Constraint Integration for Efficient Multiview Pose Estimation with Self-Occlusions

Abhinav Gupta, *Member, IEEE,* Anurag Mittal, *Member, IEEE,*
and Larry S. Davis, *Fellow, IEEE*

A. Gupta and L.S. Davis are with the Department of Computer Science, University of Maryland, College Park

A. Mittal is with the Department of Computer Science and Engg., Indian Institute of Technology, Madras, India

Abstract

Automatic initialization and tracking of human pose is an important task in visual surveillance. We present a part-based approach that incorporates a variety of constraints in a unified framework. These constraints include the kinematic constraints between parts that are physically connected to each other, the occlusion of one part by another and the high correlation between the appearance of certain parts, such as the arms. The location probability distribution of each part is determined by evaluating appropriate likelihood measures. The graphical (non-tree) structure representing the inter-dependencies between parts is utilized to "connect" such part distributions via nonparametric belief propagation. Methods are also developed to perform this optimization efficiently in the large space of pose configurations.

Index Terms

3D/stereo scene analysis, Motion capture, Tracking

I. Introduction

Automatic initialization and tracking of human pose in unconstrained and varying conditions is one of the most challenging problems in visual surveillance because of occlusion, a high dimensional search space and high variability in appearance due to shape and clothing variations. Desirable properties of a human tracker include accuracy, efficiency, ability to self-start, automatic detection of failures and ability to re-initialize [34]. Most early work focused on tracking, where an initialization is given[7], [41]. Recently, there has been an increased interest in automatic detection of body pose to initialize/re-initialize tracking systems[11], [24], [42].

In this paper, we present an efficient multiple camera based approach for estimating the 3D pose of humans in cluttered scenes[1]. The system incorporates a variety of constraints, including the occlusion of one part by another and appearance consistency across parts, in a unified framework.

Most of the current pose estimation systems fail when there is considerable self occlusion because the image likelihoods are low for the occluded parts. In our approach, we boost the likelihoods of the possibly occluded part in proportion to the expected amount of occlusion. We use an iterative approach, where at each iteration we compute a pose likelihood distribution

[1] A preliminary version of this paper appears in 3DPVT'06 [14]

which is used to infer the occluding properties and appearance of each part. The system uses these better appearance and occlusion estimates at each iteration to refine the pose estimates.

Unlike most previous approaches, our method does not require segmented human silhouettes. Our edge based likelihood model allows us to eschew the static background assumption required for background subtraction. The major features of our approach are:

- Occlusion, appearance, kinematic and temporal constraints are incorporated in a unified multi-view framework.
- A computationally efficient approach, as compared to [47], is presented to handle self-occlusions. The occluding properties of a part are utilized to determine the visibility of other parts directly and to give more weight to those views that have a less occluded view of a part.
- A method that combines bottom up and top down approaches to prune the search and make the estimation process efficient is presented. Search is performed only in high prior 3D regions and evidence is collected only once in the image, which is then combined in 3D via epipolar constraints.

The paper is organized as follows. We discuss related work in Section II. Section III discusses the human body model, followed by a discussion of the message passing framework in Section IV. Section V provides a description of visibility analysis and likelihood computations. We then explain how bottom up search is incorporated in a top down framework in Section VI. This is followed by a system overview in Section VII. Experimental results are presented in Section VIII. Finally, we conclude by a description of how to extend the framework to include temporal constraints in Section IX.

II. Related Work

There is a wide range of approaches to human pose estimation [12], [30]. These algorithms can be broadly divided into two categories:

- **Bottom-Up:** Here, possible parts are first found using part detectors and then are combined to form the whole body [19], [31], [37], [34], [27].
- **Top Down:** These algorithms use an explicit 3D human model, along with the kinematic structure and other constraints, to reconstruct the pose[22], [25], [8]. The probability distribution of the whole body configuration is then searched for through techniques such

as Monte Carlo Markov Chain (MCMC). Possible parts are then found by sampling the posterior obtained for each part.

Bottom-up part-based approaches estimate human body parts by combining image evidence with constraints on joint locations. Most prior work uses only kinematic constraints on body part locations. These constraints limit the body part positions by requiring some body parts to be close to others. This requirement leads to a tree-structured graph that can be modeled either in 2D [10], [35] or 3D [44]. Felzenszwalb et. al. [11] presented a deterministic linear time algorithm using dynamic programming to solve for the best pose configuration in such tree structures. Top-down approaches, on the other hand, try to search in the high dimensional space of whole body configurations. Lee et. all [24] combined a probabilistic proposal map representing pose likelihoods with a 3D model to recover the 3D pose from a single image. Data driven Markov chain Monte Carlo [49] is used to search in the high dimensional space of possible poses. Other approaches include Data Driven Belief Propagation [17], particle filtering [25] and annealed particle filtering [8].

Most of these methods assume a tree structure for the constraints to be satisfied. 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 pair of parts such as the hands [32].

There has been some recent work to overcome these limitations. Lan et. al [23] use factor graphs to add constraints such as the balance of a body while walking; Ren et. al [36] use Integer Quadratic Programming (IQP) to add pairwise constraints such as the similarity in the appearance of left and right body-parts. Sigal et. al [42] present an approach to detect and track humans from multiple views. Kinematic constraints combined with temporal constraints lead to the formation of a loopy graph which can be optimized using Non Parametric Belief Propagation(NBP). However, they do not explicitly model self-occlusion where one part occludes another as shown in Figure 1.

Ioffe et. al [20] proposed using a mixture of trees to handle such 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 difficult and does not provide strong constraints, leading to false part localizations. For example, the

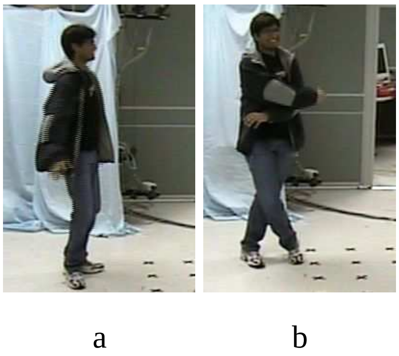


Fig. 1. Self occlusion is a problem in likelihood computation. It leads to low likelihood at the true location when one part occludes another. (a) Right leg occludes the left leg and the torso occludes the left hand. (b) Both hands occlude the torso partially.

position of the torso provides weak constraints on the possible positions of the lower arms in scenarios where the upper arms might be occluded. 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 [47] handle a different but related problem of tracking a human hand under self occlusion using NBP. They use only a single camera for tracking the hand in 3D. In order to handle occlusion, they augment the state of each particle by a set of binary hidden variables that represent the set of occluded pixels in the projection of the part. The non-tree structure obtained is then optimized in a non-parametric belief propagation framework. However, the introduction of such variables increases the problem state space exponentially and the resulting optimization problem can be quite unstable, especially in the presence of ambiguity in the part likelihoods. Furthermore, the technique does not generalize well to multiple views since the occlusion state of part pixels is view specific, and extension to multiple views would require introduction of a very large number of extraneous hidden variables. We use a similar but more tractable approach of determining the probability of visibility/occlusion directly from the probability distributions of the locations of other parts and use it to improve the estimation of the part likelihoods. The resulting problem can again be solved using non-parametric belief propagation. In work parallel to ours, Sigal et. al [43] use a formulation similar to the above[47] for handling self-occlusions, but with a different likelihood model, and apply it to 2D pose estimation.

The pose estimation problem can be simplified significantly by assuming that the person can be segmented from the image, say using background subtraction [4], [23], [29]. While this reduces the search space significantly, it does not handle the problem of self-occlusion or people occluding one another.

A complementary approach, commonly known as discriminative methods [1], [3], [9], [39], [13], [15], [40], [45], is to learn pose configurations from training images and use appearance-based associative models to ”look up” the pose from the training data. These methods use a parametric model of posterior probabilities and learn the parameters using the training data. On the other hand, generative approaches like ours model the joint probability distribution using class conditional densities and class prior probabilities. Compared to discriminative approaches, generative approaches have the following advantages:

- Generative models generalize well, whereas discriminative models depend heavily on the learned poses. Due to the large space of pose configurations, it is very difficult to identify new poses.
- Generative models can handle compositionality (e.g people with extra clothings like hats, or people with bags) whereas discriminative approaches need to see all possibilities in the training dataset.

On the other hand, discriminative approaches offer the following advantages:

- Discriminative models are generally faster due to the lower dimensionality of the models employed.
- Discriminative models generally provide better predictive performance when the training set is large and comprehensive.

