*	23.54
B	*without backward proposals*	27.55
C	*without forward or backward proposals*	31.72

Table 1. Weighted Average Error.

Trial	Error Type	Upper Body	Lower Body	Whole Body
A	**X**	9.66	9.25	9.54
	Y	8.07	14.64	9.97
	Image(XY)	13.98	18.86	15.39
	Depth(Z)	13.41	18.32	14.83
	Euclidean	21.54	28.50	23.54

Table 2. Breakdown of the Error.

6 Conclusion

We described a novel method for integrating bottom-up techniques into a human pose tracking scheme. Tracking based solely on predictions from previous

frames is not reliable as these may contain significant errors due to self-occlusion; our approach allows for self-initialization and re-initialization by integrating component cues from current observations.

In comparison with previous work, our problem is substantially more difficult and complex. The proposed method helps improve the robustness of pose tracking in realistic scenario as demonstrated by the experiments. The results were achieved without manual initialization of the body pose and shape and without prior learning of the person's specific appearance or movement.

Currently, our system requires color images as input. Shape-based features to detect body parts would allow monochrome input and help reduce the computation significantly; this will be a focus of our future research.

Acknowledgment

This research was funded, in part, by the Advanced Research and Development Activity of the U.S. Government under contract # MDA-904-03-C-1786.

7 References

[1] M. Lee, I. Cohen, "Proposal Maps driven MCMC for Estimating Human Body Pose in Static Images," *CVPR 2004.*

[2] M. Lee, R. Nevatia, "Dynamic Human Pose Estimation using Markov chain Monte Carlo Approach," *Motion 2005*.

[3] K. Choo, D.J. Fleet. People tracking with hybrid Monte Carlo, *ICCV 2001.*

[4] D. Demirdjian. Combining Geometric- and View-Based Approaches for Articulated Pose Estimation. *ECCV 2004*.

[5] J. Deutscher, A. Davison, I. Reid. Automatic partitioning of high dimensional search spaces associated with articulated body motion capture, *CVPR 2001.*

[6] W. Gilks, S. Richardson, D. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, 1996.

[7] R. Neal, M. Beal, and S. Roweis. Inferring State Sequences for Non-Linear Systems with Embedded Hidden Markov Models. *NIPS 2003*.

[8] G. Mori, X. Ren, A. Efros, J. Malik, "Recovering Human Body Configurations: Combining Segmentation and Recognition", *CVPR 2004*.

[9] D. Ramanan, D. A. Forsyth, "Finding and tracking people from the bottom up," *CVPR 2003.*

[10] T. J. Roberts, S. J. McKenna, I. W. Ricketts: Human Pose Estimation Using Learnt Probabilistic Region Similarities and Partial Configurations. *ECCV 2004*.

[11] L. Sigal, S. Bhatia, S. Roth, M. J. Black, M. Isard, Tracking Loose-limbed People, *CVPR 2004*.

[12] C. Sminchisescu, B. Triggs. Kinematic Jump Processes for Monocular Human Tracking, *CVPR 2003*.

[13] C. Sminchisescu, A. Jepson. Variational Mixture Smoothing for Non-Linear Dynamical Systems, *CVPR 2004*.

[14] Z. W. Tu, S.C. Zhu, Image Segmentation by Data-Driven Markov Chain Monte Carlo, *PAMI 24(5)*, pp. 657-672, 2002.

[15] P. Viola, M. Jones. Rapid object detection using a boosted cascade of simple features, *CVPR 2001.*

[16] S. Zhu, R. Zhang, Z. Tu. Integrating bottom-up/top-down for object recognition by data driven Markov chain Monte Carlo, *CVPR 2000.*

[17] J. Zhang, R. Collins, Y. Liu, "Representation and Matching of Articulated Shapes," *CVPR 2004.*

[18] T. Zhao, R. Nevatia, Tracking Multiple Humans in Crowded Environment, *CVPR 2004.*

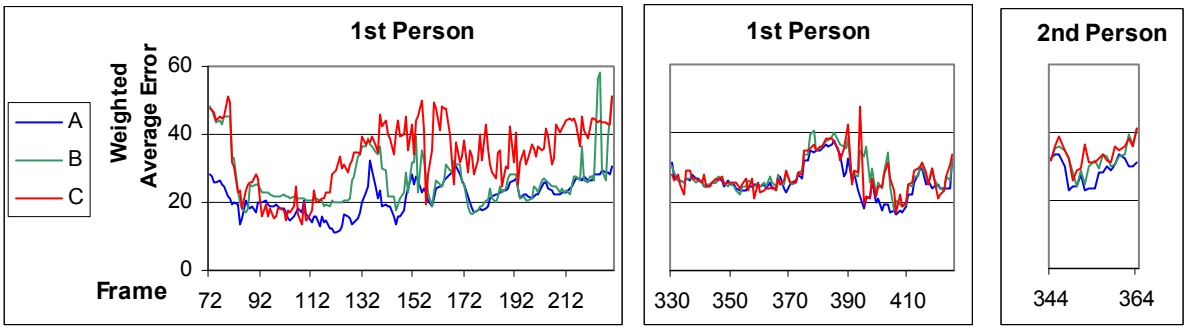


Figure 9. Weighted Average Error.

