Monocular Human Pose Tracking using Multi Frame Part Dynamics

Vivek Kumar Singh Ramakant Nevatia
University of Southern California
Los Angeles, CA
{viveksin|nevatia}@usc.edu

Abstract

Efficient monocular human pose tracking in dynamic scenes is an important problem. Existing pose tracking methods either use activity priors to restrict the search space, or use generative body models with weak kinematic constraints to infer pose over multiple frames; these often tends to be slow. We develop an efficient algorithm to track human pose by estimating multi-frame body dynamics without activity priors. We present a monte-carlo approximation of the body dynamics using spatio-temporal distributions over part tracks. To obtain tracks that favor kinematically feasible body poses, we propose a novel "kinematically constrained" particle filtering approach which results in more accurate pose tracking than other stochastic approaches that use single frame priors. We demonstrate the effectiveness of our approach on videos with actors performing various actions in indoor dynamic scenes.

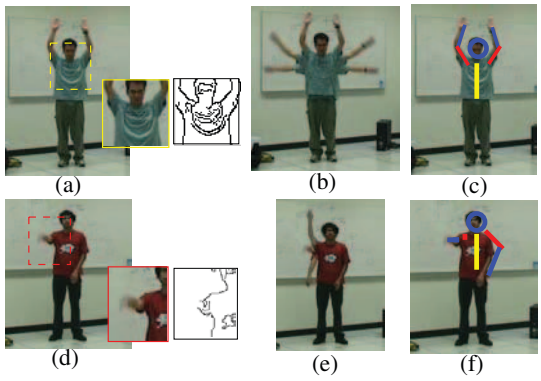


Figure 1. Challenges in Pose Tracking. (a), (d) show sample shots of actors performing limb actions in natural environments, labelled with common problems such as dense spurious edges due to clothing and missed edges due to motion blur. (b), (e) show multiple frames overlayed around the shots shown in (a) and (d) respectively; and (c), (d) show the true pose.

1. Introduction

Human pose tracking is a key problem in computer vision as it finds applications in surveillance, human activity understanding, motion capture etc. The articulate structure of human body together with variations in clothing appearance and backgrounds makes this task very challenging. We aim to persistently track upright 2D human poses in monocular videos with dynamic backgrounds. In dynamic backgrounds without the benefit of having a background model, frame-by-frame pose tracking through various pose variations often result in detection misses (see figure 1 for some problem cases). These detection misses pose major difficulties for persistent tracking. While using discriminative color appearance models have been demonstrated to work well [12], [4], here we focus our attention on using shape observations over multiple frames to disambiguate the pose estimates. Existing pose tracking algorithms tend to model multi frame dynamics of the human pose by restricting the space using activity priors [18, 2] or by imposing weak frame-by-frame spatial restrictions [12, 9, 17]. Frame-by-frame spatial restrictions fail to capture the articulate pose dynamics reliably and results in tracking failures.

In this work, we propose an approach to approximate the dynamics of a pose over a temporal window without imposing activity restrictions and use it to track the human pose. Using multi frame dynamics makes our system more robust to track losses (by avoiding drifting) as compared to previous approaches. For persistent tracking, we use a particle filtering framework to track the human pose. In particular, we use a Hierarchical HMM with a tree structured part model [3] [16] at the lower level that generates proposals for full body pose on the top level. To reduce the part search, we apply a person detector and then search for parts in and around the detection window. We use the sampled parts to initialize a particle filters and obtain multiple tracklets over a temporal window. These tracklets are then used to reweight the part samples. The advantage of using part tracklets is two fold. First, track-reweighted part samples used for estimating the pose follows a locally smooth trajec-

978-1-4244-5501-0/09/$25.00 ©2009 IEEE

tory. Secondly, the part trajectories can be used to predict part estimates otherwise missed by the part detectors. This also allows us to maintain fewer, more confident samples, and thus the overhead of computing part tracklets is low.

1.1. Related Work

Several methods have been developed towards solving human pose tracking problem under varied but often restricted conditions such as a known background, prior knowledge of activities being performed, and viewpoints. Based on the pose search methods, these can be categorized as discriminative or generative. The discriminative model based methods [15], [18] uses a training set to learn a mapping between the commonly observed features and their respective poses. However in order to keep the inference tractable, these methods often impose a strong pose prior with restricted viewpoints, pose variations and often assume that a clean segmentation of human pose is available. The generative counterpart [17], [12], [11], [4] uses a part-based model, with parts kinematically constrained by a tree structure (pictorial structures [3]), and hence can search and find more general poses. These methods use generic part detectors which are often quite noisy, and hence require an expensive inference algorithm such as message passing, integer linear programming [13], [8] to infer the most probable pose in each frame. Following the success of detection based tracking methods for faces and pedestrians, these methods offer an attractive option for tracking human poses in a similar frame-by-frame detection based tracking framework. However in natural setting, more often than not, inferring human pose from a single frame is difficult and sometimes even practically impossible (see figure 1 for illustrations). However when we observe these videos over multiple frames, the problem becomes easier, clearly implying the importance of dynamics.