Another approach [4], [26] for estimating the 3D human pose from multiple cameras is based on segmenting the visual hull based on prior knowledge of the shapes of the body parts, their relative sizes and possible configurations. While volume intersection methods like these produce accurate results, they can only be used for studio-like applications since they require static backgrounds and are too sensitive to background subtraction errors. Apart from this, occlusion and self-occlusion is a major problem in such applications, especially if the number of cameras is not very large.

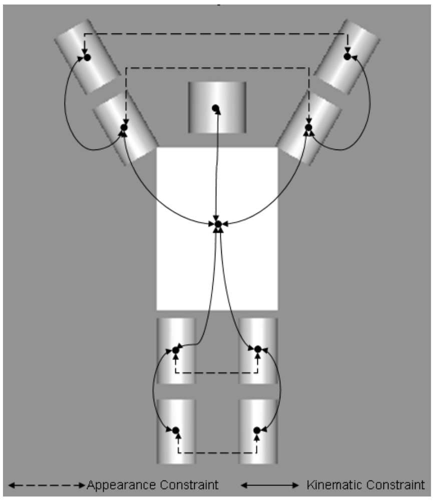


Fig. 2. The human model. The solid lines represent edges in set E_k and the dashed lines represent edges in the set E_a. Occlusion edges are not shown in the above graph. Every part is connected to all other parts by occlusion constraint edge.

III. Modeling the Human Body and Problem Formulation

Our 3D human body model (Figure 2) 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 with a random vector $\Phi_i = (l_i, a_i)$, where l_i and a_i represent the location and appearance parameters of part i respectively. The location parameters of each part, l_i, is further parameterized as $l_i = (l_i^s, l_i^e)$ where l_i^s and l_i^e are the 3D positions of the two ending points of the limb.

The nodes of the graph are connected by three types of edges. The first enforces kinematic constraints between parts. 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 $\theta = (E_k, E_a, E_o, c_k, c_a, c_o)$, where the set of edges E_k, E_a and E_o indicates which parts are connected by edges of the first, second and third type respectively; c_k, c_a and c_o 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 $\Phi \equiv (\Phi_1, \Phi_2........\Phi_n)$. In an M camera setup, if I_j denotes the image from the j^{th} camera, then $P(I_1....I_M|\Phi)$ is the likelihood of observing the set of images given the 3D locations and appearances of the body parts. The distribution of $P(\Phi)$ is the prior over the possible body configurations. The goal is to find the posterior distribution, $P(\Phi|I_1....I_M)$, which measures the probability of a particular configuration of the human body given M views and the object model. Using Bayes' rule,

$$P(\Phi|I_1....I_M) \propto P(I_1....I_M|\Phi)P(\Phi) \tag{1}$$

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

$$P(\Phi) = P(l_1.....l_n)P(a_1.....a_n) \tag{2}$$

As any particular location or orientation of a part is not prefered over another, we neglect priors of single part locations. Furthermore, we use potential functions to avoid normalization computations. Then, the joint distribution of the tree structured prior E_k and E_a can be written as:

$$P(l_1, l_2......l_n) \propto \prod_{(v_i,v_j)\in E_k} \kappa_{ij}(l_i, l_j) \tag{3}$$

$$P(a_1, a_2....., a_n) \propto \prod_{(v_i,v_j)\in E_a} \alpha_{ij}(a_i, a_j) \tag{4}$$

where κ_{ij} and α_{ij} are the potential functions for kinematic and appearance constraints over the cliques(pair of nodes in this case).

For articulated objects, pairs of parts are connected by flexible joints. Ideally, the location of the ending-point of the first part should be the same as the starting point of the second connected part. The clique potential for a pair of parts, connected by edges in E_k (kinematic connections) is modeled as a Gaussian:

$$\kappa_{ij}(l_i, l_j) = \mathcal{N}(d(l_i, l_j), 0, \sigma^{\kappa}_{ij}) \tag{5}$$

where $d(l_i, l_j)$ denotes the Euclidean distance between the connecting end points of the i^{th} and j^{th} body parts.(Figure 3).

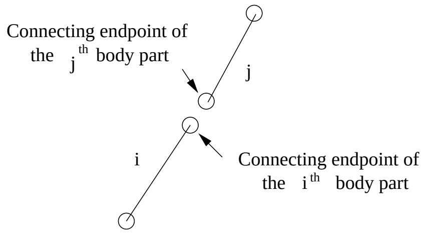

Connecting endpoint of the j th body part
j
i
Connecting endpoint of the i th body part

Fig. 3. The connecting end-points of the two connected parts.

For appearance constraints, let $D(a_i, a_j)$ 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, α_{ij}, is modeled as:

$$\alpha_{ij}(a_i, a_j) = \mathcal{N}(D(a_i, a_j), 0, \sigma_{ij}^{\alpha}) \tag{6}$$

Section V-B discusses how part appearances are modeled and how the distance, $D(a_i, a_j)$, is computed.

Computation of the likelihood $P(I_1....I_M|\Phi)$ is complicated due to occlusion. Sudderth et. al [47] introduce hidden variables to represent the occlusion mask and use only unoccluded pixels for likelihood computation. This process increases the size of the solution space exponentially. Instead, we compute the probability of visibility of each part in different views using the probability distribution of all other parts and use it to compute the likelihood over all the views as explained below.

The imaging from every camera is modeled as a conditionally independent process. Similarly, the observation of different parts is assumed to be conditionally independent. This allows us to decompose the image likelihood for the configuration Φ as:

$$P(I_1....I_M|\Phi) \propto \prod_{i=1}^{n} \prod_{j=1}^{M} P_i(I_j|l_1...l_n, a_i) \tag{7}$$

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 introduce a new set of binary 'visibility' variables $v_i^j(l_i)$, that refer to the visibility of a part i at location l_i from camera j. While these visibility variables are dependent upon the position of all other parts, the observation likelihood for part i is independent of the location of other parts if its visibility is known. Then, one can write the likelihood, $P(I_1....I_M|\Phi)$, as:

$$\prod_{i=1}^{n}\prod_{j=1}^{M}\sum_{v_i^j\in\{T,F\}} P_i(I_j|l_i, v_i^j(l_i))P(v_i^j(l_i)|l_1....l_{i-1}, l_{i+1}....l_n) \tag{8}$$

The term $P_i(I_j|l_i, v_i^j(l_i) = TRUE)$ represents the likelihood of observing the image from camera j given that the part is visible from this camera while $P_i(I_j|l_i, v_i^j(l_i) = FALSE)$ 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 $v_i^j(l_i)$ is neither true nor false. To approximate this, $v_i^j(l_i)$ is defined as the visibility of a random point on the skeleton of the part. In Section V-A, we discuss how to compute the visibility variables and in section V-C, we discuss how to compute the likelihoods.

IV. 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 with loops, a belief propagation framework can be used([33], [52]). Essentially, we optimize for the posterior of each part; the interactions between different parts are handled via belief messages. Since representing exact probability distributions is computationally and memory intensive, we use the non-parametric belief propagation framework presented in [21], [46] where the probability distributions of the part locations and appearances are represented via Monte-Carlo particles. The framework provides a natural approach for enforcing constraints across parts, including those of occlusion and appearance matching.

There are, essentially, two sets of unknowns that need to be estimated simultaneously: the locations and the appearances. The computation of the posterior distribution at a particular node requires locations, appearances and occluding properties (represented via occlusion-maps) of other connected nodes in the graph. The following messages are used to pass this 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).

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

A message from part i to part j imposes constraints on the configuration of part j for possible configurations of part i. The contribution of any configuration of part i is weighted by its posterior. At iteration r, a message m_{ij} from node i to j along an edge in E_k or E_a may be represented as:

$$m_{ij}^r(\Phi_j) = \int \kappa_{ij}(l_i, l_j)\alpha_{ij}(a_i, a_j)Pos^{r-1}(\Phi_i)d\Phi_i \tag{9}$$

where $Pos^{r-1}(\Phi_i)$ represents the posterior distribution of part i at iteration $r-1$. Note that $\kappa_{ij}(l_i, l_j) = 1$ for messages along edges in E_a and $\alpha_{ij}(a_i, a_j) = 1$ for messages along edges in E_k. The posterior distribution of a body-part $Pos^r(\Phi_i)$ can be computed as:

$$Pos^r(\Phi_i) \propto \sum_{v_i} P_i(I_1....I_M|\Phi_i, \mathbf{v^r}_i(l_i))P(\mathbf{v^r}_i(l_i)) \prod_{k \in E_k \setminus j} m_{ki}^r(\Phi_i) \prod_{o \in E_a \setminus j} m_{oi}^r(\Phi_i) \tag{10}$$

where $\mathbf{v^r}_i = (v_i^{r,1}, ..., v_i^{r,M})$ represents the visibility maps of part i at iteration r in all the cameras. The visibility maps are computed by combining the probabilistic occlusion maps which are passed as messages along the edges in E_o. Section V-A discusses how to compute visibility maps from the probabilistic occlusion maps.