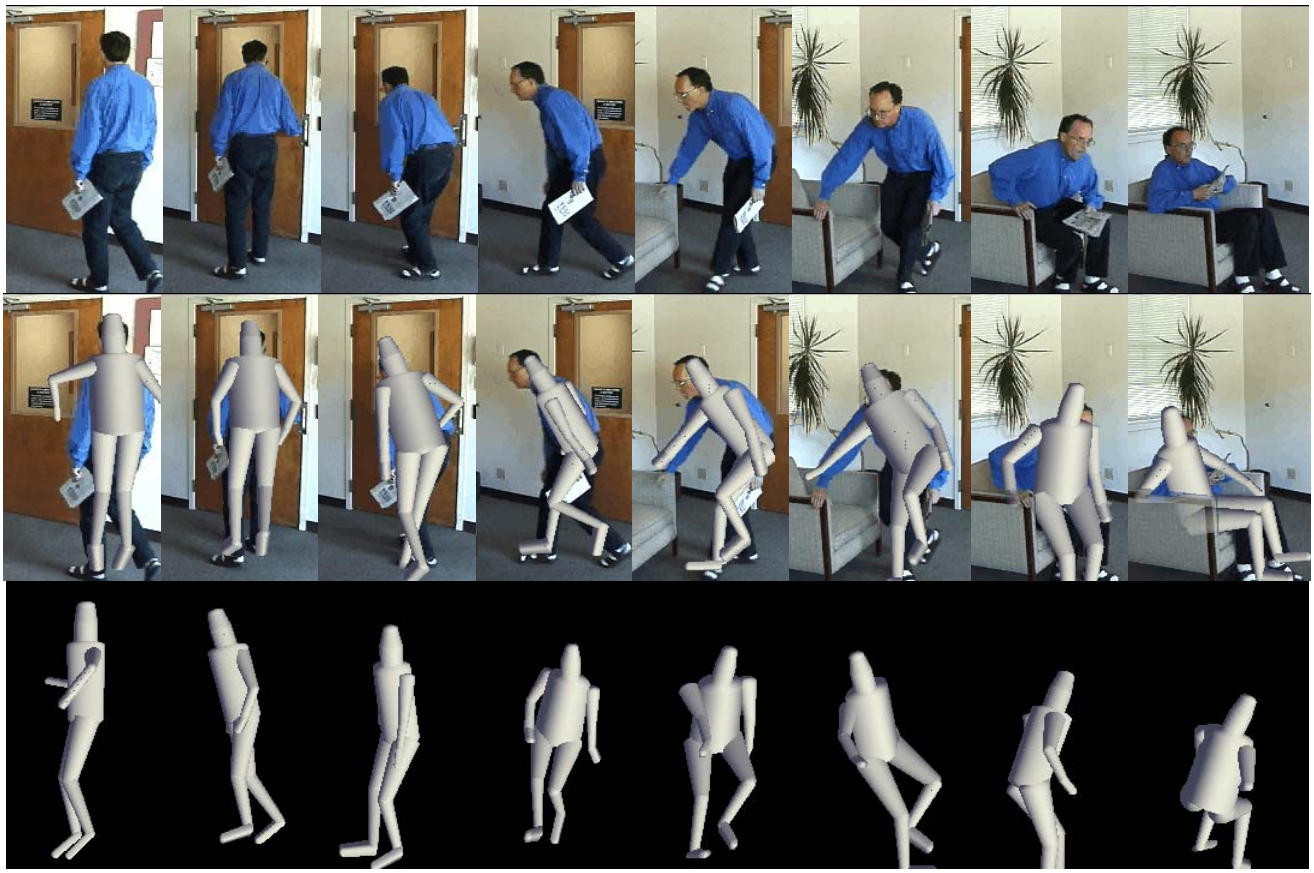


Figure 10. Pose Estimation. On separate rows are shown the original images, estimated pose in image view, and in depth. Errors include (i) the arms in the first frame; (ii) confusion over left and right legs (Frame 147, 152); and (iii) the feet are too low in Frame 162. In later frames, parts of the body are occluded by the chair's arm, as illustrated by a semi-transparent layer. The occluding mask is provided as prior scene information. Unintentionally, the person's footwear has a distinct pattern, but this is not used as a marker to aid tracking. The camera was static while the images above were cropped for display.

Figure 11. Tracking during video segment that includes two people and moving door. For the left person, errors include (i) the arms in Frame 360, 412, (ii) confusion of left and right arms in 377, and (iii) feet in 377, (iv) elbow in 397. For the right person, errors include (i) right foot in 350, (ii) left arm in 350, (iii) both arms in 360.