Multiple Tree Models for Occlusion and Spatial Constraints in Human Pose Estimation

Yang Wang and Greg Mori

School of Computing Science, Simon Fraser University, Canada
{ywang12,mori}@cs.sfu.ca

Abstract. Tree-structured models have been widely used for human pose estimation, in either 2D or 3D. While such models allow efficient learning and inference, they fail to capture additional dependencies between body parts, other than kinematic constraints between connected parts. In this paper, we consider the use of multiple tree models, rather than a single tree model for human pose estimation. Our model can alleviate the limitations of a single tree-structured model by combining information provided across different tree models. The parameters of each individual tree model are trained via standard learning algorithms in a single tree-structured model. Different tree models can be combined in a discriminative fashion by a boosting procedure. We present experimental results showing the improvement of our approaches on two different datasets. On the first dataset, we use our multiple tree framework for occlusion reasoning. On the second dataset, we combine multiple deformable trees for capturing spatial constraints between non-connected body parts.

1 Introduction

Estimating human body poses from still images is arguably one of the most difficult object recognition problems in computer vision. The difficulties of this problem are manifold – humans are articulated objects, and can bend and contort their bodies into a wide variety of poses; the parts which make up a human figure are varied in appearance (due to clothing), which makes them difficult to reliably detect; and parts often have small support in the image or are occluded. In order to reliably interpret still images of human figures, it is likely that multiple cues relating different parts of the figure will need to be exploited.

Many existing approaches to this problem model the human body as a combination of rigid parts, connected together in some fashion. The typical configuration constraints used are kinematic constraints between adjacent parts, such as torso-upper half-limb connection, or upper-lower half-limb connection. This set of constraints has a distinct computational advantage – since the constraints form a tree-structured model, inferring the optimal pose of the person using this model is tractable.

However, this computational advantage comes at a cost. Simply put, the single tree model does not adequately model the full set of relationships between parts of the body. Relationships between parts not connected in the kinematic tree cannot be directly captured by this model.

The main contribution of this paper is developing a framework for modeling human figures as a collection of trees. We argue that this framework has the advantage of being able to locally capture constraints between the parts which constitute the model. With a collection of trees, a global set of constraints can be modeled. We demonstrate that the computational advantages of tree-structured models can be kept, and provide tractable algorithms for learning and inference in these multiple tree models. We present two applications of our framework. The first application uses the multiple tree model framework for occlusion reasoning. The second application combines multiple deformable trees to capture a richer set of spatial constraints between body parts. A preliminary version of this work appeared in a workshop paper [24] in which spatial constraints between parts were modeled. In this paper we demonstrate how the multiple tree model can be used for occlusion reasoning. We provide a model, new inference algorithms and experimental validations. We also provide an analysis of our approach which compares to existing approaches for combining multiple trees [10, 23].

The rest of this paper is organized as follows. Section 2 reviews previous work. Sections 3, 4 and 5 give the details of our approach. Section 6 presents our experimental results. Section 7 concludes this paper.

2 Related Work

One of the earliest lines of research related to finding people from images is in the setting of detecting and tracking pedestrians. Starting with the work of Hogg [5], there has been a lot of work in tracking with kinematic models in both 2D and 3D. Forsyth et al. [3] provide a survey of this work.

Some of these approaches are exemplar-based. For example, Toyama & Blake [21] use 2D exemplars for people tracking. Mori & Malik [11] and Sullivan & Carlsson [19] address the pose estimation problems as 2D template matching using pre-stored exemplars upon which joint locations have been marked. In order to deal with the complexity due to variations of pose and clothing, Shakhnarovich et al. [14] adopt a brute-force search, using a variant of locality sensitive hashing for speed. Exemplar-based approaches are effective when dealing with regular human poses. However, they cannot handle those poses that rarely occur. See Fig. 7 for some examples.

There are many approaches which explicitly model the human body as an assembly of parts. Ju et al. [7] introduce the "cardboard people" model, where body parts are represented by a set of connected planar patches. Felzenszwalb & Huttenlocher [2] develop the tree-structured pictorial structure (PS) model and apply it in 2D human pose estimation. There is also some work using non-tree structured models. Sudderth et al. [18] introduce a non-parametric belief propagation method with occlusion reasoning for hand tracking. Sigal & Black [15] use a similar idea for pose estimation. Both of them use loopy belief propagation (LBP) for the inference, so the convergence is not guaranteed. Ren et al. [13] use bottom-up detections of parallel lines as part hypotheses, and combine these hypotheses with various pairwise part constraints via an integer quadratic programming. While this sidesteps the problems of LBP, the solution relies heavily on the performance of lower-level limb detectors. Song et al. [17] detect corner features in video sequences and model them using a decomposable triangulated graph,

where the graph structure is found by a greedy search. Ioffe & Forsyth [6] use a mixture of tree model for human tracking. Again, they both rely on good low-level detectors, and cannot search the images exhaustively.

Our work is closely related to some recent work on learning discriminative models for localization. Ramanan [12] uses a variant of Conditional Random Fields (CRF) [8] for training localization models for articulated objects, such as human figures, horses, etc. Sminchisescu et al. [16] uses "mixture of experts" for visual tracking.