Some approaches have been recently developed that perform 2D pose tracking through challenging movie sequences [12, 4] using tree structured part models and discriminatively updating appearance color models. However, in dynamic scenes with drastic appearance changes, maintaining color models (especially for body parts) becomes hard and a noisy appearance model may lead to tracking errors. So instead of putting in extra effort in maintaining a consistent appearance model, here we focus our efforts on disambiguating the part estimates using the part shape observations over multiple frames.

For estimating pose tracks over multiple frames without activity priors, [5, 17, 4] extend the tree model over multiple frames and use message passing for inference. However, as the number of frames increases, these methods slow down due to increase in the search space. [19] approximate dynamics by maintaining velocities of the joints with the pose (*dynamic pose*). The likelihood of a dynamic pose using observations in current and next frame. To ensure kinematically correct pose tracking, they use a physics based engine in a motion control loop. Since we are working with 2D image sequences with a potentially moving camera, maintaining a physical model for body-world interaction becomes complex.

Recently, hierarchical frameworks for pose tracking [16, 4, 9] have been proposed that progressively reduce the search space and hence speed up the inference process. [4] uses dense part representation and restrict the search using a HOG-SVM body detector. The search space is reduced further using a graph cut segmentation to obtain a potential person mask. For inference, belief propagation over multiple-frames is used to exhaustive search for parts. The transition potential over time is modeled with simple box neighborhood constraints, which are fairly weak and may only work with dense representation. Here, we present an alternate approach for estimating pose by approximating the multi-frame part dynamics with Monte-Carlo methods and using it to better constrain the transitions over time. Our approach is more efficient and sometimes more accurate than methods using belief propagation over multi-frames.

Rest of the paper is structured as follows. Section 2 describes the particle filtering based pose tracking framework. Section 3 describes the part sampling method for a single frame. Section 4 describes our novel part sampling approach based on locally estimated part dynamics. Section 5 includes experiments over different parameters and evaluation results, followed by the conclusion.

2. Pose Tracking using Particle Filtering

In this section, we describe the particle filtering framework for tracking articulate poses using a hierarchical graphical model. Similar frameworks have been used in previous works and here we put these works in perspective and discuss how our approach differs from the existing ones.

2.1. Representation

We represent the full pose state $\mathbf{x}$ using coordinates of the body joints, and the body parts using state variables $\{x_i\}$.

The hierarchical graphical model comprises of the full object pose (2D human pose) at the top level and parts at the lower level. Parts are kinematically constrained using a generative tree structured graphical model (pictorial structures) with torso at the root with head and limbs as its branches [3]. The suggested graphical model is shown in figure 2. Each state node x has an observation y (not shown in the figure for clarity). The state variable at time t, x_i^t depends on the state at time $t-1$ (x_i^{t-1}) and the observation sequence over a temporal window $y_i^{t-\delta:t+\delta}$.

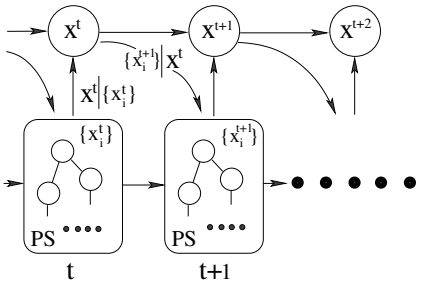


Figure 2. Hierarchical Model for Pose Particle Filtering

2.2. Pose Observation Likelihood

We compute the likelihood of a pose x by matching the projected edges from a configuration of x with the edges points in the image I. For efficiency, we use the Hausdorff distance [7] for matching [10].

$$d_H(\mathbf{x}, I) = \max_{a \in A} \min_{b \in B} ||a - b||$$

where d_H is the weighted directed Hausdorff distance and $||.||$ is any norm. For simplicity and efficiency, we used the euclidean norm between the model points and edge points, weighted by the edge strength (magnitude of gradient).

