Constraint Integration for Multiview Pose Estimation of Humans with Self-Occlusions

Abhinav Gupta*
Dept. of Computer Science
Univ. of Maryland
College Park, MD, USA
agupta@cs.umd.edu

Anurag Mittal*
Dept. of Computer Science
IIT Madras
Chennai, India
amittal@cse.iitm.ernet.in

Larry S. Davis
Dept. of Computer Science
Univ. of Maryland
College Park, MD, USA
lsd@cs.umd.edu

Abstract

Detection of articulated objects such as humans is an important task in computer vision. We present a system that incorporates a variety of constraints in a unified multi-view framework to automatically detect humans in possibly crowded scenes. These constraints include the kinematic constraints, the occlusion of one part by another and the high correlation between the appearance of parts such as the two arms. The graphical structure (non-tree) obtained is optimized in a nonparametric belief propagation framework using prior based search.

1. Introduction

Detection and tracking of humans in crowded scenes is an important, albeit unsolved problem in computer vision. The problem is hard because of occlusions, a high dimensional problem space and high variability in the appearance of humans due to body shape and clothing. Most prior work has focussed solely on tracking, where initialization is given[3, 17]. Recently, there has been a focus on automatic detection of body pose that could then be used to initialize/re-initialize tracking systems[5, 10, 18].

There are a wide range of approaches to human pose estimation, much of which models human body as a tree structure where each part is represented by a node in the tree and there is an edge joining parts between which there are kinematic relations. The edges impose constraints on the locations of different parts. These constraints may be applied either in 2D [4, 13] or 3D [18]. Felzenszwalb et. al. [5] presented a deterministic linear time algorithm using dynamic programming to solve for the best pose configuration in such tree structures. Other optimization approaches like Data Driven Belief Propagation [6] and Markov Chain Monte Carlo algorithm [10] have also been used to estimate the probability distributions of the locations of body parts.

However, there are limitations to a tree structure. Kinematic relations between parts that are not connected to each other cannot be represented. Furthermore, occlusion of one part by another cannot be modeled nor can the constraint due to the high correlation between the appearance of parts such as the hands [12].

There has been some recent work to overcome these limitations. Lan et. al [9] use factor graphs to add constraints like the balance of a body while walking; Ren et. al [14] use Integer Quadratic Programming (IQP) to add pairwise constraints such as similarity in the appearance of left and right body-parts.

Sigal et. al [18] present an approach to detect and track humans from multiple views using Non-Parametric Belief Propagation(NBP). However, they do not include the constraints for occlusion and similarity in appearance of certain body parts in the same framework.

Ioffe et. al [7] proposed using a mixture of trees to handle occlusions. The mixture includes all possible trees resulting from removing nodes from the base tree under different occlusion scenarios. However, modeling the conditionals between non-connected parts is very difficult; it does not provide very strong constraints, leading to false part localizations. At the same time, the problem space becomes very large due to the need to optimize over the entire ensemble of trees.

Sudderth et. al [19] handle a different but related problem of tracking human hand under self occlusion using NBP. They use only a single camera for 3D tracking of hand and solve for occlusion using posterior optimization. They augment the state of each particle by a set of binary hidden variables which signify the set of occluded pixels. This not only makes the whole computation expensive but also error prone when there is ambiguity.

*Part of this work was done while the authors were at Siemens Corporate Research in Princeton, New Jersey

The problem can be simplified by assuming that one can segment the person, say using background subtraction [2, 9, 11]. While this reduces the search space significantly, these approaches generally do not handle self-occlusion or people occluding one another.

A complementary approach [1, 16] is to learn pose configurations from training images and sequences. Like all appearance based techniques, they have difficulty generalizing to new views or unconventional poses.

In this paper, a multiple camera based approach for estimating the 3D pose of humans in a crowded scene is presented. The system incorporates a variety of constraints, including the occlusion of one part by another and appearance consistency across parts, in a unified framework. The paper presents an efficient approach to solve for occlusion using geometry which is less prone to errors than [19], with evidences being gathered from cameras where there is no occlusion.

The paper is organized as follows. In section 2 we discuss our human body model followed by a discussion on how to pass information between the parts in section 3. Section 4 provides a description of visibility analysis and likelihood computations. We provide a system overview in section 5. Finally, results are presented in section 6 before concluding in section 7.