Our work is also related to boosting on structured outputs. Boosting was originally proposed for classification problems. Recently people have adopted it for various tasks where the outputs have certain structures (e.g., chains, trees, graphs). For example, Torralba et al. [20] use boosted random fields for object detection with contextual information. Truyen et al. [22] use a boosting algorithm on Markov Random Fields (MRF) for multilevel activity recognition.

3 Modeling the Human Body

In our method we use a combination of tree-structured deformable models for human pose estimation. The basic idea is to model a human figure as a weighted combination of several tree-structured deformable models. The parameters of each tree model are learned from training data in a discriminative fashion.

We first describe how we model the human body, and then demonstrate how this model of multiple trees can be used for modeling spatial constraints and occlusion reasoning. We also relate the pictorial structures [2] defined with pixel likelihoods and the CRF model [12] defined with patch likelihoods. This connection turns out to be useful when we develop our occlusion reasoning scheme in Sect. 5.

3.1 Single Tree-structured Deformable Body Models

Consider a human body model with K parts, where each part is represented by an oriented rectangle with fixed size. We can construct an undirected graph $G = (V, E)$ to represent the K parts (Fig. 1(a)). Each part is represented by a vertex $v_i \in V$, and there exists an undirected edge $e_{ij} = (v_i, v_j) \in E$ between vertices v_i and v_j if v_i and v_j has some dependency. Let $l_i = (x_i, y_i, \theta_i)$ be a random variable encoding the image position and orientation of the i-th part, we denote the configuration of the K part model as $L = (l_1, l_2, ..., l_K)$. Given the model parameters Θ and assuming no occlusions, the conditional probability of L in an image I can be written as:

$$P(L|I,\Theta) \propto P(L|\Theta)P(I|L,\Theta) = P(L|\alpha)\prod_i P(I|l_i,\beta_i) \tag{1}$$

where we explicitly decompose $P(L|I,\Theta)$ into the prior term $P(L|\alpha)$ and the product of several likelihood terms $P(I|l_i,\beta_i)$. Each $P(I|l_i,\beta_i)$ is a local likelihood for the part i. Assuming pixel independence, we can write each local likelihood as:

$$P(I|l_i,\beta_i) = \prod_{u\in\Omega(l_i)} P_{l_i(u)}\left(f(I_u)\right) \prod_{\Upsilon-\Omega(l_i)} P_{bg(u)}\left(f(I_u)\right) \propto \prod_{u\in\Omega(l_i)} \frac{P_{l_i(u)}(f(I_u))}{P_{bg(u)}(f(I_u))} \tag{2}$$

In the above equation, we have used the following notation. u is a pixel location in the image I, I_u is the image evidence at pixel u. In this paper, we use binary edges as our image evidences. $f(I_u)$ is a function returning 1 if I_u is an edge point, and 0 otherwise. $\Omega(l_i)$ is the set of pixels enclosed by part i as defined by l_i. Υ is the set of all pixels in the whole image. $P_{l_i(u)}$ is a binomial distribution indicating how likely pixel u is an edge point under the part l_i. $P_{bg(u)}$ is a binomial distribution for the background.

Let $\frac{P_{l_i(u)}(f(I_u))}{P_{bg(u)}(f(I_u))} = \exp(\beta_{i(u)} f(I_u))$, we will have:

$$P(I|l_i, \beta_i) \propto \prod_{u \in \Omega(l_i)} \exp\left(\beta_{i(u)} f(I_u)\right) = \exp\left(\sum_{u \in \Omega(l_i)} \beta_{i(u)} f(I_u)\right) \tag{3}$$

$$= \exp\left(\beta_i^T f_i(I(l_i))\right) \tag{4}$$

where $f_i(I(l_i))$ is the part-specific feature vector extracted from the oriented image patch at location l_i. In our case, it is a binary vector of edges for part i. β_i is a part-specific parameter that favors certain edge patterns for an oriented rectangle patch $I(l_i)$ in image I. In our formulation, β_i is simply the concatenation of $\{\beta_{i(u)}\}_{u \in \Omega(l_i)}$. We visualize β_i in Fig. 1(d).

Following previous work [12], we assume the prior term $P(L|\alpha)$ is defined in terms of the relative locations of the parts as follows:

$$P(L|\alpha) \propto \exp\left(\sum_{(i,j) \in E} \psi(l_i - l_j)\right) \tag{5}$$

Most previous approaches use Gaussian shape priors $\psi(l_i - l_j) \propto \mathcal{N}(l_i - l_j; \mu_i, \Sigma_i)$ [2]. However, since we are dealing with images with a wide range of poses and aspects, Gaussian shape priors seem too rigid. Instead we choose a spatial prior using discrete binning (Fig. 1(c)) similar to the one used in Ramanan [12]:

$$\psi(l_i - l_j) = \alpha_i^T \text{bin}(l_i - l_j) \tag{6}$$

$\text{bin}(\cdot)$ is a vector of all zeros with a single one for the occupied bin. α_i is a parameter that favors certain spatial and angular bins for part i with respect to its parent j. This spatial prior captures more intricate distributions than a Gaussian prior.