2.3. Pose Inference

For ease of discussion, here we assume that we already know how to generate sample distribution for individual body parts given a rough estimate of the person scale and position, and focus on the pose inference from the part distributions. In brief, we us shape template based part detectors to hierarchical search from torso to lower parts and use belief propagation to favor kinematically consistent parts. Particle filters are then initialized on sampled parts to get kinematically consistent part tracklets over a window, which are then used to update the current part distribution. The details of the part sampling is discussed in section 3 and 4.

For initialization, we sample full poses $\mathbf{x}$ from the part distribution $p(x_i|y_i^{1:T})$ using importance sampling. This is achieved by sampling torso and then subsequently sampling the children given the sampled torso estimate (see [3] for more details). Since our model is tree structured, we also include the most likely pose samples obtained using dynamic programming over the sampled set of parts [3]. The sampled poses are then reevaluated using global pose likelihood function (eqn 1).

For tracking, we obtain part distributions for the next frame by

- sampling in the neighborhood of the part samples in the current frame i.e $x_i^{t+1} \sim p(x_i^{t+1}|x_i^t, y_i^{t-\delta:t+\delta})$
- sampling in the neighborhood of the pose samples in previous frame i.e $x_i^{t+1} \sim p(x_i^{t+1}|\mathbf{x}^t, y_i^{t-\delta:t+\delta})$; this ensures that the parts that correspond to the globally consistent poses in the current frame are well represented

To favor kinematically consistent pose estimates, the part distributions $\{x_i^{t+1}\}$ are then refined using a round of belief propagation (particle message passing [17]). The pose samples $\mathbf{x}^{t+1}$ are then obtained by sampling parts from the part distributions $\{x_i^{t+1}\}$ using importance sampling and reevaluated using the global pose likelihood.

In order to prevent sample impoverishment, we use importance resampling if the effective sampling size of the sample distribution is small. This use of effective sample size measure for resampling has now become a standard in particle filtering literature. If the likelihood of the best pose stays below a threshold value for n frames, we reinitialize the tracker (ignoring the pose samples in the previous frame). This is necessary in order to avoid drifting.

The final track is obtained by finding the maximum likelihood path (using Viterbi algorithm) over the sampled pose distribution over the entire sequence.

2.4. Discussion on related approaches

The success of stochastic tracking methods depends on the efficiency and accuracy of the sampling process. The key difference from existing models is that the part likelihood depends on observations over multiple frames. This allows us to estimate multi-frame dynamics of a part which we then use to obtain better part sample distributions. In particular, we use part tracklet distributions to approximate body dynamics, which carries more accurate information than first order dynamics [19]. We achieve this without using the expensive multi-frame belief propagation [4], action priors [2, 10] or pose priors [19, 14]. Note that estimation of multi-frame dynamics must be efficient as the part distribution in a frame depends on the pose estimates in the previous frame, so multi-frame estimation must be done at each frame. This step further slows down the tracking based on multi-frame belief propagation.

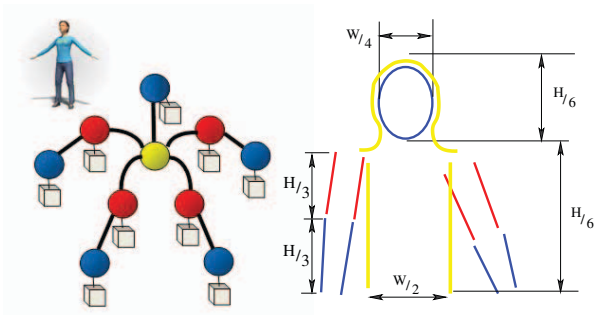
3. Part Sampling: Without priors

To generate part distributions in a single frame, we use the generative part model (pictorial structures) proposed in [3, 17]. In these models, each part is represented as a node in the graph and each edge represent the kinematic relation between the connected parts. Likelihood of each part is computed by applying a part detector over the image, followed by belief propagation to infer kinematically feasible pose. We use particle approximations instead of dense representation of part spaces [11] so in that respect our model is closer to [17, 6]. The likelihood of the generative part

model is given by

$$arg\min_{\{x_i\}} \left(\sum_{i=1}^{n} \phi_i(x_i, I) + \sum_{(x_i,x_j)\in E} \psi_{ij}(x_i, x_j) \right)$$

where, $\{x_i\}_1^n$ correspond to the set of body parts, ϕ_i is the log likelihood function of part x_i and ψ_{ij} is the potential function that model the kinematic constraints between parts x_i and x_j.