To initialize the system, uniform appearance priors and full visibility. 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 different parts in turn, the method eventually leads to stable parameter estimation after several iterations. The particle-based belief propagation is especially effective since the probability distributions are typically not gaussian in nature and hence using any parametric model would lead to a loss of accuracy.

The study of convergence properties of belief propagation systems is an active area of research [53], [54], [48], [51], [18]. When the graph is singly connected, belief propagation systems are guaranteed to converge to the correct posterior probabilities. However, in the case of graphs with loops, the convergence behavior is more complicated. Weiss et. al [53] studied the convergence properties of belief propagation with single loops. Recently, some papers [48] have derived sufficient (although not necessary) conditions that guarantee the convergence of loopy belief propagation systems. Especially when complicated continuous distributions are approximated through parametric (Gaussian belief propagation) and non-parametric techniques (NBP [46], [21]) , the convergence properties are poorly understood. Many recent papers [42], [46], [21] have empirically demonstrated good performance of the NBP algorithm on graphs with loops.

V. Computing Priors and Likelihoods

A. *Computing Part Visibility*

We discuss how to compute $P(v_i^j(l_i)|l_1..l_{i-1}, l_{i+1}, ..l_n)$, which represents the probability of visibility of a random point on the skeleton of part i in view j, given the pdf's of the locations of parts $l_1 \dots l_n$. If the exact positions of parts in 3D were known, computing $P(v_i^j(l_i)|l_1..l_{i-1}, l_{i+1}, ..l_n)$ 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 part, 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, $P(v_i^j(l_i)|l_1..l_{i-1}, l_{i+1}..l_n)$ represented by Pv_i^j, can be broken down into the product of the probability of visibilities from different parts:

$$\begin{aligned} Pv_i^j &= \prod_{k=1,2..i-1,i+1...n} P(v_{ik}^j(l_i)|l_1..l_{i-1}, l_{i+1}..l_n) && (11) \\ &= \prod_{k=1,2..i-1,i+1...n} P(v_{ik}^j(l_i)|l_k) && (12) \end{aligned}$$

The above equation requires computing $P(v_{ik}^j(l_i)|l_k)$, 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, $O_k^j(x, y, z)$, denotes the probability that a 3D point (x, y, z) will be occluded by part k in view j (Figure 4 illustrates an occlusion map of a sphere). The occlusion map of a body part depends on its shape and location. The occlusion maps are updated at every iteration because the probability distribution of the part locations change at each iteration. For computing the occlusion map of part k, the region of occlusion[2] for each particle of k is computed. The occlusion map is defined by the following equation:

$$O_k^{r+1,j}(x,y,z) = \frac{n_{occ}}{n} \tag{13}$$

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 that point.

To provide smooth updates to the occlusion maps, it is useful to update the occlusion maps incrementally:

$$O_k^{r+1,j}(x,y,z) = (1-\beta)O_k^{r,j}(x,y,z) + \beta(\frac{n_{occ}}{n}) \tag{14}$$

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

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

$$P(v_{ik}^j(p_i)|l_k) = 1 - O_k^j(x,y,z) \tag{15}$$

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 linear in the number of particles, typically just a few hundred.

B. Part Appearance

The appearance of a part is modeled by its color distribution along the length of the part. While a single color model for the whole part would not be able to capture the color variations, modeling the appearance using a histogram will be computationally expensive. A part is divided into regions along its length, and a single color model is developed for each such region (See

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

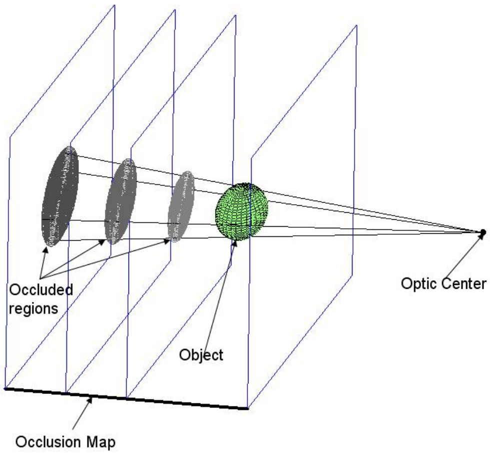


Fig. 4. The occlusion map created by a sphere. The cone behind the sphere is the region of occlusion in 3D. The probability of visibility is decreased for every 3D point lying within the cone.

Fig 5). For each part hypothesis, a few pixel are sampled along the skeleton of the part to capture the color variations along the length. To handle occlusions, we also associate a confidence variable with each region, which represents the certainity in our estimate of appearance. In our experiments, we assume certainity is proportional to the probability of visibility of the region. To reduce the effects of illumination changes, we use normalized color (i.e the ratios $\frac{r}{r+g+b}$ and $\frac{g}{r+g+b}$) instead of RGB color components.

The distance/difference between the appearance of two parts is computed using the weighted Euclidean distance. If a part is divided into r regions, (a_{ik}^1, a_{ik}^2) respresent the normalized color components of region k of part i and c_{ik} represents the confidence in above estimate, the difference in appearance of the two parts is given by

$$D(a_i, a_j) = \frac{\sum_{k=1}^{r} c_{ik} c_{jk} \sqrt{(a_{ik}^1 - a_{jk}^1)^2 + (a_{ik}^2 - a_{jk}^2)^2}}{\sum_{k=1}^{r} c_{ik} c_{jk}} \tag{16}$$

C. Image Likelihoods

Each body part is modeled as a cylinder. Under orthographic projection, the image of a cylinder will consist of parallel lines for its two occluding contours, and two circular surfaces at the joints, which are normally not detectable via image analysis. The response of a filter shown

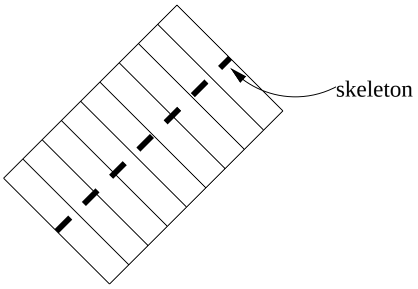


Fig. 5. The division of a part along the length.

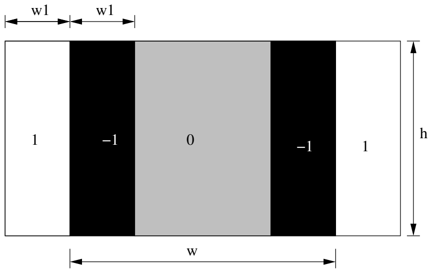


Fig. 6. The filter used for finding image likelihoods for vertical parallel lines. w represents the projected width of the body part and h represents the height of the part. The white, black and grey portions have weights 1,-1 and 0 respectively.

in Figure 6 is used to measure the likelihood of 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.

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

$$P_i(I_j|l_i, v_i^j(l_i) = TRUE) \propto e^{(1-\rho(l_i^j))} \tag{17}$$

where l_i^j is the location where part i projects in image j, and ρ is the response of the filter at a particular location. More complicated models and filters can also be used[38]. 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 [3]. Therefore, the likelihood can be assigned a fixed constant

[3] although this is not entirely true since the observation is correlated to the appearance of the occluding part.

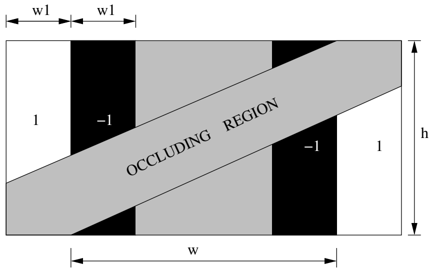


Fig. 7. An example of complicated filters which can be used for likelihood computation. Such filters compute the response for visible portions of part only. In this example, the middle portion(horizontally) of the part is occluded and hence a weight of 0 is used for that portion.

in this case.

Likelihood can be better computed using more complicated filters as shown in Figure 7. In such a filter, the response is calculated only for the visible portions of the part(each pixel weighted by its probability of being visible). The response of such a filter has to be computed pixel-wise as opposed to the integral images formulation [50] used in our system. The approximation used in our experiments is a trade-off made for efficiency, yielding an algorithm which has complexity $O(n_p) + O(m_c)$ instead of $O(m_c n_p)$ where m_c is the number of possible part configurations and n_p is the number of pixels.