Combining (1), (4) and (5), we obtain the following formulation:

$$P(L|I, \Theta) \propto \exp\left(\sum_{(i,j) \in E} \psi(l_i - l_j) + \sum_{i=1}^{K} \phi(l_i)\right) \tag{7}$$

where $\phi(l_i)$ is a potential function that models the local image evidence for part i located at l_i. $\phi(l_i)$ is defined as $\phi(l_i) = \beta_i^T f_i(I(l_i))$.

Equation (7) is exactly the same Conditional Random Field (CRF) formulation of human pose estimation problem in Ramanan [12]. This shows the two different formulations (pictorial structures [2] and CRF [12]) of human pose estimation problems are in fact equivalent. The only difference is that they use different criteria for model

parameter learning, i.e., maximizing the joint likelihood (ML) or the conditional likelihood (CL). In the following, we will use the CRF formulation in (7). But we will come back to the pictorial structure formulation of (1), when we develop our occlusion reasoning method.

To facilitate tractable learning and inference, G is usually assumed to form a tree $T = (V, E_T)$ [2, 12]. In particular, most work uses the kinematic tree (Fig. 1(b)) as the underlying tree model.

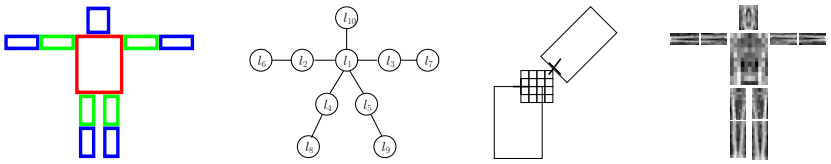


Fig. 1. Representation of a human body: (a) human body represented as a 10-part model; (b) corresponding kinematic tree structured model; (c) discrete binning for spatial prior; (d) visualization of the learned edge-based appearance model β_i for each body part. Dark areas correspond to small values of β_i, and bright areas correspond to large values of β_i

Inference in a single tree-structured model can be done by message-passing. Using 3D convolution, one can search exhaustively over all part locations in an image without relying on feature detectors. Learning of the model parameters can be done by closed-form solutions (ML) or gradient ascent methods (CL). See [12, 24] for details.

3.2 Multiple Tree Models

In our work we model the human body by a collection of multiple tree-structured models. For example, one can use the two tree models in Fig. 4 to model the kinematic constraints and the occlusion relationships between the legs. One can also use different tree structures (e.g., Fig. 6) to model the spatial constraints that are not captured in the kinematic tree. The weighting parameters which combine the multiple tree models can also be learned in a discriminative fashion using boosting (Sect. 4).

The final form of our model is:

$$F(L, I; \Theta) = \sum_t w_t f_t(L, I; \Theta) \tag{8}$$

where $f_t(L, I; \Theta)$ is a single tree model with tree structure τ_t, w_t is the weight associated with this single tree model. The optimal pose L^* can be obtained in our model as $L^* = \arg\max_L F(L, I; \Theta)$. In the next section, we describe the algorithm for learning all the model parameters $\Theta = \{w_t, \Theta_t\}$.

4 Spatial Constraints with Multiple Trees

Our learning algorithm for spatial constraints is based on AdaBoost.MRF proposed in Truyen et al. [22]. Given an image I, the problem of pose estimation is to find the

best part labeling L^* that maximizes $F(L, I)$, i.e. $L^* = \arg\max_L F(L, I)$. $F(L, I)$ is known as the "strong learner" in the boosting literature. Given a set of training examples $(I^i, L^i), i = 1, 2, ..., N$. $F(L, I)$ is found by minimizing the following loss function L_O:

$$L_O = \sum_i \sum_L \exp\left(F(I^i, L) - F(I^i, L^i)\right) \tag{9}$$

We assume $F(L, I)$ is a linear combination of a set of so-called "weak learners", i.e., $F(I, L) = \sum_t w_t f_t(L, I)$. The t-th weak learner $f_t(L, I)$ and its corresponding weight w_t are found by minimizing the loss function defined above, i.e. $(f_t, w_t) = \arg\max_{f,w} L_O$. In our case, we choose the weak learner as $f(L, I) = \log p(L|I)$. To achieve computational tractability, we assume each weak learner is defined on a tree model.

If we can successfully learn a set of tree-based weak learners $f_t(L, I)$ and their weights w_t, the combination of these weak learners captures more dependencies than a single tree model. At the same time, the inference in this model is still tractable, since each component is a tree.

Optimizing L_O is difficult, Truyen et al. [22] suggest optimizing the following alternative loss function: $L_H = \sum_i \exp\left(-F(L^i, I^i)\right)$. It can be shown that L_H is an upper bound of the original loss function L_O, provided that we can make sure $\sum_j w_j = 1$. In Truyen et al. [22], the requirement $\sum_j w_j = 1$ is met by scaling down each previous weak learner's weight by a factor of $1-w_t$ as $w_j^{'} \leftarrow w_j(1-w_t)$, for $j = 1, 2, ..., t-1$, so that $\sum_{j=1}^{t-1} w_j^{'} + w_t = \sum_{j=1}^{t-1} w_j(1-w_t) + w_t = 1$, since $\sum_{j=1}^{t-1} w_j = 1$. In practice, we find that this trick sometimes scales down previous weak learners to have zero weights. So we use a slightly different method by scaling down each weak learner's weight up to t by a factor of $1/(1 + w_t)$. It can be shown that we still have $\sum_{j=1}^{t} w_j = 1$. Figure 2 shows the overall algorithm.