2 Modeling the Human Body and Problem Formulation

Our 3D human body model (Figure 1) consists of $n = 10$ body parts (head, torso, left upper arm etc.). Each body part (except the torso which is modeled as a cuboid) is modeled as a cylinder and is represented by a node in a graph associated with a random vector $\Phi_i = (l_i, \psi_i)$, where l_i and ψ_i represent the location and appearance parameters of part i respectively. The location of each part, l_i, is further parameterized by $l_i = (l_i^s, l_i^e)$ where l_i^s is the 3D position of the starting point of the limb and l_i^e is the 3D position of the ending point of the limb.

The nodes of the graph are connected by three types of edges. The first enforces kinematic constraints between parts. To obtain a tree model, like those typically used, one would only connect parts using edges of this type. The second represents appearance constraints which are introduced by the symmetry of left and right body part appearances. The third represents occlusion constraints across parts that can occlude each other. The model is represented by $\theta = (E_1, E_2, E_3, c^1, c^2, c^3)$, where the set of edges E_1, E_2 and E_3 indicates which parts are connected by edges of the first, second and third type respectively; c^1, c^2 and c^3 are the connection parameters for these edges.

Our goal is to find the probability distribution of the pose configuration of a human body, given by $L \equiv (\Phi_1, \Phi_2........\Phi_n)$. In an M camera setup, if I_j denotes the image from the j^{th} camera, then $P(I_1....I_M|L)$ is the likelihood of observing the set of images given the 3D locations and appearances of the body parts. The distribution of $P(L)$ is the prior over the possible configurations. The goal is to maximize the posterior distribution, $P(L|I_1....I_M)$, which measures the probability of a particular configuration of the human body given M views and the object model. Using Bayes' rule,

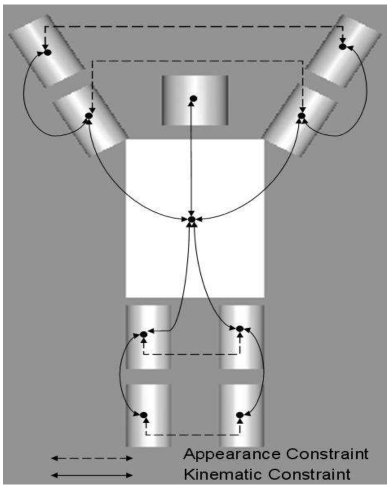


Figure 1. The human model. The solid lines represent edges in set E_1 and dashed lines represent edges in set E_2. Occlusion edges are not shown in the above graph. Every part is connected to all other parts by occlusion constraint edge.

$$P(L|I_1....I_M) \propto P(I_1....I_M|L)P(L) \qquad (1)$$

Assuming that the location and appearance priors are independent of each other, the prior distribution $P(L)$ is

$$P(L) = P(l_1.....l_n)P(\psi_1.....\psi_n) \qquad (2)$$

The prior distribution over the object part locations and appearances are modeled by two separate Markov random fields with edge sets E_1 and E_2. The joint distribution for the tree-structured prior defined by E_1 can be expressed as:

$$P(l_1, l_2...l_n) = \frac{\prod_{(v_i,v_j)\in E_1} P(l_i, l_j)}{\prod_{v_i \in V} p(l_i)^{deg(v_i)-1}} \qquad (3)$$

where V is the set of nodes in the graph and $deg(v_i)$ is the degree of vertex, v_i, in the tree, $G = (V, E_1)$ (subgraph consisting of edges in E_1 only). A similar expression can be written for $P(\psi_1, \psi_2....., \psi_n)$. Since any absolute location or appearance is not preferred over another, the terms representing the priors for single part locations can be neglected. Furthermore, as in most prior work[9, 18, 19], potential functions rather than distributions are used to avoid normalization computations. Then, one obtains:

$$P(l_1, l_2......l_n) \propto \prod_{(v_i,v_j)\in E_1} \varphi_{ij}(l_i, l_j) \tag{4}$$

$$P(\psi_1, \psi_2....., \psi_n) \propto \prod_{(v_i,v_j)\in E_2} \phi_{ij}(\psi_i, \psi_j) \tag{5}$$

where φ_{ij} and ϕ_{ij} are the potential functions over the cliques.

For articulated objects, pair of parts are connected by flexible joints. Ideally, the distance between the ending-point of the first part and the starting point of the second connected part in 3D should be zero. Thus, the clique potential for a pair of parts, connected by edges in E_1, can be modeled as a gaussian:

$$\varphi_{ij}(l_i, l_j) = \mathcal{N}(d(l_i, l_j), 0, \sigma^1_{ij}) \tag{6}$$

where $d(l_i, l_j)$ denotes the euclidean distance between the points l^e_i and l^s_j.