It should also be observed that the 2D likelihood model favors part configurations(length and orientation) which project onto smaller image regions (for example, the likelihood of a limb pointing forward is generally higher than the likelihood of a limb visible in full length). However, this is generally not observed in our estimates since in a wide baseline stereo any limb which projects onto a small region in one camera (due to its orientation) generally projects onto a bigger region in the other camera, and the 3D likelihood of such a configuration is low if there is absence of support in any camera. The configurations with smaller 3D limb lengths are rejected because they do not satisfy anthropometric constraints.

VI. Combining Bottom-Up Evidence with Top-Down Priors for Efficient Estimation

The computation of the probability distribution for each part can be quite expensive due to the very large space of possible part configurations (location and orientation) ($O(m_c k^d)$ where m_c

is the total number of possible configurations, k is the number of particles retained for message passing and d is the degree of a node). In order to deal with this computational complexity, Coughlan et. al [5] discussed accelerating belief propagation by belief-pruning and focussed message updates. Belief pruning removes states with very low posteriors from consideration during future stages. However, in the case of occluded limbs, the belief will be initially very low and the approach might fail. Sigal et. al [42] use independent part detectors, called shouters, to create the sampling function, which leads to better sampling. Similar sampling approaches for belief propagation have been used in [17].

We propose using 2D evidence from images and combining information from multiple cameras using epipolar geometry to obtain high likelihood bodypart regions in 3D. Additionally, regions with high priors for a given part are obtained using the probability distribution of connecting parts. Since a high probability region must have either a high likelihood or a high prior, the search in the configuration space can then be confined to these two types of regions.

In order to determine regions with high priors, we use the parameters of appropriate connecting parts and anthropometric data. For example, after finding the posterior distribution of the upper arm, one can prune the search area in 3D for the lower arm.

Pruning via priors alone is not sufficient, especially for parts such as the four end limbs. One can further constrain the search space by considering cues from a bottom up search process. The approach is motivated by the fact that multiple 3D part configurations can project onto the same 2D configuration and thus a full search in 3D leads to a large number of repeated likelihood computations in the 2D images. Furthermore, search for high likelihood regions in 3D requires transformations from 3D to 2D which are expensive compared to the 2D likelihood computations themselves. These transformations are not required when the corresponding likelihood is very small for the corresponding 2D locations. Our approach first computes 2D likelihoods and combines only those instances that are above a certain threshold using epipolar geometry to compute high likelihood configurations in 3D.

We first compute the search region for the starting and ending points of each part in each view using the priors from the connected parts. Figure 9(a) shows the search region for the starting point of the lower right leg in cyan color. For each possible limb in one view, there is a set of possible limbs which satisfy epipolar constraints in the other view. Figure 9(b) shows the set of possible limbs in the second image corresponding to the limb in the reference image. Searching

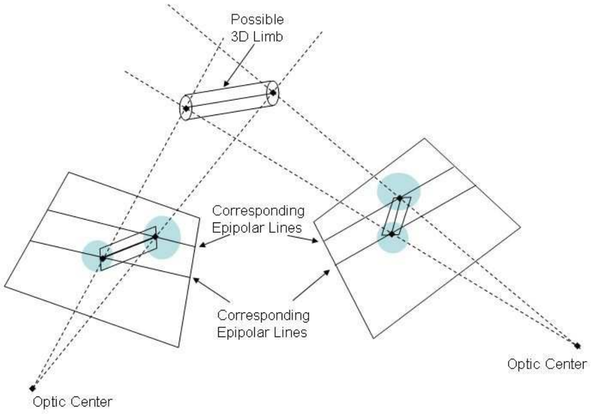


Fig. 8. Search for possible parts in 2D along the epipolar lines is constrained by regions(maked in cyan) where starting and endpoints should lie. Limbs with higher image likelihoods are then back-projected to find possible parts.

along epipolar lines for the starting and ending points, the instances where the 2D likelihood is above threshold in both images are back-projected to compute the 3D position of these high likelihood parts(See Figure 8). Such a pruning procedure is not applicable when the limb is in an occluded region and thus not used in such regions. The threshold is kept low to avoid false negatives and handle partial occlusions.

Pruning by likelihoods is data-dependent but even in an image with a cluttered background, pruning via likelihoods resulted in an additional speed-up of 10x compared to using only pruning via priors.

VII. System Overview

The entire search space is very large. In order to tackle this large search space, the system adopts a hierarchical approach where the crude locations that have a high probability of having a person are found first. This is followed by a belief propagation procedure which finds the pose of the person. To find the crude locations, an independent part detector or a person detector [6] can be used. In our experiments, we use face detector from [50] since the face is the most discriminative body part. 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 for the rest of the body in high probability regions.

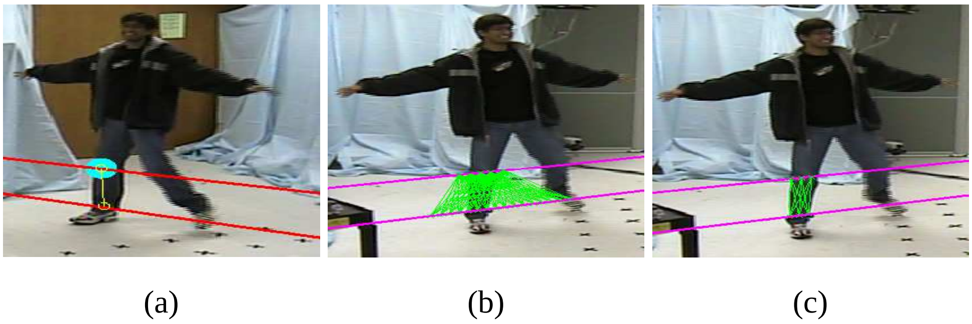


Fig. 9. (a) The cyan colored circular region shows possible start points for the lower right leg. This region is obtained using the belief propagation particles. (b) Search along epipolar lines for a part in the other image. Again the top point is restricted by the search region created from the upper leg. (c) Possible parts after pruning via likelihoods. For a candidate part in the reference image, we have very few parts in the other image that have a high likelihood.

The cameras are placed in a wide-baseline configuration to obtain viewing angles which allows better handling of occlusion. The system is able to find parts even if they are only partially visible in both the views and yields a good probability distribution of the part location even when the part is completely occluded in one of the views. This is due to the inclusion of visibility constraints in the likelihood calculations.

The system flow is shown in Figure 10. Potential faces are first detected using the face-detector. Then at each iteration of belief propagation, we find the torso and then search for the other connected parts, in turn. The two search methods described above are used to search for each part. 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. The process is iterated until there is no change in the part distributions.

Anthropometric data was acquired using the NIST dataset [16]. This data includes ratios of heights and widths of different body parts and is used for pruning the search region for a given part. The angular constraints of the body parts were based on the possible movements of each joint. For example, the maximum possible angular motion between the upper and lower arm was kept at 150 degrees. The constraints were relaxed to reduce the number of missed parts.

VIII. Experimental Results and Evaluation

We performed a series of experiments to evaluate our algorithm, comparing it to the algorithm in [42] that does not use occlusion or appearance consistency constraints. The test dataset was

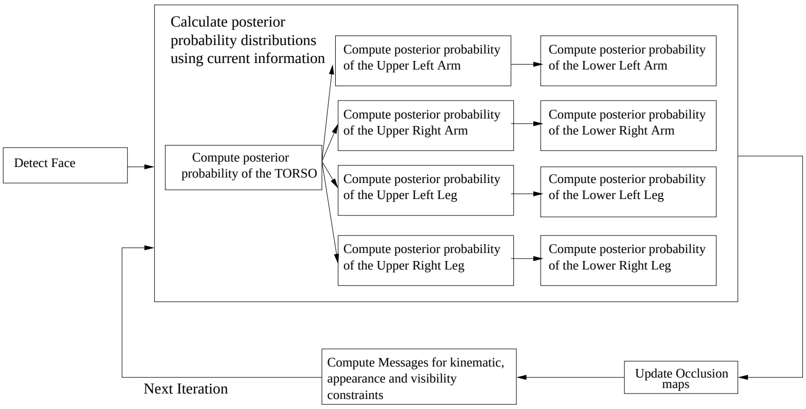

Calculate posterior probability distributions using current information
Detect Face
Compute posterior probability of the TORSO
Compute posterior probability of the Upper Left Arm
Compute posterior probability of the Lower Left Arm
Compute posterior probability of the Upper Right Arm
Compute posterior probability of the Lower Right Arm
Compute posterior probability of the Upper Left Leg
Compute posterior probability of the Lower Left Leg
Compute posterior probability of the Upper Right Leg
Compute posterior probability of the Lower Right Leg
Compute Messages for kinematic, appearance and visibility constraints
Update Occlusion maps
Next Iteration

Fig. 10. System Overview.