Discussion: Our model is similar to "mixtures of trees" (MoT) [10] at a first glance, but there are some important differences. MoT is a generative model developed for density modeling problem. It is not designed for classification or prediction. Although one can use MoT as the spatial prior in a generative fashion, it is not clear how to learn the model in a discriminative way. Instead, our model is trained discriminatively, and our objective function is more closely tied to inference.

Another similar work is the tree-reweighted message passing (TRW)[23]. TRW aims to approximate the partition function in MRF, it does not answer the question of learning a good model for recognition, i.e., TRW assumes the MRF model is given, and it simply tries to solve the inference problem. Plus, TRW is an iterative algorithm, and its convergence is still an unsolved problem.

5 Occlusion Reasoning with Multiple Trees

In this section, we apply the multiple tree framework to the "double counting of image evidence" problem in human pose estimation illustrated in the top row of Fig. 3, where the same image patch is used twice to explain two different body parts. Previous approaches [9] have focused on using strong priors of body poses to solve this problem.

Input: $i = 1, 2, ..., D$ data pairs, graphs $\{G_i = (V_i, E_i)\}$
Output: set of trees with learned parameters and weights
Select a set of spanning trees $\{\tau\}$
Choose the number of boosting iterations T
Initialize $\{w_{i,0} = \frac{1}{D}\}$, and $w_1 = 1$
for each boosting round $t = 1, 2, ..., T$
    Select a spanning tree τ_t
    /* Add a weak learner */
    $\Theta_t = \arg\max_{\Theta} \sum_i w_{i,t-1} \log P_{\tau_t}(L_i, I_i|\Theta)$
    $f_t = \log P_{\tau_t}(L|I, \Theta_t)$
    if $t > 1$ **then**
        select the step size $0 < w_t < 1$ using line searches
    end if
    /* Update the strong learner */
    $F_t = \frac{1}{1+w_t} F_{t-1} + \frac{w_t}{1+w_t} f_t$
    /* Scale down the previous learners' weights*/
    $w_j \leftarrow \frac{w_j}{1+w_t}$, for $j = 1, 2, ..., t$
    /* Re-weight training data*/
    $w_{i,t} \propto w_{i,t-1} \exp(-w_t f_{i,t})$
end for
Output $\{\tau_t\}$,$\{\Theta_t\}$ and $\{w_t\}$, $t = 1, 2, ..., T$

Fig. 2. Algorithm of boosted multiple trees

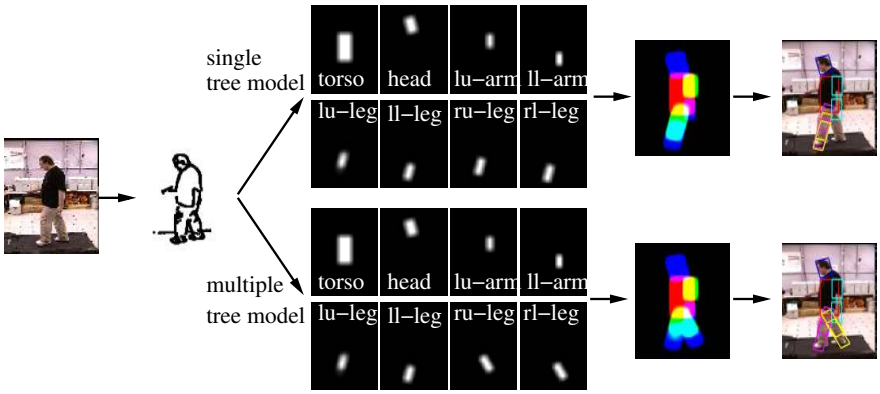


Fig. 3. Illustration of "double counting of image evidence" problem: top row shows how the same piece of image patch is used to explain two body parts, the bottom row shows how our occlusion reasoning mechanism using multiple trees can alleviate this problem

However, these approaches are limited to the cases of normal poses and known activities. We believe the proper way to solve this problem is to introduce occlusion reasoning in the model. In our multiple tree framework, we can define one tree for the kinematic constraint (e.g., Fig. 4(a)), and a second tree for the occlusion relationships (e.g., Fig. 4(b)). In this section, we discuss how to incorporate occlusion reasoning into the human body model introduced in Sect. 3, and how to do inference in a tree model involving occlusion relationships (see Fig. 4(b)). Before we proceed, we first clarify the terminology we are using. By "occlusion reasoning", we do not necessarily mean the body parts in the image are occluding each other, instead we use "occlusion" to refer to the particular problem of using the same image patch to explain different body parts, as illustrated in Fig. 3.