(a) (b)

Figure 3. Pose Model: (a) Tree structure generative part model for full body (b) Part templates used for detection; note that the marked template dimensions is approximately the mean size, for robustness multiple templates around the mean are applied

3.1. Person Detection

For person detection task, we use a shape based full body detector similar to that in [20]. The detector is trained by boosting over edgelet features and is quite efficient. For robustness, we use multiple detectors with varying detection rate and efficiency. More specifically we use a reliable high confidence detector trained to give high accuracy and low false alarm rate, and a low confidence but more efficient detector with potentially greater false alarm rate but fewer missed detections. Only the detection responses from the strong detector are used for track initialization. The transition likelihood between the detections is modeled with a zero mean Gaussian.

3.2. Detection of Body Parts

For efficiency, we use simple shape template models (geometric constructs) for finding body parts in the image. These shape templates are trained over manually annotated samples. We represent the state x for each part using a 4 element tuple, $\langle p_x, p_y, s, \theta \rangle$ i.e the position, scale and orientation. Figure 3(b) shows the mean 2D shape templates for each body part. Note however that the figure only shows the mean sizes, the search for each part is done in local neighborhood of its mean size.

Likelihood Model: The shape likelihood $\phi(x)$ for each sample x combines both the strength and orientation of the gradient at each point in the model.

$$\phi(x) \quad = \quad \sum_{x_i \in x} d_{mag}(I(x_i)) \times d_{ori}(x_i, I(x_i))$$

where, $d_{mag}(I(p))$ is the Euclidean distance to the nearest edge pixel from the image point p, weighted by the edge strength. This can be calculated very efficiently using generalized distance transform over the edge likelihood map [3]; $d_{ori}(p, I(p))$ is the orientation likelihood, which is the dot product between the normals at the model point p and corresponding point in the image $I(p)$ (quantized into 8 bins). In our experiments, we observed that using orientation information greatly improves the detection accuracy compared to only using edge correlation.

Hierarchical Part Search: The dense evaluation of the part templates (as in [11]) over all position, scales and orientations is expensive. [4] uses person detector to reduce the scale search but still does a dense search over the reduced space. For greater efficiency, we exploit the tree structure of the model and use a hierarchical search by restricting search for a part in the neighborhood of the parent.

More specifically, given a rough scale and position estimate from the person detector, we apply head and torso detector at appropriate scale and position masks. Note that the person detector detects upright person, so it sufficiently restricts the head and torso position. We also update the scale and position estimate of the person based on the sampled head and torso estimates. This refinement of person scale reduces jitter in scale estimate and hence reduce the search space for limb detection. Given the head and torso samples, we generate the shoulder masks. For more accurate masks, we apply an omega template detector [9] for head shoulder contour to locate the shoulder joints given a head and torso locations. The shoulder estimates significantly reduce the position search space for finding upper arms. We then perform a dense search for upper limbs (arms and legs) over all orientation with proximal joint within the mask. The response from the upper limb detections are then used as seed mask to restrict the search for lower limbs.

3.3. Refining Part Distributions using Particle Message Passing

From part detection process, we obtain samples for each part. These sample approximated part distributions are then used to sample full body pose. However due to weak nature of the part detectors, often times even if the correct hypotheses are generated by the part detector, they have a low likelihood score, and may not be selected during pose sampling. Furthermore due to the restricted (tree ordered) nature of the part search process, spurious part samples get generated in the background especially when the background is

cluttered. One may increase the sample set for each part, but that makes the inference inefficient. To favor the selection of part hypotheses that result in kinematically connected poses, we run one iteration of message passing between the part nodes. For efficient message passing, we use a factor graph representation of the tree structure. The message passing methods using particle approximations have now become quite common and we refer reader to [17], [6] on how to compute the messages and belief update rules.

Figure 4 shows a stage wise results during part sampling on a sample image (from our experiments).