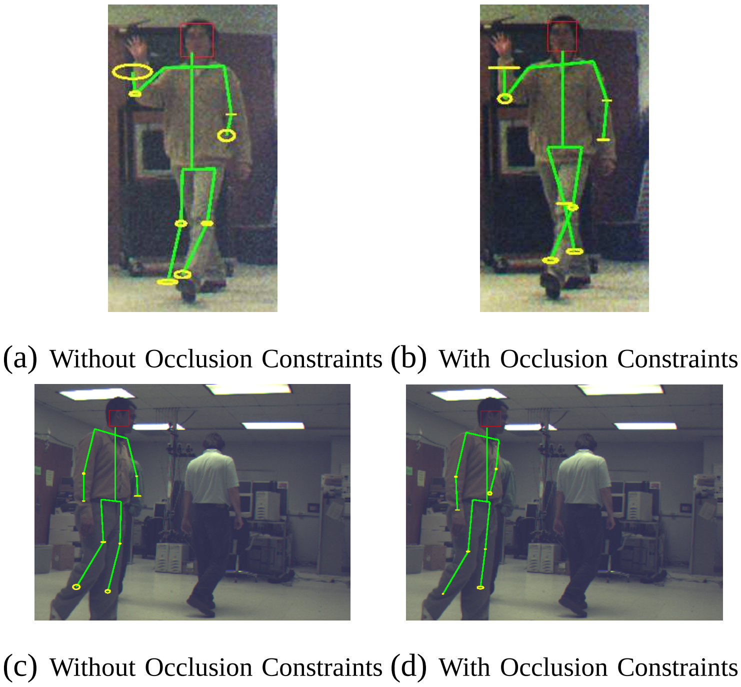
Class	Self-occlusion	Occlusion from others
1	NO	NO
2	YES	NO
3	MAYBE	YES

TABLE I

THREE CLASSES OF DATASETS USED FOR EVALUATION

divided into three classes. The first two classes consist of scenes where only a single person is present. In the first class, there is little or no self-occlusion of body parts; in the second class, one or more body parts are severely occluded by other body parts. The third class consists of multiple people sequences where confusion between the parts of different persons can be significant. Table I lists the properties of each of the three classes.

All experiments were performed using two views. The importance of modeling occlusion is illustrated in Figure 11. Whenever a body part is occluded, the likelihood at the true position will be generally reduced. Thus, without modeling the occlusion, the observed posterior probability will be lower compared to the one where the occlusion has been modeled. Figure 11(a) shows the results of the algorithm without the occlusion constraints (this is equivalent to the algorithm in [42] with our likelihood model) for a scene where the right leg is occluded in one view. The right leg is missed by the algorithm because the posterior peaks at some other location. Figure 11 (b) shows the result of the algorithm in one of the views with all of the constraints



(a) Without Occlusion Constraints (b) With Occlusion Constraints

(c) Without Occlusion Constraints (d) With Occlusion Constraints

Fig. 11. Illustration of the advantage of using occlusion constraints. (a) and (b): The right leg is missed if occlusion constraints are not used. (c) and (d): If occlusion constraints are not used, the hand of the person in back is confused to be the hand of the person whose pose is being estimated. By using occlusion constraints, the likelihoods in the region of occlusion are taken to be unknown.

considered. When occlusion information is passed between body-parts, the left leg creates a region of occlusion which leads to higher likelihood in this region using the evidence from the other view. Figures 11(c) and (d) show another example where the algorithm would fail if occlusion constraints were not used. Here, there is a high likelihood at an incorrect location because the left hand of the person in the back is confused with the left hand of the person whose pose is being estimated. The likelihood at the true location is low since the hand is occluded by the torso. However, if occlusion constraints are used, the hand can be located at the true location because of the evidence from the view where the part is not occluded.

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 12(a) that the lower right arm was missed. However, when the appearance constraints are added, correct detection of the lower left arm guides the search for the lower right arm [Figure 12(b))]. Another example of the importance of using appearance constraints is shown in Figures 12(c) and (d).

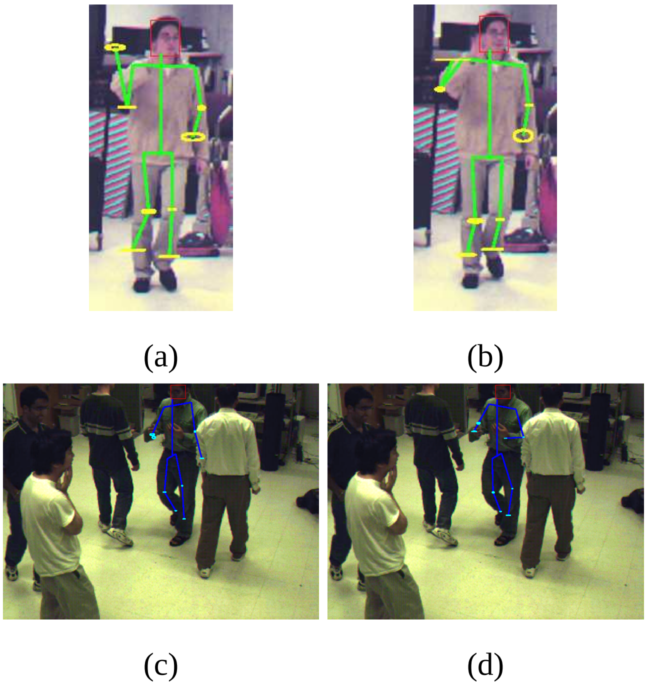


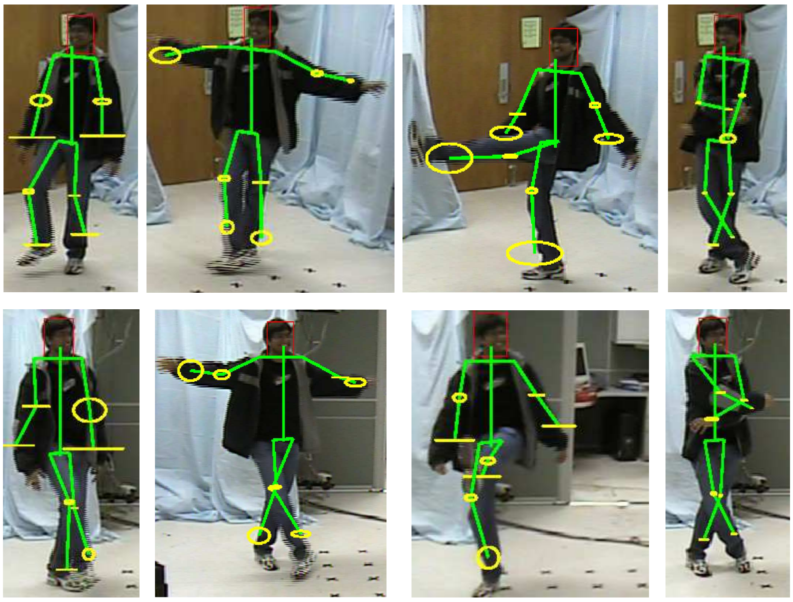
Fig. 12. (a): The lower right hand is missed when appearance constraints are not used. (b) Appearance consistency with the other hand helps in peaking the posterior at the correct location. (c) If appearance constraints are not used, the lower left hand is confused with the lower hand of another person. (d) Appearance of the lower hand correctly guides the search. Results for both the views has been shown in figure 14

When appearance constraints are not used, the lower arm of the person in front is assumed to be the lower arm of the person whose pose is being estimated. When appearance constraints are used, correct detection of the lower right arm guides the search for the lower left arm.

Figure 13 shows the performance of the algorithm on a variety of poses. Most of these poses have significant self-occlusion. Figure 13(d), which has severe amount of self-occlusion shows the limitation of our algorithm when edge information is very weak.

When there are multiple people in the scene(*Class 3 sequences*), it is very difficult to segment one person from another. Figure 14 illustrates the performance of our algorithm in such cases. The algorithm was also tested to estimate the pose of two people using the same image pairs. A separate belief-net was used for estimating the pose of each person. (Figure 15).

The current un-optimized implementation of the algorithm in Visual C++ takes on average 45 second per frame for pose estimation. The running time is large due to the search in the space of possible part configurations.



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

Fig. 13. Results of our Detection algorithm on different poses.