Occlusion-sensitive formulation: The factorization of the global likelihood into local likelihood terms in Eq. 1 is valid only if the local terms $P(I|l_i, \beta_i)$ for $i \in \{1..K\}$ are independent. This assumption holds when there are no occlusions among different parts. In order to obtain a similar decomposition (hence distributed inference) when occlusions exist, we augment the configuration l_i of part i with a set of binary hidden variables $z_i = \{z_{i(u)}\}_{u \in \Upsilon}$, similar to [18]. Note that there is a binary variable $z_{i(u)}$ for each *pixel*. Let $z_{i(u)} = 0$ if pixel u in the area enclosed by part i is occluded by *any* other part, and 1 otherwise. If a part is partially occluded, only a subset of these binary variables are zeros. Letting $Z = \{z_1, z_2, ..., z_K\}$, the local likelihood term (2) can be rewritten as:

$$P(I|L, Z, \Theta) = \prod_i P(I|l_i, z_i, \beta_i) \tag{10}$$

$$\propto \prod_i \prod_{u \in \Omega(l_i)} \left(\frac{P_{l_i(u)}(f(I_u))}{P_{bg(u)}(f(I_u))} \right)^{z_i(u)} \tag{11}$$

$$= \prod_i \prod_{u \in \Omega(l_i)} \left(\exp\left(\beta_{i(u)} f(I_u)\right) \right)^{z_i(u)} \tag{12}$$

It is important to note that if all the occlusion variables z_i are consistent, the global likelihood $P(I|L, Z, \Theta)$ truly factorizes as (12). Similar to [18], we enforce the consistency of the occlusion variables using the following function:

$$\eta(l_j, z_{i(u)}; l_i) = \begin{cases} 0 \text{ if } l_j \text{ occludes } l_i,\ u \in \Omega(x_j), \text{ and } z_{i(u)} = 1 \\ 1 \text{ otherwise} \end{cases}$$

The consistency relationship of occlusion variable z_i and z_j can be enforced by the following potential function:

$$\psi^O(l_i, z_i, x_j, z_j) = \prod_{u \in \Upsilon} \eta(x_j, z_{i(u)}; x_i) \eta(x_i, z_{j(u)}; x_j) \tag{13}$$

Letting $\mathcal{E}_O$ be the set of edges corresponding to pairs of parts that are prone to occlusions, and defining $P_O(L, Z) \propto \prod_{(i,j) \in \mathcal{E}_O} \psi^O_{i,j}(l_i, z_i, l_j, z_j)$, we obtain the final occlusion sensitive version of our model:

$$P(L|I, Z, \Theta) \propto P(L|\alpha) P_O(L, Z) P(I|L, Z, \beta) \tag{14}$$

Occlusion-sensitive message passing: Now we discuss how to do message passing that involves occlusion variables z_i. Similar to previous work [18, 15], we assume that potentially occluding parts have a known relative depth in order to simplify the formulation. In general, one could introduce another discrete hidden variable indicating the relative depth order between parts and perform inference for each value.

Our inference scheme is similar to [18]. It is based on the following intuition. Suppose part j is occluding part i and we have a distribution of $P(l_j)$, we can use $P(l_j)$ to calculate an occlusion probability $P[z_{i(u)} = 0]$ for each pixel u. Then we can discount the image evidence at pixel u according to $P[z_{i(u)} = 0]$ when we use that image evidence to infer the configuration of l_i. If $P[z_{i(u)} = 0]$ is close to 1, it means pixel u has a higher probability of being claimed by part j. In this case, we will discount more of the image evidence at u. In the extreme case of $P[z_{i(u)} = 0]$ approaches 0 for all $\{u : u \in \Upsilon\}$, it is equivalent to inference without occlusion reasoning.

Consider the BP message sent from l_j to (l_i, z_i) in message passing. At this point, we already have a pseudo-marginal $P(\hat{l_j}|I))$ (it is the true marginal $P(l_j|I)$ if the underlying graph structure is a tree, and the message is passed from the root to the leaves). If l_i lies in front of l_j (remember that we known the depth order), the BP message $\mu_{j,i(u)}(z_{i(u)})$ is uninformative. If l_i is occluded and l_j is the only potentially occluding part, we firstly determine an approximation to the marginal occlusion probability $\nu_{i(u)} \approx Pr[z_{i(u)} = 0]$. If we think of $P(\hat{l_j}|I)$ as a 3D image (x, y, θ), $\nu_{i(u)}$ (which can be thought as a 2D image) can be efficiently calculated by convolving $P(\hat{l_j}|I)$ with rotated version of a uniform rectangle (with size proportional to the size of l_j) filter, then summing over θ dimension. Then the BP approximation to l_i can be written in terms of these marginal occlusion probabilities (see [18] for the rationale behind (15)):

$$P(I|l_i) \propto \prod_{u \in \Omega(l_i)} \left[\nu_{i(u)} + (1 - \nu_{i(u)}) \left(\frac{P_{l_i(u)}(f(I_u))}{P_{bg(u)}(f(I_u))} \right) \right] \tag{15}$$

$$= \prod_{u \in \Omega(l_i)} \left[\nu_{i(u)} + (1 - \nu_{i(u)}) \exp\left(\beta_{i(u)} f(I_u)\right) \right] \tag{16}$$

$$\approx \prod_{u \in \Omega(l_i)} \left[\exp\left(\left(1 - \nu_{i(u)}\right) \beta_{i(u)} f(I_u) \right) \right] \tag{17}$$