For appearance constraints, let $D(\psi_i, \psi_j)$ denote the distance between two appearance vectors. Ideally, the distance should be zero, assuming left and right body parts have similar appearance. The appearance potential, ϕ_{ij}, is modeled as:

$$\phi_{ij}(\psi_i, \psi_j) = \mathcal{N}(D(\psi_i, \psi_j), 0, \sigma^2_{ij}) \tag{7}$$

Section 4.2 discusses how part appearances are modeled and how the distance, $D(\psi_i, \psi_j)$, is computed.

The computation of the likelihood $P(I_1....I_M|L)$ is tricky due to the consideration of occlusion. The imaging of every camera is modeled as a conditionally independent processes. Similarly, the observation of different parts is assumed to be conditionally independent. This allows us to decompose the likelihood as:

$$P(I_1....I_M|L) \propto \prod_{i=1}^{n}\prod_{j=1}^{M} P_i(I_j|l_1...l_n, \psi_i) \tag{8}$$

Note that, due to the possibility of occlusion, the likelihood of each part depends not only on the position of the part, but also on the positions of other parts. While one may be able to use the likelihood in this form in tracking applications, using it for automatic "detection" is prohibitively expensive. To overcome this, we could introduce a new set of binary 'visibility' variables $v^j_i(l_i)$, that refer to the visibility of a part i at location l_i from camera j. While these visibility variables would be dependent upon the position of all other parts, the likelihood for part i would be independent of the location of other parts if its visibility were given. Then, one could write the likelihood, $P(I_1....I_M|L)$, as:

$$\prod_{i=1}^{n}\prod_{j=1}^{M} \sum_{v^j_i \in \{T,F\}} P_i(I_j|l_i, v^j_i(l_i))P(v^j_i(l_i)|l_1....l_{i-1}, l_{i+1}....l_n) \tag{9}$$

The term $P_i(I_j|l_i, v^j_i(l_i) = TRUE)$ represents the likelihood of observing the image from camera j given that the part is visible from this camera while $P_i(I_j|l_i, v^j_i(l_i) = FALSE)$ represents the likelihood of observing the image given that the part is occluded from the camera. However, parts may be partially visible in which case $v^j_i(l_i)$ is neither true nor false. To approximate this, $v^j_i(l_i)$ is defined as the visibility of a random point on the skeleton of the part. In Section 4.1, we discuss how to compute the visibility variables and in section 4.3, we discuss how to compute the likelihoods.

3 Particle Based Belief Propagation

In the previous section, a graphical model for human body parts was developed. In order to solve for the best configuration in such a graphical model, the framework proposed in [18] can be utilized. Essentially, the system optimizes for the posterior of each part and the interactions between different parts are handled via messages in a non-parametric belief propagation framework. A variant of the PAMPAS algorithm is used for non-parametric belief propagation [8]. The framework provides a natural approach for enforcing constraints across parts, including those of occlusion and appearance matching.

There are, essentially, three sets of unknowns that need to be estimated simultaneously: the locations, the appearances and the visibility variables. The probability densities of part location and appearance are represented via Monte Carlo particles while visibility variables are computed from probabilistic occlusion maps.

The following messages are used to pass information to a part:

- The locations of neighboring connected body parts (e.g. the locations of the lower left leg and torso are passed to the upper left leg). These location are used to apply kinematic constraints.

- The appearance of the corresponding symmetric part (e.g. the appearance of the right upper leg is passed to the left upper leg).

- The visibility information from other parts that may occlude this part (e.g. the upper left leg receives the occlusion map from all other parts in order to update its likelihood distribution)

At iteration r, a message m_{ij} from node i to j along an edge in E_1 or E_2 may be represented as:

$$\begin{aligned} m^r_{ij}(\Phi_j) &= \int \varphi_{ij}(l_i, l_j)\phi_{ij}(\psi_i, \psi_j) \\ &\quad \sum_{v_i} P_i(I_1....I_M|\Phi_i, \mathbf{v}_i(l_i))P(\mathbf{v}_i(l_i)) \\ &\quad \prod_{k\in E_1\backslash j} m^{r-1}_{ki}(\Phi_i) \prod_{o\in E_2\backslash j} m^{r-1}_{oi}(\Phi_i) dl_i \end{aligned}$$