A. Quantitative Comparison

Balan et. al [2] discussed methods to quantitatively compare 3D person tracking algorithms. There are several issues related to the quantitative evaluation of pose estimation algorithms which we discuss next. First, Balan et. al [2] proposed the use of ground truth data of 3D joint locations obtained via markers for any quantitative evaluation. However, such data is available only in laboratory environments and require people to wear special clothes and markers. Apart from such data, we also propose to use hand-labeled position of joints in images for evaluation. We analyze the data in three classes separately. Some of the images from the data-set used are shown in Figure 16.

Secondly, Balan et. al [2] suggest using the average joint error as the full body pose error. It might also be important to determine if some body parts have been missed completely, so we additionally consider the number of body parts missed as a measure of full body pose error.

Furthermore, Bayesian approaches do not estimate a single joint location but a posterior distribution on the joint location. Hence, we have to compute the error of a posterior distribution for each joint. Balan et. al [2] suggest approaches to measure this error ranging from the expected

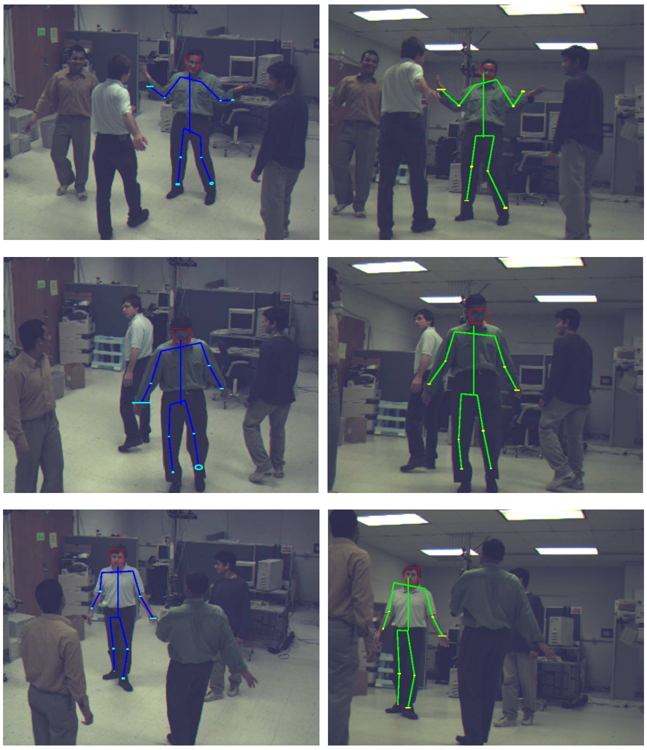


Fig. 14. Results of our Detection algorithm on Class 3 video sequences.

joint location error to the minimum possible error over all samples. However, an important issue to consider is the potential multi-modality of the distributions that are being estimated. Assuming that the posterior distribution of a joint location is multi-modal, we group the sample points into Gaussian clusters and the cluster with minimum possible error is chosen. The error between the true joint location and this cluster of samples is computed using a root-mean-square error (RMSE). If $\bar{x}$ is given joint location and the cluster is represented by a Gaussian $\mathcal{N}(x, \mu, \sigma)$, the error is given by

$$Error = \sqrt{\int (x - \bar{x})^2 P(x)} = \sqrt{(\mu - \bar{x})^2 + \sigma^2} \tag{18}$$

Using these methods for quantitative evaluations, we compared the performance of our approach with that of [42]. In the first experiment, we used the data from Brown University [42] which is of *class 2* (See Figure 17). The ground truth (the 3D joint position) for the data is available. We used 7 random image pairs from this sequence for pose estimation. The full body

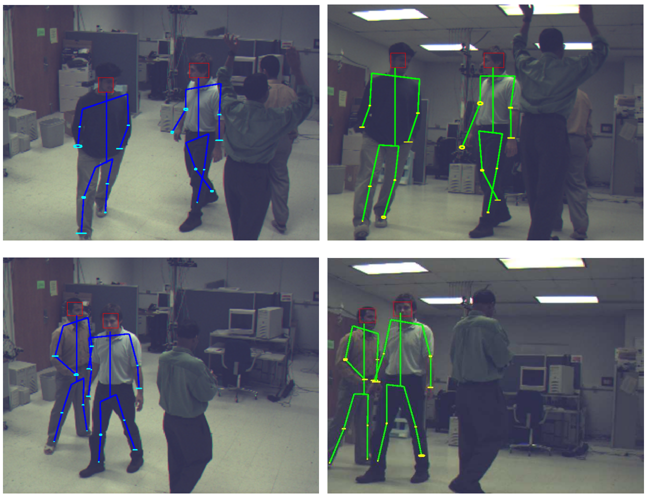


Fig. 15. More results of our Detection algorithm on Class 3 video sequences where the pose of two persons has been estimated simultaneously in the same views.

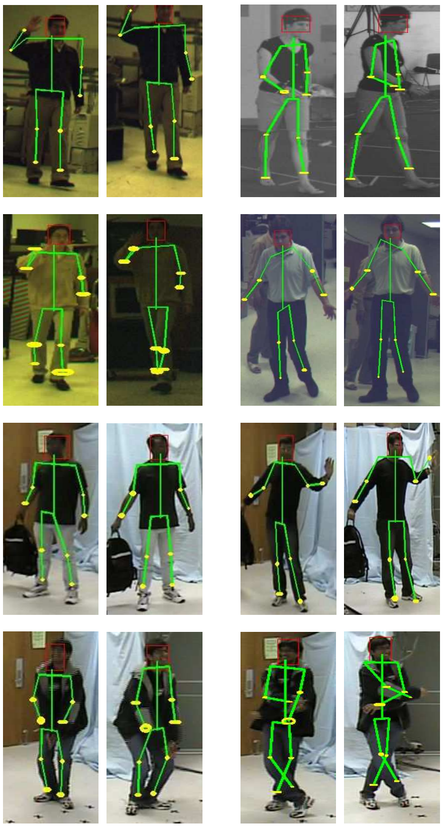


Fig. 16. Sample Views from the data-set used for quantitative evaluation.

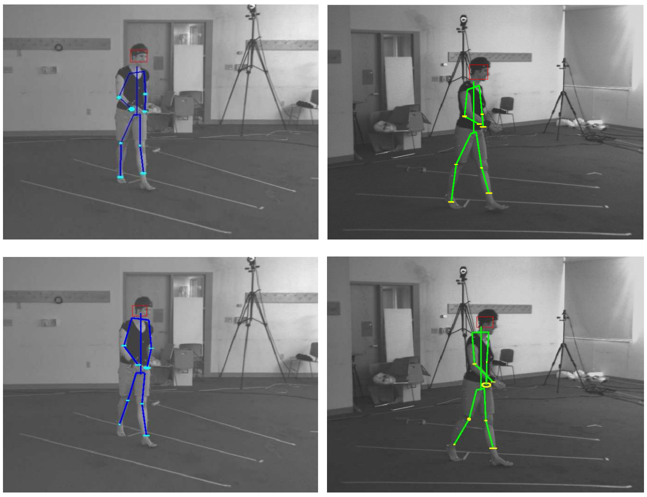


Fig. 17. Pose estimation results on the frames from the Brown University sequence that has ground truth data.

pose errors for both approaches are shown in Figure 18. Our approach improves significantly upon the performance of the basic model used in [42].

In the second experiment, we randomly selected 30 scenes from our dataset. This dataset is much more complex to process because of the following reasons:

- Presence of multiple people creates confusion
- The images have low contrast
- The images have very cluttered backgrounds, which create lots of false likelihood peaks.

While the first 21 image pairs had some occlusion, the remaining 9 image pairs had minimum or no occlusion. The joint locations in the images were hand-labeled. Figure 19 compares the performance of our algorithm with [42] by plotting the average and the maximum of the error in joint location estimation for each person. The maximum error for the approach in [42] is very high for the first 21 image pairs because of the mis-detection of occluded parts. Table II shows the average and the standard deviation of RMSE in estimation of the eight joint locations in a 22 frame video. The average height of the person in these images was 320 pixels.