$$= \exp\left(\sum_{u \in \Omega(l_i)} \left(\left(1 - \nu_{i(u)}\right) \beta_{i(u)} f\left(I_u\right) \right) \right) \tag{18}$$

$$= \exp\left(\beta_i g_i(I(l_i), \nu_i)\right) \tag{19}$$

where $g_i(I(l_i), \nu_i)$ is a function similar to $f_i(I(l_i))$, but instead of returning 1, it returns a fractional number $(1 - \nu_{i(u)})$ at pixel u if I_u is an edge point. The approximation in (17) is based on the fact that absolute values of $\beta_{i(u)}$ are usually small (e.g., less than 0.6 in our experiments). When $|x|$ is small, $\exp(x)$ can be approximated by $1 + x$ based on the truncated Taylor expansion of $\exp(x)$.

Unlike previous methods [18, 15] which handle occlusion reasoning using sampling, our final result (19) has a surprisingly simple form. It can be efficiently calculated by first getting $g_i(I(l_i), \nu_i)$ through a simple dot-product between $f(I)$ (a binary

2D edge map of the whole image I) and $(1 - \nu_i)$ (a 2D image of occlusion marginals), then convolving g_i with rotated versions of β_i. The dot-product has the nice intuition of discounting the image evidences by their occlusion variables. Our method can be applied efficiently and exhaustively over all the image pixel locations. This is due to the convolution trick. However, if the structure of the graphical model is not a tree, one has to use loopy belief propagations. In that case, the convolution trick is no longer valid, since the message stored at a node is no longer in a simple form that allows the derivation of (19) to go through. This further justifies the advantage of using tree-structured models.

6 Experiments

CMU MoBo dataset: We first test our algorithm on the rescaled versions of side-view persons of CMU mobo dataset [4] for the occlusion reasoning. Since people's right arm in this dataset is almost always occluded, we only try to infer one arm. We use the background subtraction masks that come with this dataset to remove the edges found in the background.

We use the two tree structures shown in Fig. 4. The first tree captures the kinematic spatial constraint. The second tree captures the occlusion relationships between the left and right legs. Inference in the second tree uses the message passing algorithm described in Sect. 5. Learning the model parameters is a bit tricky. If we use CMU mobo dataset for training, we will probably end up with a strong spatial prior specifically tuned to side-view walking. Instead, we learn the model parameters $\Theta = \{\alpha_i, \beta_i\}$ using the same training set in our second experiment (see below). That dataset contains images of people with a variety of poses. We manually set the weights of these two trees to be equal, since we do not have appropriate datasets with ground truths, and we do not want to learn the parameters from the mobo dataset. In principle, this parameter can be learned from some labeled dataset where the relative depth order of parts is known.

Some of the sample results are shown in Fig. 5. We can see that the single tree model tends to put the legs on top of each other. But our method correctly infers the configurations of both legs. To quantify the results, we manually label 300 mobo images as ground truths and measure their *perplexity* (or *negative log-probability* [12]) under the learned model. Instead of measuring the perplexity for the whole body pose L, we measure them separately for each body part $l_i (i = 1, 2, ..., K)$ to emphasize the effect of occlusion reasoning between two legs. As shown in Table 1, our method achieves lower perplexity on the lower and upper right legs. The perplexities for other body parts are not shown in the table since they are the same for both methods. This is because we have only modeled the occlusion relationships between the legs.

People dataset: We test our algorithm on the people dataset used in previous work [12]. This dataset contains 305 images of people in various poses. First 100 images and their mirror-flipped versions are used for training, and the remaining 205 images for testing. We manually select three tree structures shown in Fig. 6, although it will be an

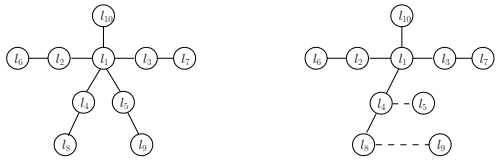


Fig. 4. Two tree structures used on CMU Mobo dataset. We use dashed lines to indicate occlusion relationships, rather than spatial constraints

Table 1. Quantitative measurement on mobo dataset for the right upper and lower legs. Smaller perplexities mean better performance