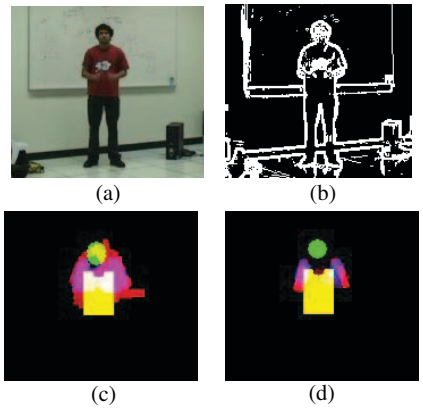


Figure 4. Part Sampling in single frame. (a) sample image with detected person (after smoothing); (b) edge map. (c) part distribution obtained by hierarchical search: upper arm in red, head & lower arm in blue and torso in yellow. (d) Part distribution after particle message passing

4. Part Sampling using Multi Frame Dynamics

In this section, we describe our approach of estimating multi frame body dynamics and using the estimated dynamics to refine/update the part distributions. To model the dynamics, we use Monte Carlo approximation method. In particular, we obtain a distribution over the part trajectories (tracklets) as an approximation to the multi frame body dynamics. Our approximation of the full body dynamics using parts utilizes the tree structure of the model. In order to efficiently sample part tracklets such that the resulting distribution favors kinematically feasible poses, we use *kinematically constrained* particle filtering.

Our approach of using Monte Carlo methods (particle filtering) to perform kinematically constrained sampling follow principles similar to an independent work on Inverse Kinematic Particle Filter [1]. However, [1] presents a generic filtering framework for fitting an articulate structure given position constraints and simultaneously avoiding complex matrix inversions, while ours is a data driven approach where the objective is to obtain the tracklet distribution that best approximates the data and simultaneously satisfies the kinematic constraints.

Besides updating the current part distributions, the estimated tracklet distributions are also used to generate part samples for the upcoming frames. This also help generate part samples that may otherwise be missed due to the restricted search and size of the part templates.

4.1. Approximating local pose dynamics using parts

To capture the dynamics of the pose $\mathbf{x^t}$ at time t, we use the velocity model in pose space $\dot{\mathbf{x}}^\mathbf{t}$. We approximate the dynamics of the pose using a first order dynamical system

$$\mathbf{x^{t+1}} = \mathbf{x^t} + \dot{\mathbf{x}}^\mathbf{t}\Delta t + \epsilon$$

where ϵ is an error function. This representation is similar to dynamic pose introduced in [19]. If we simply propagate the pose using first order dynamics, the kinematic constraints may get violated. [19] propagates the 3D pose using locally linear velocity model and use physical simulation to ensure kinematic feasibility. In absence of such physical simulation in 2D, we propose to capture the dynamics by decomposing it into parts.

We approximate the pose dynamics by decomposing the pose into the tree structured part model and using first order dynamics on each of the constituent parts. i.e $\dot{x}^t \sim \{\dot{x_i}^t\}$. Note however that independently propagating the part using first order dynamics may violate the kinematic constraints on the full pose. We propose to use the tree structure of the part model to ensure kinematic constraints during part propagation. We first independently obtain the trajectory for the root part i.e the torso. Given the torso trajectory, we sample the head and upper arm trajectories by observing the kinematic constraints on the associated joints. More formally, for each parent-child pair $\langle x_p, x_c \rangle$ in the kinematic tree, we obtain child trajectory by observing the kinematic constraints due to the parent trajectory. Recall that the a part is represented by the tuple $\langle p_x, p_y, s, \theta \rangle$. For each non-root node, fixing the parent trajectory fixes the position of the proximal joint and its velocity. Hence only l and θ need to be sampled. As the estimated track for the parent-child joint obtained from the parent trajectory may not be accurate, we relax the proximal joint position constraint on the child trajectory by allowing it to be within a bounded neighborhood.

Besides kinematics, we also observe constraints on the dynamics such as the rate of change of angle for each body part. To summarize, given the parent trajectory $x_p^{1:T}$, we observe the following constraints on the child trajectory $x_c^{1:T}$,

$$l_c^t \in [0, L_c^{max}) \quad ; \quad \dot{l_c^t} \in [\dot{L_c^{min}}, \dot{L_c^{max}})$$
$$\dot{\theta_c^t} \in [\dot{\theta_c^{min}}, \dot{\theta_c^{max}}) \quad ; \quad dist(x_p^t, x_c^t) \in [0, d)$$

where, $l_c, \dot{l_c}, \dot{\theta}_c$ denote length, radial velocity and angular velocity of x_c at time t, and the corresponding min and max values indicates the feasibility bounds for each; $dist(x_p^t, x_c^t)$ denotes the euclidean distance between the distal joint of x_p and proximal joint of x_c at time t.

4.2. Sampling part tracklets using Kinematically Constrained Particle Filters

Conforming to the *predict-correct* paradigm of particle filtering, we realize constrained particle filtering by predicting the particles using constrained dynamics and compute/update the particle weights based on the observation likelihood and constraint-violation penalty.