We next consider performance in terms of detected and missed parts. A part is said to be missed if the RMSE is higher than a tolerance level. Ramanan et al. [34] keep a very high tolerance level and assume a part to be detected if there is any overlap between the true and estimated limb. We compare the performance by varying the tolerance level for the joint location

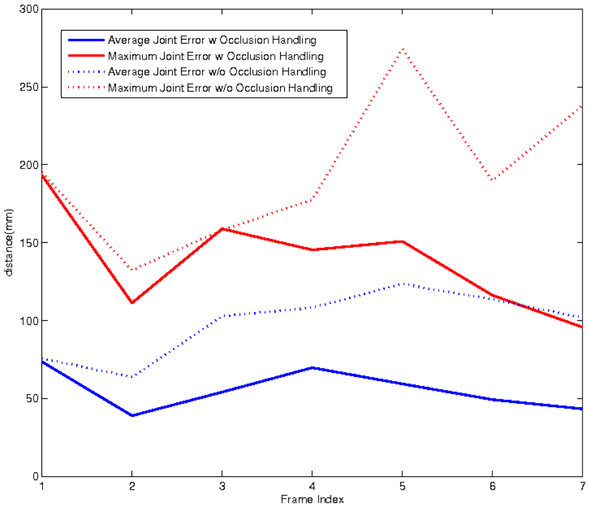


Fig. 18. Comparison of full body pose estimation error (mm) on brown data set. Distance between joints is computed in 3D space

error in terms of the limb length.

The performance of both algorithms is compared in Figure 20. 15% of the limbs in the test dataset were termed as occluded by human subjects. Our algorithm gives a limb detection rate of 97% which is significantly higher than [42]. We have a lower detection rate at low tolerance levels because of the confusion introduced by occlusion boosting. However, this occlusion boosting helps in detecting the limbs which are completely missed by Sigal et. al [42] even at higher tolerances.

We also compared the performance of our algorithm across different classes of videos. While the detection rate is above 96% for Class 1 and 2 videos, it is 92.86% for Class 3 videos. Figure 21 shows the comparison across video classes for different tolerance levels.

IX. Extension to Tracking - Including temporal constraints

A simple way to incorporate temporal consistency constraints is to utilize the locations and appearances of different parts at time $t-1$ to create priors for locations and appearances of parts at time t. These constraints can be incorporated in a belief propagation framework by adding the potentials $\tau_{t-1,t}(\Phi_i^t, \Phi_i^{t-1})$. The belief propagation message equation then changes to:

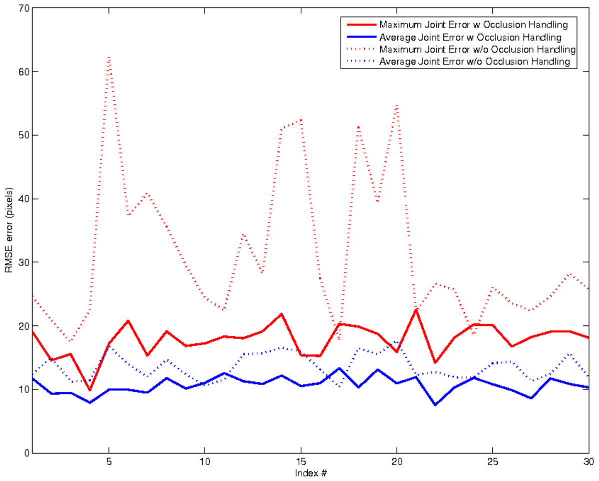


Fig. 19. Comparison of performance of the algorithm with [42] on our dataset. The RMSE error is measured in pixels.

	RMSE	Std. Dev of the RMSE	RMSE	Std. Dev of the RMSE
	(Our approach)	(Our Approach)	([42])	([42])
L. Elbow	11.53	4.6	11.77	7.37
L. Wrist	14.06	3.35	13.71	4.7
R. Elbow	7.77	3.24	9.64	3.33
R. Wrist	10.37	3.33	29.18	13.97
L. Knee	6.79	3.01	7.63	2.73
L. Ankle	14.42	4.67	15.53	5.77
R. Knee	7.22	2.59	7.13	2.91
R. Ankle	12.36	4.34	16.84	6.05

TABLE II

Average RMSE(in pixels) of the eight joints projected from 3D pose in different views for a video where Right hand and Right Leg have been occluded by other parts.

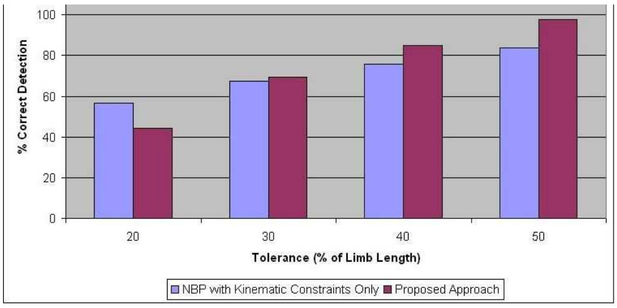


Fig. 20. Performance of our algorithm in terms of percentage of detections compared to Sigal et. al [42]

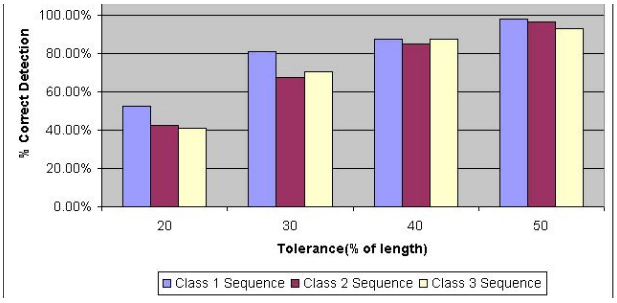


Fig. 21. Performance of our algorithm across different classes of videos.

$$m^r_{ij}(\Phi^t_j) = \int \kappa_{ij}(l^t_i, l^t_j)\alpha_{ij}(a^t_i, a^t_j)\tau_{t-1,t}(\Phi^t_i, \Phi^{t-1}_i)Pos^{r-1}(\Phi^t_i)d\Phi^t_i \tag{19}$$

Hence, the posterior at the r^{th} iteration can be written as

$$Pos^r(\Phi^t_i) \propto \sum_{v_i} P_i(I_1....I_M|\Phi^t_i, \mathbf{v^r}_i(l^t_i))P(\mathbf{v^r}_i(l^t_i))\tau_{t-1,t}(\Phi^t_i, \Phi^{t-1}_i) \prod_{k\in E_k\setminus j} m^r_{ki}(\Phi^t_i) \prod_{o\in E_a\setminus j} m^r_{oi}(\Phi^t_i) \tag{20}$$

We illustrate the approach using a very simple temporal constraint of small motion (no major location changes between two frames). This constraint is valid for videos with high frame rates and imposes the least restrictions in terms of body motions. Hence, $\tau_{t-1,t}(\Phi^t_i, \Phi^{t-1}_i)$ is modeled as a Gaussian given by

$$\tau_{t-1,t}(\Phi^t_i, \Phi^{t-1}_i) = \mathcal{N}(d(l^t_i, l^{t-1}_i), 0, \sigma^t_{ij}) \tag{21}$$

We assume that the appearance remains the same over time, and incorporate appearance consistency in our likelihood model. This assumption is true for short period of time and the appearance models can be updated after every few frames, if required. The new likelihood model can then be written as:

$$P_i(I_1....I_M|\Phi_i^t, \mathbf{v^r}_i(l_i^t)) = P_i^{Edge}(I_1....I_M|\Phi_i^t, \mathbf{v^r}_i(l_i^t))P_i^{appearance}(I_1....I_M|\Phi_i^t, \mathbf{v^r}_i(l_i^t)) \tag{22}$$

Also, instead of using full visibility to initialize belief propagation iterations, we use the occlusion maps estimated from the previous frames. The application of these constraints speeds up the inference substantially in tracking applications.

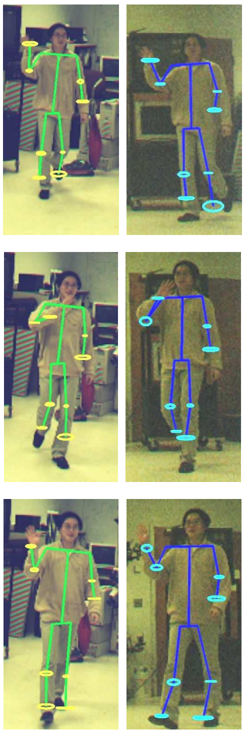


Fig. 22. Results of tracking on Class 2 videos.