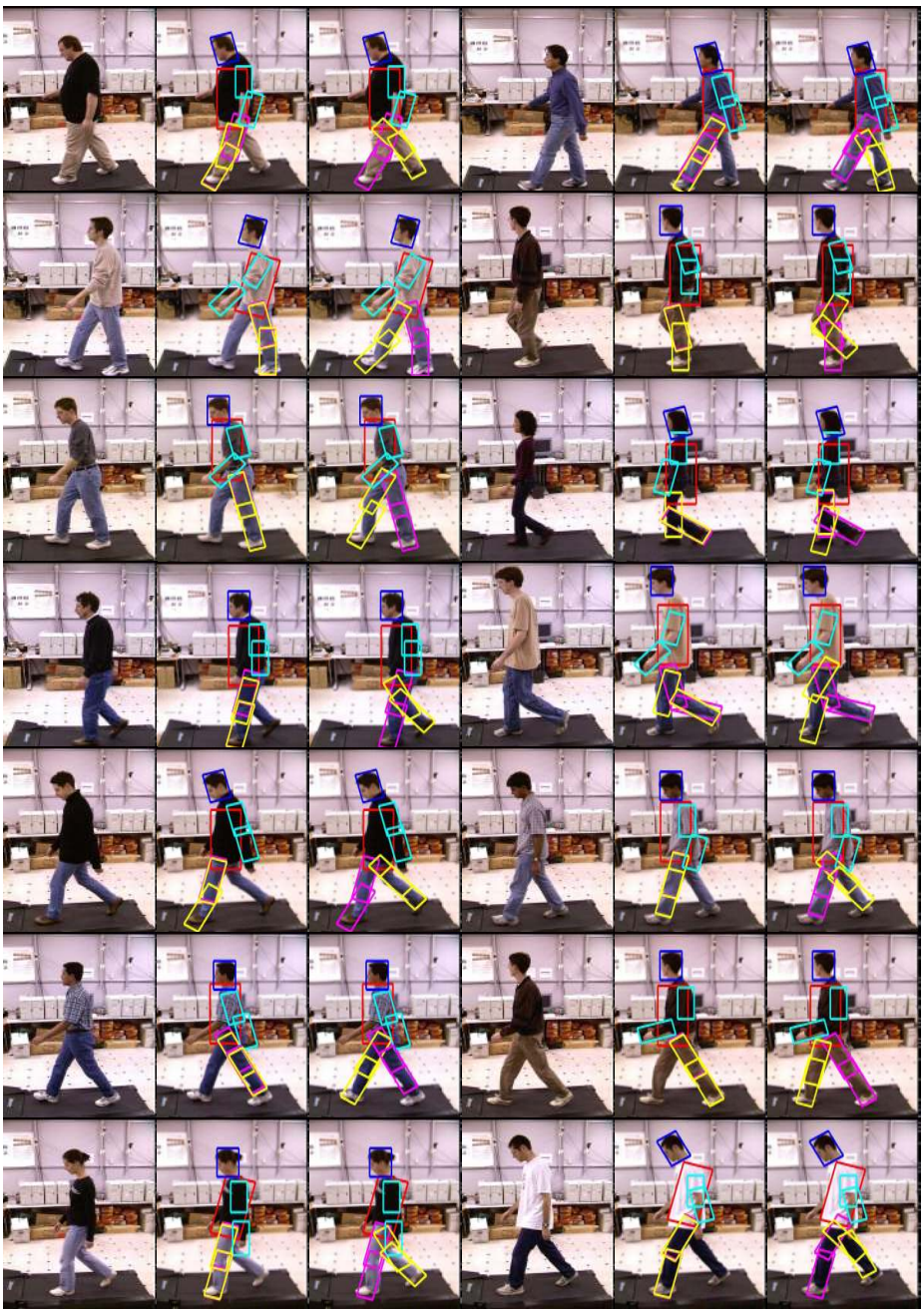
Part	Perplexity(two trees)	Perplexity(one tree)
ru-leg	32.4939	33.9706
rl-leg	26.7597	33.6693

interesting future work on how to automatically learn the tree structure at each iteration in an efficient way. We visualize the distribution $P(L|I)$ as a 2D image using the same technique in [12], where the torso is rendered as red, the upper-limbs as green, the lower-limbs and the head as blue. Some of the parsing results are shown in Fig. 7. We can see that our parsing results are much clearer than the one using the kinematic tree. In many images, the body parts are almost clearly visible from our parsing results. In the results of using the kinematic tree, there are many white pixels, indicating high uncertainty about body parts at those locations. But with multiple trees, a lot of these white pixels are cleaned up. It is plausible that if we sample the part candidates l_i according to $P(l_i|I)$ and use them as the inputs to other pose estimation algorithms (e.g., Ren et al. [13]), the samples generated from our parsing results are more likely to be the true part locations.

7 Conclusion

We have presented a framework for modeling human figures as a collection of tree-structured models. This framework has the computational advantages of previous tree-structured models used for human pose estimation. At the same time, it models a richer set of constraints between body parts. We demonstrate our results on side-walking persons in CMU mobo dataset, and a challenging people dataset with substantial pose variations.

Human pose estimation is an extremely difficult computer vision problem. The solution of this problem probably requires the symbiosis of various kinds of visual cues. Our framework provides a flexible way of modeling dependencies between non-connected body parts.