where $\mathbf{v}_i = (v^1_i, ..., v^M_i)$. Note that $\varphi_{ij}(l_i, l_j) = 1$ for messages along edges in E_2 and $\phi_{ij}(\psi_i, \psi_j) = 1$ for messages along edges in E_1. Messages along E_3 alter the visibility variables:

$$\begin{aligned} m^r_{ij}(\mathbf{v}_j) &= \int Occl(l_i) \sum_{v_i} P_i(I_1....I_M|\Phi_i, \mathbf{v}_i(l_i))P(\mathbf{v}_i(l_i)) \\ &\quad \prod_{k\in E_1\backslash j} m^{r-1}_{ki}(\Phi_i) \prod_{o\in E_2\backslash j} m^{r-1}_{oi}(\Phi_i) dl_i \end{aligned}$$

where $Occl(l_i)$ defines the occluding characteristics of part l_i and affects the visibility parameters of part j.

Then, the posterior distribution of a body-part $Pos^r(\Phi_i)$ can be computed as:

$$\begin{aligned} Pos^r(\Phi_i) &\propto \sum_{v_i} P_i(I_1....I_M|\Phi_i, \mathbf{v}_i(l_i))P(\mathbf{v}_i(l_i)) \\ &\quad \prod_{k\in E_1\backslash j} m^r_{ki}(\Phi_i) \prod_{o\in E_2\backslash j} m^r_{oi}(\Phi_i) \qquad (10) \end{aligned}$$

To initialize the system, uniform appearance priors and full visibility of each part are used; that is, it is assumed that all parts are fully visible. At any iteration, the posterior distribution of each part is approximated by a set of particles which are sampled using importance sampling. These particles are used to generate the messages to be passed along appropriate edges to enforce inter-part relationships. Updating the parameters for the different parts in turn, the method eventually leads to a stable parameter estimation after several iterations. The particle-based belief propagation is especially effective since the probability distributions are typically not gaussian in nature, especially in the initial iterations, and hence using any parametric model would lead to a loss of information.

4 Computing Priors and Likelihoods

4.1 Computing Part Visibility

We discuss how to compute $P(v^j_i(l_i)|l_1..l_{i-1}, l_{i+1}, ..l_n)$, which represents the probability of visibility of a random point of the skeleton of part i in view j, given the pdf's of locations of parts $l_1 \dots l_n$. If the exact positions of parts in 3D were known, computing $P(v^j_i(l_i)|l_1..l_{i-1}, l_{i+1}, ..l_n)$ would be straightforward. However, only the posterior distributions of the locations of the parts after the previous iteration are known. To compute the probability, notice that a part is not occluded if and only if it is not occluded by any of the parts, allowing us to utilize an independence relation between the occlusion from different parts. Thus, the probability of visibility of a part i in view j, $P(v^j_i(l_i)|l_1..l_{i-1}, l_{i+1}..l_n)$ represented by Pv^j_i, can be broken down into the product of probability of visibilities from different parts as:

$$\begin{aligned} Pv^j_i &= \prod_{k=1,2..i-1,i+1...n} P(v^j_{ik}(l_i)|l_1..l_{i-1}, l_{i+1}..l_n) \\ &= \prod_{k=1,2..i-1,i+1...n} P(v^j_{ik}(l_i)|l_k) \qquad (11) \end{aligned}$$

The above equation requires us to compute $P(v^j_{ik}(l_i)|l_k)$, which represents the probability that part i is not occluded by part k.

To compute this probability efficiently, "occlusion maps" are introduced. An occlusion map of a part k, $O^j_k(x, y, z)$, stores the probability that a 3D point (x, y, z) will be occluded by part k in view j (Figure 2 illustrates an occlusion map of a sphere). The occlusion map of a body part depends on the shape and location of the part. It has to be updated at every iteration because the probability distribution of locations change at each iteration. For updating the occlusion map of part k, the region of occlusion[1] for each particle of k is computed. The update is made using the following equation:

$$O^{r+1,j}_k(x, y, z) = \frac{n_{occ}}{n} \qquad (12)$$

where r is the iteration number, n_{occ} is the number of particles that support the fact that a point (x, y, z) will be occluded by part k in view j, and n is the total number of particles used for computing the message. Intuitively, the probability that a 3D point (x, y, z) is occluded by part k is proportional to the number of particles of part k that occlude the point.

To provide smoother updates to the occlusion maps and handle errors in approximating the probability calculations, it is useful to update the occlusion maps incrementally:

$$O^{r+1,j}_k(x, y, z) = (1 - \beta)O^{r,j}_k(x, y, z) + \beta(\frac{n_{occ}}{n}) \quad (13)$$

where β determines the rate of change of the occlusion maps ($\beta = 0.2$ was used in our experiments).