A. *Experimental Results*

We show the performance of our algorithm in the tracking framework using the temporal constraints discussed above. We use sequences that have previously been used in [28]. The

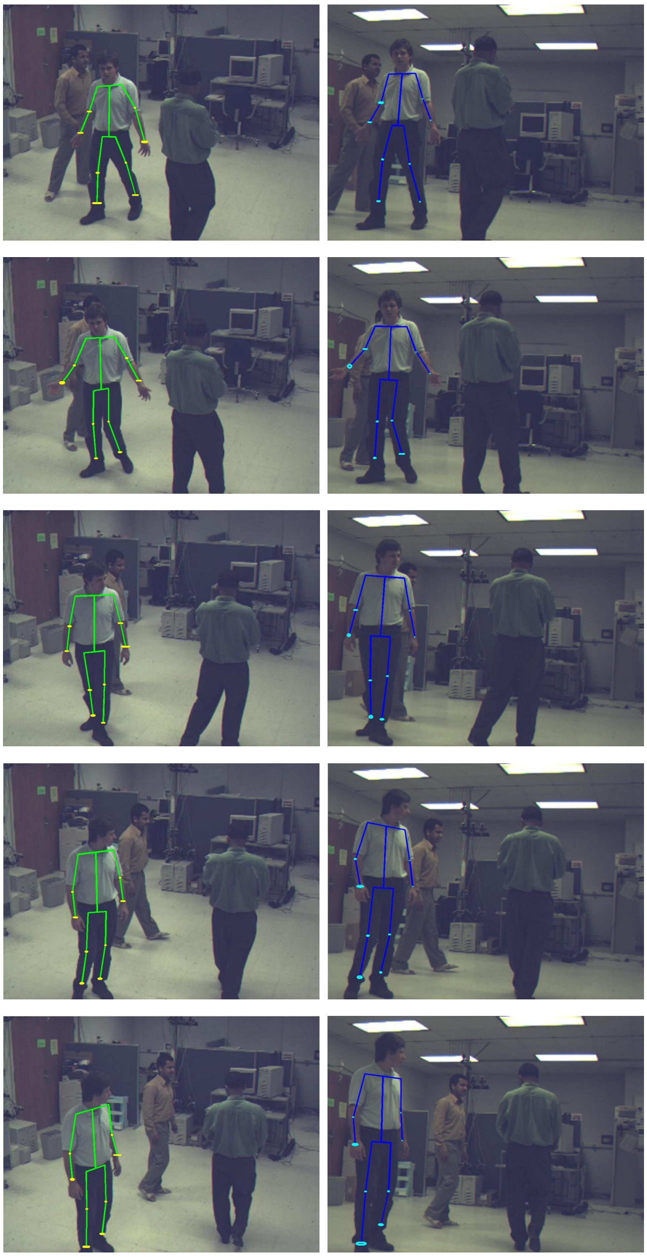


Fig. 23. Results of tracking on Class 3 videos.

average execution time for the algorithm reduced to 30 sec per frames by using the pruning via temporal priors in addition to the pruning approaches discussed earlier. For smooth variations between frames, a smoothing filter was applied to our results.

Figure 22 shows the performance of the algorithm using temporal constraints on a Class 2 video sequence. The accuracy of the system increased to 98% from 96% when temporal constraints were used (with a tolerance level of 50% of limb length).

Figure 23 shows the performance of the tracker when multiple persons are present(a Class 3 video) in the same view. In many frames there is considerable occlusion and confusion in the localization of parts because of high likelihoods from other people. Our system had 96% correct part detection when the tolerance level in the joint error was 50% of limb length.

X. Conclusion

We presented an approach for automatic initialization and tracking of the human pose in cluttered scenes. The paper integrates several constraints represented by a general non-tree constraint graph in a unified framework. These constraints include the occlusion of one part by another and the similarity in the appearance of certain parts. The approach avoids background subtraction and does not require any pixel-wise computation for reasoning about self-occlusions. We also presented an efficient method based on 2D likelihoods and epipolar geometry to search for the high likelihoods regions in the large 3D search space. This speeds up the performance of the system substantially and leads to better and faster convergence in many cases. We achieve significant improvement in results compared to existing techniques, especially when some parts are occluded in one or more views.

Acknowledgment

The authors would like to thank Homeland Security Advanced Research Project Agency for supporting the research described in the paper. The authors would also like to thank Leonid Sigal from Brown University for providing the videos with ground truth for experimental evaluation.

References

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

[2] A. Balan, L. Sigal, and M. Black. A quantitative evaluation of video-based 3d person tracking. In *VS-PETS*, 2005.

[3] M. Brand. Shadow puppetry. In *ICCV*, pages 1237-1244, 1999.

[4] 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.

[5] J. M. Coughlan and S. J. Ferreira. Finding deformable shapes using loopy belief propagation. In *ECCV*, pages 453-468, 2002.

[6] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *CVPR*, pages 886-893, 2005.

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

[8] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *CVPR*, pages 126-133, 2000.

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

[10] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, pages 66-73, 2000.

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

[12] D. Gavrila. The visual analysis of human movement. *CVIU*, pages 82-98, 1999.

[13] K. Grauman, G. Shankhnarovich, and T. Darrell. Inferring 3d structure with a statistical image-based shape model. In *ICCV*, pages 641-647, 2003.

[14] A. Gupta, A. Mittal, and L. S. Davis. Constraint integration for multi-view pose estimation of humans. In *3DPVT*, 2006.

[15] N. R. Howe. Silhouette lookup for automatic pose tracking. In *IEEE Workshop on Articulated and Non-Rigid Motion*, 2004.

[16] http://www.itl.nist.gov/iaui/ovrt/projects/anthrokids/.

[17] 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.

[18] A. Ihler, J. Fisher, and A. Wilsky. Loopy belief propagation: Convergence and effects of message errors. *Journal of Machine Learning Research*, 6:905-936, 2005.

[19] S. Ioffe and D. A. Forsyth. Finding people by sampling. In *ICCV*, pages 1092-1097, 1999.

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

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

[22] S. Ju, M. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated motion. In *Intl. Conf. on Automatic Face and Gesture Recognition*, pages 38-44, 1996.

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

[24] 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.

[25] J. MacCormick and M. Isard. Partitioned sampling, articulated objects and interface-quality hand tracking. In *ECCV*, pages 3-19, 2000.

[26] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *IJCV*, 53(3):199-223, 2003.

[27] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*, pages 69-82, 2004.

[28] A. Mittal and L. S. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 51(3):189-203, 2003.

[29] A. Mittal, L. Zhao, and L. Davis. Human body pose estimation by shape analysis of silhouettes. In *AVSS*, 2003.

[30] T. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231-268, 2001.

[31] A. Mohan, C. Papageorgiou, and T. Poggio. Example based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):349-361, 2001.

[32] 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.

[33] J. Pearl. Probabilistic reasoning in intelligent systems: Networks of plausible inference. *Morgan Kaufmann*, 1988.

[34] D. Ramanan and D. A. Forsyth. Finding and tracking people from the bottom up. In *CVPR*, pages 467-474, 2003.

[35] 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.

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

[37] R. Ronfard, C. Schmid, and B. Triggs. Learning to parse picture of people. In *ECCV*, pages 700-714, 2002.

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

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

[40] M. Siddiqui, R. Rosales, J. Alon, and S. Sclaroff. Estimating 3d body pose using uncalibrated cameras. In *CVPR*, pages 821-827, 2001.

[41] 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.

[42] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard. Tracking loose-limbed people. In *CVPR*, pages 421-428, 2004.

[43] L. Sigal and M. J. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *CVPR*, pages 2041-2048, 2006.

[44] L. Sigal, M. Isard, B. H. Sigelman, and M. J. Black. Attractive people: Assembling loose-limbed models using nonparametric belief propagation. In *NIPS(16)*, pages 1539-1546, 2004.

[45] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Discriminative density propagation for 3d human motion estimation. In *CVPR*, pages 390-397, 2005.

[46] E. Sudderth, A. lhler, W. Freeman, and A. Willsky. Nonparametric belief propagation. In *CVPR*, pages 605-612, 2003.

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

[48] S. Tatikonda and M. Jordan. Loopy belief propagation and gibbs measures. In *Uncertainty in Artificial Intelligence*, pages 493-500, 2002.

[49] Z. Tu and S. Zhu. Image segmentation by data driven markov chain monte carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):131-138, 2002.

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

[51] M. Wainwright, T. Jaakkola, and A. Wilsky. Tree-based reparameterization analysis of sum-product and its generalizations. *IEEE Transactions on Information Theory*, 49(5):1120-1146, 2003.

[52] Y. Weiss. Interpreting images by propagating bayesian beliefs.*NIPS*, 1996.

[53] Y. Weiss. Correctness of local probability propagation in graphical model with loops. *Neural Computation*, 12(1):1-41, 2000.

[54] Y. Weiss and W. Freeman. Correctness of belief propagation in gaussian graphical models of arbitrary topology. *Neural Computation*, 13:2173-2200, 2001.