Suppose we want to sample a tracklet for a part over a window of n frames that is kinematically consistent with a given parent tracklet sample. We assume that we have part samples available in at least one other frame. This is easy during tracking by simply including the previous frame in the window (fig.5(a)). We initialize particle filters in the current frame and obtain part samples over all n frames using first order transition model for propagation (fig.5(b)). During propagation, we restrict the search region using constraints from parent tracklet sample. We refer to this as constrained particle filtering. Note that the tracklet space over the all part samples over n frames is combinatorial. So we use importance sampling to obtain a tracklet sample, using kinematic likelihood with parent tracklet samples as the importance function (fig.5(c)). The tracklet sample is obtained by sampling a part in each frame. The sampled tracklets are often not smooth due to relaxed constraints and weak edge observations. So we then smooth each tracklet by interpolating the part samples in the tracklet (fig.5(d)). This step is important as it also handles the cases where the observed part is very small in size or is partially occluded.

Figure 5 shows an illustration of the sampling method.

4.3. Resampling parts using the local dynamics

Using constrained particle filters, we now efficiently approximate the dynamics of the human pose by tracklet distributions of body parts. The tracklets are obtained over a temporal window around the current frame for each body part. The score for each tracklet, say t_i for part i, is then computed as a product of the transition and observation likelihoods over all the frames in the temporal window i.e

$$p(t_i) = exp(\sum_{t=0}^{n} \phi_i(x_i^t) + \sum_{t=0}^{n-1} \varphi_i(x_i^t, x_i^{t+1}))$$

where $\varphi_i(.,.)$ gives the transition likelihood for part i with zero mean gaussian. The tracklet distributions for all parts are then sum normalized to 1.

Now using the tracklet distribution for part i, we update the particle weight for each part sample in the current frame.

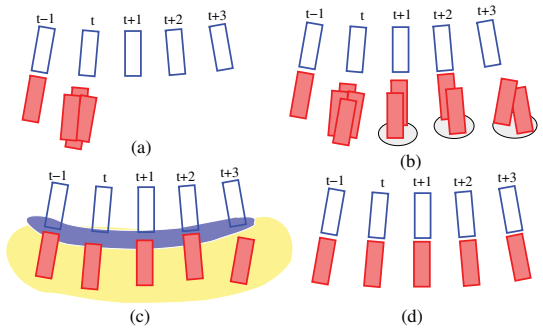


Figure 5. Illustration of Tracklet Sampling (a) Parent Tracklet (in blue), prior and current samples of child part (in red); (b) Child samples obtained by propagating samples (constraint are marked as gray region for distal joint position); (c) Sample tracklet with highlighted kinematic constraints (in blue). Yellow highlight shows the kinematically constrained spatio-temporal volume for the child tracklets (d) Tracklet after smoothing.

The idea is to increase the weight of the samples that are close to tracklets with high score. We define the distance of a part sample x_i from a tracklet t_j to be the distance of the part sample in the tracklet corresponding to the current frame with the given sample x_i. The new weight $\tilde{p}(x_i)$ of x_i is computed using equation 1.

$$\tilde{p}(x_i) = p(x_i) \times \min_j (dist(x_i, t_j) p(t_j)) \quad (1)$$

where, $p(x_i)$ is current sample weight, $dist(x_i, t_j)$ is the distance of the sample x_i from tracklet t_j.

Furthermore, we use this association of part samples with the tracklets to generate part samples in the next frame. Note that the since the part distributions in next frame is affected by the pose samples in the current frame, hence the part samples generated by propagating the parts during the tracklet sampling are no longer sufficient and part propagation using sampled poses is still required.

Figure 6 shows a sample overlaid sequence of a lift arm action 6(d), with part tracklet samples 6(e) and the comparison between the part distribution after message passing 6(c) and those obtained after updating the distribution using tracklets 6(f).

5. Experiments and Results

We tested our approach on videos captured from static as well as hand held cameras in an indoor environment. Videos are captured in a lab environment and have significant camera jitter. Each video has one person performing various limb actions such as flap, point upwards, jumping jack, hand wave, pray (putting both hands together) and walk (towards camera). The entire set has a total of 6 videos with 3 shot indoors with a stationary camera and 3 shot from a hand