[1] The region of occlusion is the 3D region that will be occluded by the part

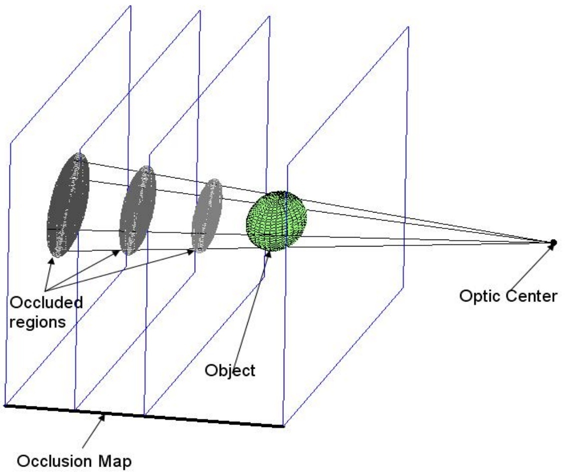


Figure 2. The occlusion map created by a sphere. The cone behind the sphere is the region of occlusion in 3D. The probability of visibility is decreased for every 3D point lying within the cone.

Using the occlusion map of part k for view j, the probability of visibility of a point object i at location, $l_i = (x, y, z)$ in view j, can be computed as:

$$P(v_{ik}^j(l_i)|l_k) = 1 - O_k^j(x, y, z) \qquad (14)$$

In order to address the finite size of the part, $P(v_i^j(l_i)|l_k)$ is approximated by averaging the visibility probabilities along the part skeleton. Computation of occlusion maps is inexpensive since their computational complexity is linear in the number of particles, typically just a few hundred.

4.2 Part Appearance

The appearance of a part is modeled by computing its color as a function of height. The appearance of a part can be represented by a vector that contains n_1 different color vectors along the part. The euclidean distance is used to compute the distance between two appearance vectors.

4.3 Image Likelihoods

Each body part is modeled as a cylinder. Under orthographic projection, the image of a cylinder will consist of parallel lines for two occluding contours of the part, and two circular surfaces at the joints which are normally not detectable. The response of a filter shown in Figure 3 is used to find such parallel lines. The filter gives high response for parallel lines separated by distance w and is robust to moderate deviation from the parallel line assumption.

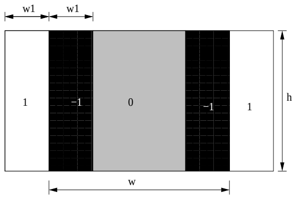


Figure 3. The filter used for finding image likelihoods for parallel lines. w represents the projected width of the body part and h represents the height of the part. The white, black and grey portions have weights 1,-1 and 0 respectively.

An exponential dependence of the likelihood on the filter response is employed, so the likelihood of the image given that the object-part is visible from the camera is:

$$P_i(I_j|l_i, v_i^j(l_i) = TRUE) \propto e^{(1-resp(l_i^j))} \qquad (15)$$

where l_i^j is the location where part i will be projected in image j. More complicated models and filters can also be used[15]. Computation of $P_i(I_j|l_i, v_i^j(l_i) = FALSE)$ represents the case when the part is occluded. It can also be treated as computing the likelihood of observing a random pattern at location l_i^j with no preference given to one pattern over another [2].

Therefore, the likelihood can be assigned a fixed constant in this case.

5 System Overview

Our method requires the computation of the posterior distribution for each part in the graphical model. The computation of such a distribution, however, can be prohibitively expensive since it requires search over a large configuration space. We use prior based search in our case to make the system efficient. Here, search is done only in regions which are allowed according to joint motions.

One can also use independent part detectors to initialize the search process. The most discriminative of these parts is perhaps the face, which may be detected using a face detector (we use a popular one based on [20]). We apply epipolar constraints and matching across views in order to obtain a rough localization of faces in 3D, which are used to initiate search in certain high probability regions.

The cameras are placed in a wide-baseline configuration to reduce occlusions. The system is able to find parts even if they are visible in only one view and yields a good probability distribution of part location even when the part is not

[2] although this is not entirely true since the observation is correlated to the appearance of the part that occludes this part.