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

Fig. 5. Sample results on the CMU mobo dataset: (a) original images; (b) results of using one kinematic tree; (c) results of using multiple trees for occlusion reasoning

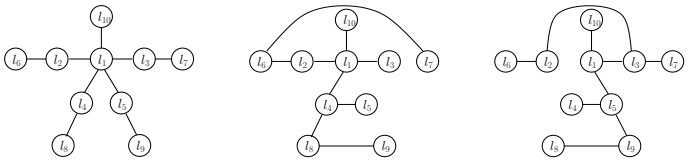
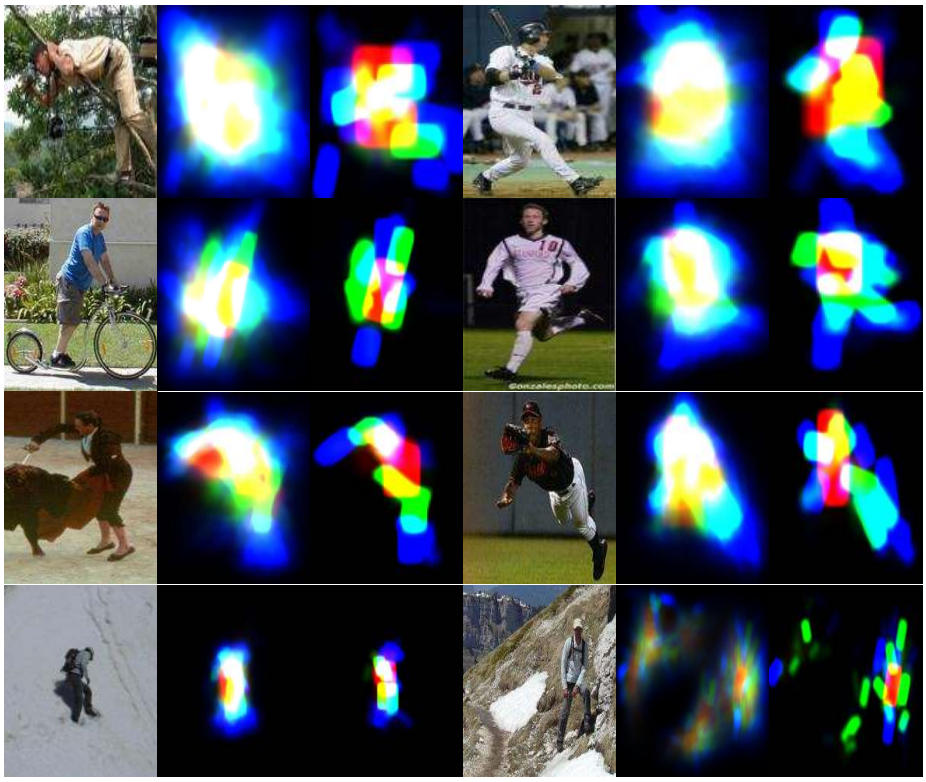


Fig. 6. Three tree structures used on people dataset



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

Fig. 7. Sample results on the people dataset: (a) original images; (b) results of using one kinematic tree; (c) results of using multiple trees

References

1. D. Crandell, P. F. Felzenszwalb, and D. P. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *IEEE CVPR*, 2005.
2. P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2003.
3. D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, and D. Ramanan. Computational studies of human motion: Part 1, tracking and motion synthesis. *Foundations and Trends in Computer Graphics and Vision*, 1(2/3):77–254, July 2006.
4. R. Gross and J. Shi. The cmu motion of body(mobo) database. Technical Report CMU-RI-TR-01-18, CMU, 2001.
5. D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision Computing*, 1(1):5–20, 1983.
6. S. Ioffe and D. Forsyth. Human tracking with mixtures of trees. In *IEEE ICCV*, 2001.
7. S. X. Ju, M. J. Black, and Y. Yaccob. Cardboard people: A parameterized model of articulated image motion. In *Proc. Automatic Face and Gesture Recognition*, 1996.
8. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.
9. X. Lan and D. P. Huttenlocher. Beyond trees: Common-factor models for 2d human pose recovery. In *IEEE ICCV*, 2005.
10. M. Meila and M. I. Jordan. Learning with mixtures of trees. *Journal of Machine Learning Research*, 1:1–48, 2000.
11. G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *ECCV*, 2002.
12. D. Ramanan. Learning to parse images of articulated bodies. In *NIPS 19*, 2007.
13. X. Ren, A. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *IEEE ICCV*, 2005.
14. G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *IEEE ICCV*, 2003.
15. L. Sigal and M. J. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *IEEE CVPR*, 2006.
16. C. Sminchisescu and A. Kanaujia and D. Metaxas. BM^3E: Discriminative Density Propagation for Visual Tracking. *IEEE PAMI*, 29(11):2030-2044, November 2007.
17. Y. Song, L. Goncalves, and P. Perona. Unsupervised learning of human motion. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25(7):814–827, July 2003.
18. E. B. Sudderth, M. I. Mandel, W. T. Freeman, and A. S. Willsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, 2004.
19. J. Sullivan and S. Carlsson. Recognizing and tracking human action. In *ECCV*, 2002.
20. A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *NIPS 17*, 2005.
21. K. Toyama and A. Blake. Probabilistic exemplar-based tracking in a metric space. In *IEEE ICCV*, 2001.
22. T. T. Truyen, D. Q. Phung, H. H. Bui, and S. Venkatesh. AdaBoost.MRF: Boosted markov random forests and application to multilevel activity recognition. In *IEEE CVPR*, 2006.
23. M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. A new class of upper bounds on the log partition function. *IEEE Transactions on Information Theory*, 51(7):2313–2335, July 2005.
24. Y. Wang and G. Mori. Boosted multiple deformable trees for parsing human poses. In *ICCV Workshop on Human Motion Understanding, Modeling, Capture and Animation*, 2007.