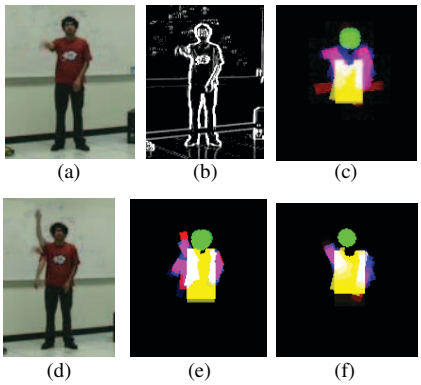


Figure 6. Resampling parts using tracklet distribution (a) sample frame (b) corresponding edge map (c) part distribution after message passing (d) neighboring frames shown on an overlaid mosaic; (e) tracklet distribution over 5 frames; (f) part distribution after updating weights using tracklets

held. Each video is ~ 2000 frames long, with frame size of 420×240. The size of the person varies from ~ 90 pixels to ~ 200 pixels.

Figure 7 shows some sample frames from the dataset and the results obtained by applying our method. The results clearly indicate that our method successfully finds the body pose across various scales and in presence of cluttered edges and significant camera jitter.

Quantitative Evaluation: To quantitatively measure the performance of our system, we marked ground truth poses for every 50th frame in each sequence (~ 300 pose annotations). To evaluate the pose estimates, we use the same metric as in [4]. An estimated body part is considered correct if it lies within 25% of the length of the ground-truth segment from the annotated location. With this metric, the person detection itself gives quite accurate results for head and torso, so we evaluate our system only on limbs. On the average, we correctly find the parts about 84% of the time. The average error of the estimated joint positions over the correctly detected parts was about $\sim$10 pixels.

Table 1 shows the comparitive results. Method A corresponds to pose tracks obtained by finding the MAP estimate from the parts using belief propagation i.e without using transition model for pose. The low accuracy score indicates that the part distributions obtained by the bottom-up part search are not accurate enough for reliable pose tracking. Method B corresponds to our implementation of pictorial structures with frame-by-frame transitions (similar to [17]). In both stationary and non-stationary cases, the method C (proposed algorithm) is about 10% more accurate. We attribute the increase in accuracy to reduction in tracking losses. This is due to multiple frame lookahead in our approach as opposed to just using observations in the current frame.

	Method	Static Scene	Non-Stationary Camera
A	MAP-Parts	0.693	0.62
B	First-Order PF	0.805	0.73
C	Multi-Frame Dynamics	0.89	0.82
D	Strike-a-Pose [12]	0.45	0.38

Table 1. Quantitative Results on dataset. Accuracy is computed as fraction of arms correctly detected;

To put results in perspective with state-or-art algorithms, we applied strike-a-pose approach [12] on our dataset. [12] finds key poses in the first pass and use the appearance model learnt from the detected keyposes to find the person in each frame. We use the software provided by the author. The softwares requires a scale estimate such that the size of person torso is roughly ~ 50 pixels (during the keypose search). We set the scale to be 0.6 as it gave the best results.

While the proposed tracker works quite well, it still results in some tracking failures and occasional track discontinuities. Such failures usually occur in the presence of consistent observation ambiguities such as edge distractions from clothing. In the event of such failures, the tracker gets re-initialized based on bottom up part estimates thereby regaining the correct pose but with occasional track discontinuities. Such failure cases may be dealt with by increasing the lookahead window size for estimating track dynamics and/or using more particles, but both these steps would add to the computational load. An alternative solution would be to use better observations such as appearance models for each body part [12].

Analysis of speed: Over the entire dataset, our system takes about ~ 1.2 sec to process each frame (with temporal window size of 5). Out of 1.2 sec, person detection takes about 0.3 sec. The overhead induced by the resampling of part using local dynamics was only ~ 0.4 sec per frame. Note that the processing time of the system is dependent on the number of part and pose hypotheses maintained for each frame. In our experiments, we maintained 30 upper body hypotheses for tracking; 50 particles for each body part.

6. Conclusion

We presented an efficient approach for 2D pose tracking that estimates the pose dynamics over multiple frames and uses it to assist the full body tracking. We present an approximation of the spatio-temporal volume of human pose dynamics using part tracks and proposed a principled approach to efficiently sample kinematically consistent part tracks. We demonstrated our approach on videos with significant camera jitter and dynamic background. We have quantitatively shown that a pose tracking system greatly benefits from using part trajectories to approximate the dynamics. In future, we plan to incorporate the use to local