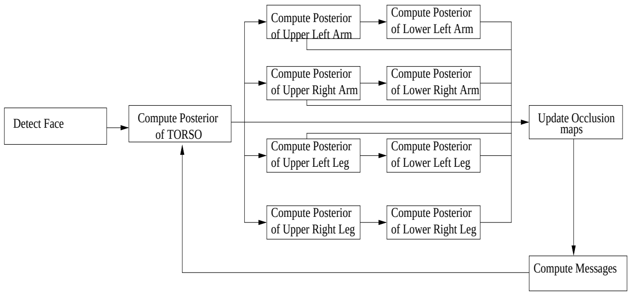
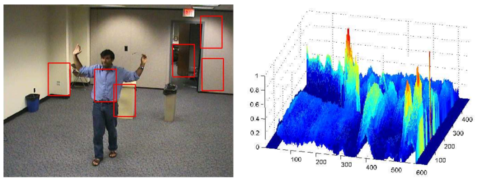

Detect Face
Compute Posterior of TORSO
Compute Posterior of Upper Left Arm
Compute Posterior of Lower Left Arm
Compute Posterior of Upper Right Arm
Compute Posterior of Lower Right Arm
Compute Posterior of Upper Left Leg
Compute Posterior of Lower Left Leg
Compute Posterior of Upper Right Leg
Compute Posterior of Lower Right Leg
Update Occlusion maps
Compute Messages

Figure 4. System Flowchart



(a) The Image with likelihood peaks marked

(b) The Image Likelihood

Figure 5. The parallel line feature is very weak as too many parallel lines occur in nature.

visible in other views. This is due to the inclusion of visibility constraints in the likelihood calculations.

The system flow is shown in Figure 4. Potential faces are first detected using the face-detector. Then, at each iteration, the first step is to find the torso and then search for the other connected parts in turn. Once the posterior distribution of all the parts is estimated at the end of an iteration, messages are passed that update the visibility variables and apply the appearance constraints across parts.

6 Experimental Results and Evaluation

Anthropometric data was acquired using hand-labeled images. This anthropometric data includes ratios of heights and widths of different body-parts and is used for pruning the search area. The angular constraints used on body parts were based on the possible movement of the parts. For example, the maximum possible motion between upper arm and lower arm was kept at 150 degrees (assuming the same volume in 3d cannot be occupied by 2 parts). The constraints were relaxed to reduce the number of missed parts.

We tested the effectiveness of our likelihood model when the parts are visible. Figure 5 shows the computed likelihood of the torso in the image. While the *x,y axes* show the location in the image, the *z-axis* correspond to the likelihood of a torso being present there. The result demonstrates the need to prune the search areas based on priors as the likelihood has a local maxima at several places in the image.

Several experiments were performed to demonstrate the importance of the constraints incorporated in our system. All the experiments were performed using two-views. The importance of modeling occlusion is demonstrated in Figure 6. In this example, the right upper arm is occluded in view 2 and the right leg is occluded in view 1. Figure 6(a) shows the results of the approach taken in [18] with our likelihood model. The right leg is missed by the algorithm because the posterior peaks at some other location with higher likelihood. Figures 6 (b) and (c) show the results of the algorithm with all the constraints and occlusion reasoning. When occlusion information is passed between the body-parts, the left leg creates a region of occlusion which causes an increase in the likelihood of the right leg being present at its actual location.

In another experiment, the algorithm was tested without using appearance constraints while occlusion information and kinematic constraints were used. It can be seen from Figure 7(a) that the lower right arm was missed due to conflicting likelihoods. However, when the appearance constraints are added, correct detection of the lower left arm guides the search for the lower right arm as the appearance of the two are expected to be similar [Figure 7(b))].

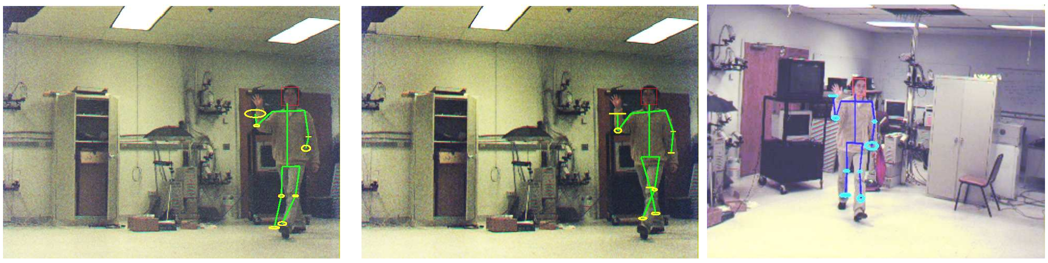
We tested our algorithm on several poses having considerable self-occlusion. The algorithm uses likelihoods from two images to successfully reason about occlusion and localize the limbs in both the views. The performance of our algorithm has been shown in figure 8

The algorithm was also evaluated when multiple people are present and very close to each other. In such cases, it would be very difficult to first segment one person from the image. Figure 9 show the performance of the algorithm in such cases. Figure 10 is a frame from a commonly used sequence from Brown University [18] .

To evaluate the performance of our algorithm, RMSE errors of various joints were computed using 24 hand labeled test images. The error was averaged over various views. Table 1 shows the average RMSE error (in pixels) along with standard deviations. The average height of the person in these images was 320 pixels.

7 Conclusion

We describe an algorithm for estimating the 3D pose of humans. Probabilistic distribution of various parts are used to compute region of occlusions and compute the probability of visibility of each object part given its location. Unlike

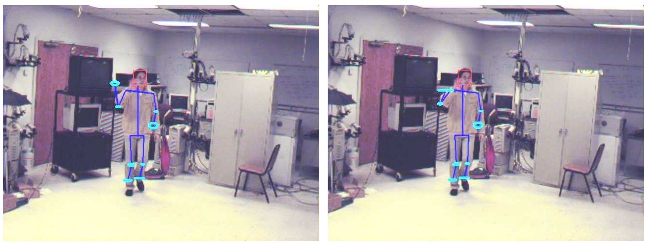


(a) Without Occlusion Constraints View 1 (b) With Occlusion Constraints View 1 (c)With Occlusion Constraints View 2

Figure 6. Illustration of the advantage of using occlusion constraints. Note that the right leg is missed if occlusion constraints are not used.

	L. Elbow	L. Wrist	R. Elbow	R. Wrist	L. Knee	L. Ankle	R. Knee	R. Ankle
RMSE	10.53	15.38	8.84	10.23	8.77	13.38	7.41	13.74
Std. Dev	5.26	6.25	4.53	4.13	4.05	7.83	3.97	7.98

Table 1. Average RMSE(in pixels) of joints projected from 3D pose in different views.



(a) (b)

Figure 7. (a) The lower right hand is missed when appearance constraints are not used. b) Appearance consistency with the other hand helps in peaking the posterior at correct location.

previous approaches, which either assume that all parts are fully visible or compute visibility of the part using optimization we use multi-view geometry to find likelihoods and visibility. This not only makes the NBP more efficient but also allows us to handle unconventional poses which have high degree of self-occlusion. We also consider the high correlation between the appearance of left-right part pairs and use it to better localize the part locations. Experimental results demonstrate the importance and effectiveness of incorporating these additional constraints in real scenes with multiple people.

Acknowledgement

This research was supported by Homeland Security Advanced Research Project Agency Award N0001405C0218.

References

[1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *CVPR*, volume 2, pages 882–888, 2004.

[2] K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *CVPR*, volume 1, pages 77–84, 2003.

[3] Q. Delamarre and O. Faugeras. 3d articulated models and multi-view tracking with silhouettes. In *ICCV*, pages 716–721, 1999.

[4] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, volume 2, pages 66–73, 2000.

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

[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, 2005.

[7] S. Ioffe and D. A. Forsyth. Human tracking with mixtures of trees. In *ICCV*, pages 690–695, 2001.

[8] M. Isard. Pampas: Real-valued graphical models for computer vision. In *CVPR*, volume 1, pages 613–620, 2003.

[9] X. Lan and D. P. Huttenlocher. Beyond trees: Common-factor models for 2d human pose recovery. In *ICCV*, volume 1, pages 470–477, 2005.

[10] M. W. Lee and I. Cohen. Proposal maps driven mcmc for estimating human body pose in static images. In *CVPR*, volume 2, pages 334–341, 2004.