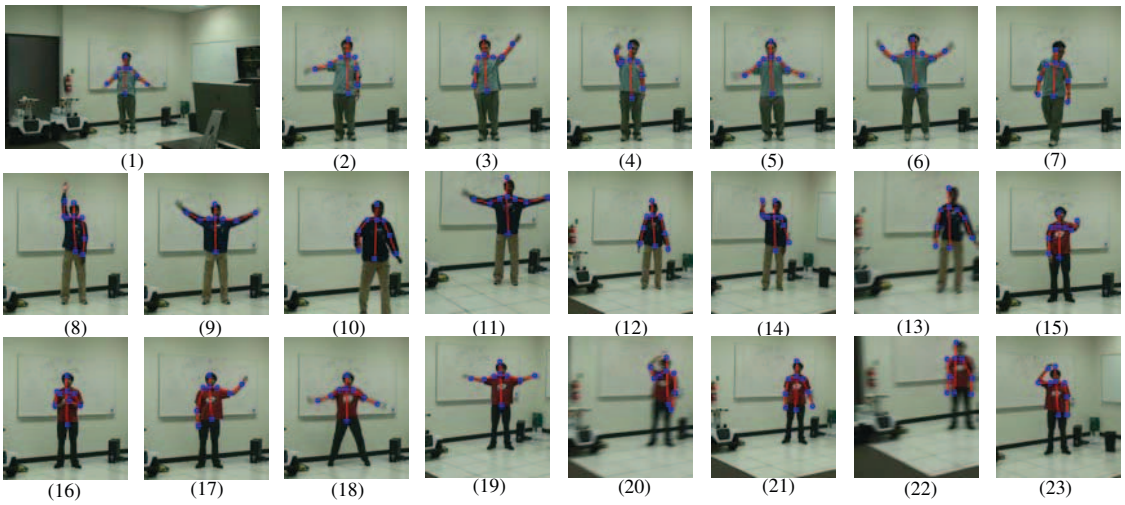


Figure 7. Pose Tracking Results on videos with actors performing various arm gestures - flap, lift up, jumping jack, walk. (1) shows the full camera view, (2)-(23) show sample pose results overlaid on cropped images (cropped around image center for clarity); (5), (18) has severe motion blur due to fast arm movement and (20), (22) show blur due to sudden camera movement.

discriminative appearance based trackers to further assist the overall tracker, in a hierarchical framework.

Acknowledgements

This research was funded, in part, by the U.S. Government VACE program.

References

[1] N. Courty and E. Arnaud. Inverse kinematics using sequential monte carlo methods. In *AMDO*, july 2008.

[2] A. Elgammal and C. su Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *CVPR*, pages 681–688, 2004.

[3] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.

[4] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, 2008.

[5] J. Gao and J. Shi. Multiple frame motion inference using belief propagation. In *AFGR*, pages 875–882, 2004.

[6] G. Hua, M.-H. Yang, and Y. Wu. Learning to estimate human pose with data driven belief propagation. In *CVPR*, volume 2, pages 747–754 vol. 2, June 2005.

[7] D. Huttenlocher, G. Klanderman, and W. Rucklidge. Comparing images using the hausdorff distance. *T-PAMI*, 15(9):850–863, 1993.

[8] H. Jiang and D. Martin. Global pose estimation using non-tree models. In *CVPR*, June 2008.

[9] M. Lee and R. Nevatia. Human pose tracking using multi-level structured models. In *ECCV (3)*, pages 368–381, 2006.

[10] P. Natarajan and R. Nevatia. View and scale invariant action recognition using multiview shape-flow models. In *CVPR*, 2008.

[11] D. Ramanan. Learning to parse images of articulated bodies. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *NIPS 19*, pages 1129–1136, 2007.

[12] D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: tracking people by finding stylized poses. In *CVPR*, pages 271–278, 2005.

[13] X. Ren, A. C. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *ICCV*, pages 824–831, 2005.

[14] R. Rosales and S. Sclaroff. Combining generative and discriminative models in a framework for articulated pose estimation. *IJCV*, 67(3):251–276, 2006.

[15] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *ICCV*, 2003.

[16] L. Sigal and M. J. Black. Predicting 3d people from 2d pictures. In *AMDO*, pages 185–195, 2006.

[17] L. Sigal, B. Sidharth, S. Roth, M. Black, and M. Isard. Tracking loose-limbed people. In *CVPR*, volume I, pages 421–428, June 2004.

[18] R. Urtasun and T. Darrell. Sparse probabilistic regression for activity-independent human pose inference. In *CVPR*, June 2008.

[19] M. Vondrak, L. Sigal, and O. Jenkins. Physical simulation for probabilistic motion tracking. In *CVPR*, Jun 2008.

[20] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, pages 90–97, 2005.