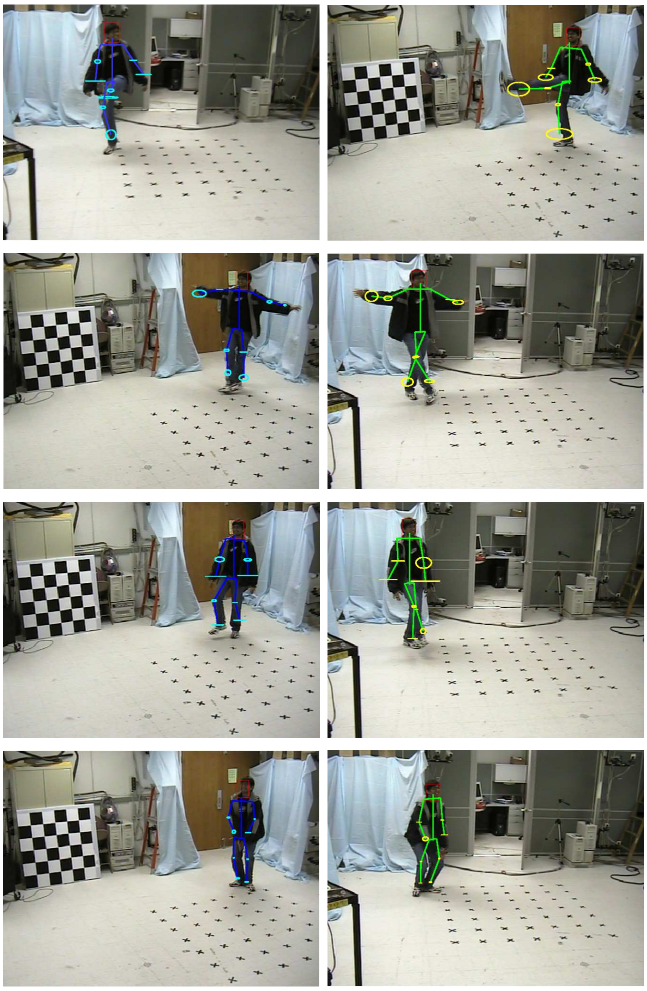


Figure 8. Results of our algorithm on different type of poses.

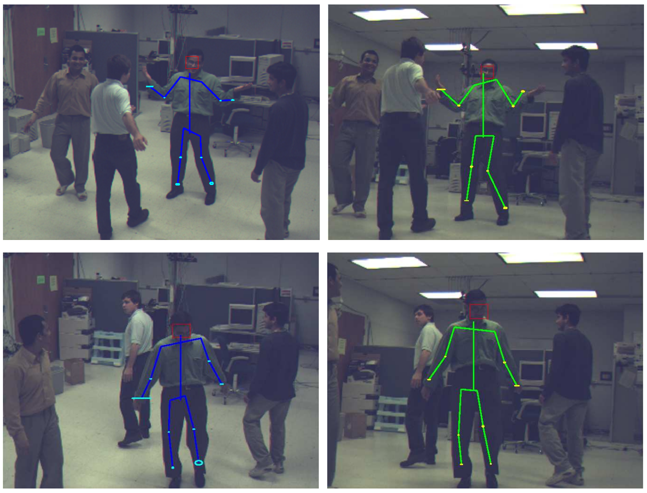


Figure 9. Results of our algorithm when Multiple People are present.

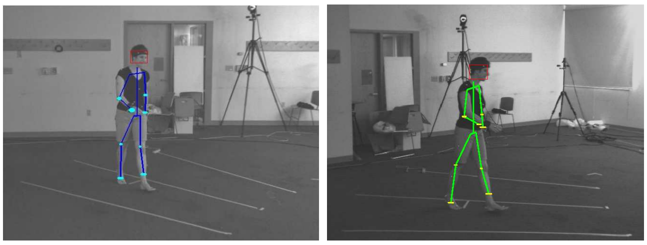


Figure 10. Results from a commonly used sequence from Brown University.

[11] A. Mittal, L. Zhao, and L. Davis. Human body pose estimation by shape analysis of silhouettes. In *AVSS*, pages 263–270, 2003.

[12] G. Mori, X. Ren, A. A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *CVPR*, volume 2, pages 326–333, 2004.

[13] D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *CVPR*, volume 1, pages 271–278, 2005.

[14] X. Ren, A. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *ICCV*, volume 1, pages 824–831, 2005.

[15] S. Roth, L. Sigal, and M. J. Black. Gibbs likelihoods for bayesian tracking. In *CVPR*, volume 1, pages 886–893, 2004.

[16] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *ICCV*, pages 750–757, 2003.

[17] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV*, volume 2, pages 702–718, 2000.

[18] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard. Tracking loose-linbed people. In *CVPR*, volume 1, pages 421–428, 2004.

[19] E. Sudderth, M. Mandel, W. Freeman, and A. Willsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, 2004.

[20] P. Viola and M. Jones. Rapid object detection using boosted cascade of simple features. In *CVPR*, volume 1, pages 511–518, 2001.