Beyond Trees: Common-Factor Models for 2D Human Pose Recovery

Xiangyang Lan Daniel P. Huttenlocher
Department of Computer Science, Cornell University
{xylan,dph}@cs.cornell.edu

Abstract

Tree structured models have been widely used for determining the pose of a human body, from either 2D or 3D data. While such models can effectively represent the kinematic constraints of the skeletal structure, they do not capture additional constraints such as coordination of the limbs. Tree structured models thus miss an important source of information about human body pose, as limb coordination is necessary for balance while standing, walking, or running, as well as being evident in other activities such as dancing and throwing. In this paper we consider the use of undirected graphical models that augment a tree structure with latent variables in order to account for coordination between limbs. We refer to these as common-factor models, since they are constructed by using factor analysis to identify additional correlations in limb position that are not accounted for by the kinematic tree structure. These common-factor models have an underlying tree structure and thus a variant of the standard Viterbi algorithm for a tree can be applied for efficient estimation. We present some experimental results contrasting common-factor models with tree models, and quantify the improvement in pose estimation for 2D image data.

1. Introduction

Human pose estimation from a single viewpoint is a challenging and important problem in computer vision. As cameras become standard computer peripherals there are many possible applications for "looking at people" (e.g., see [10]). There has been substantial progress on estimating human pose from a single viewpoint, however the problem remains quite difficult. Many recent approaches are based on using a tree-structured model that captures the kinematic relations between parts such as the torso and limbs (e.g., [4, 6, 9, 11]). In such kinematic tree models each body part corresponds to a node in a graph, and two nodes are connected by an edge when there is a joint connecting the corresponding body parts.

Kinematic tree models are powerful because they enable pose estimation to be done efficiently, in time linear in the number of body parts, while capturing what is arguably the most important source of constraint on human body pose, the joints connecting the limbs. However such models are also limited by the fact that they do not represent information about relations between limbs that are not connected by joints. Thus important sources of constraint such as balance and coordination are not captured. It is not a simple matter to add more constraints, because the computational complexity of estimation is exponential in the size of the largest clique in the graph. Thus for example, adding constraints between the arms and legs to account for balance would result in a nearly fully connected graph, and quickly make the estimation problem intractable.

In this paper we investigate a technique for adding constraint to the model while not greatly increasing the computational cost of estimation. The key idea is to introduce a small number of latent variables to represent *residual* correlations between parts that are not captured by a tree model. This kind of approach has recently been investigated in more general graphical models (e.g., [12]). In a bit more detail, we start with a kinematic tree model and identify parts where correlation in their locations violates the conditional independence assumption of the tree model. We then use factor analysis to find the best common factor that accounts for these correlations. This common factor is added to the tree model as a *latent variable*. The resulting common-factor model preserves the underlying tree structure, which allows a variant of the Viterbi algorithm to be used for efficient pose estimation. Intuitively, the tree captures the kinematic constraints and the common-factor model then seeks additional constraints that can be represented by augmenting the tree structure in a manner that does not substantially change the computational tractability.

To demonstrate the approach we compare a standard kinematic tree model with a model that has a single additional latent variable to account for coordination of the upper arms and legs. Intuitively this corresponds to the symmetry in the orientations of the upper limbs with respect to the torso that is used to maintain balance (note that the constraint is determined automatically from training data as briefly described above and discussed in more detail in

subsequent sections). The addition of this single constraint leads to substantial improvement in the accuracy of pose estimation when compared with a tree model. We show this qualitatively as well as quantitatively using a sequence from [13] where data from motion capture markers serves as ground truth. The common-factor model provides substantially better performance, more than halving the average localization error compared to the tree model.

2. Related Work

There are a wide range of approaches to human pose estimation. Much of the work uses 3D models and multiple image sources (e.g. see the recent paper by [13]). In contrast our focus is on the use of 2D models and a single viewpoint. Another popular class of approaches is based on active contour models (e.g., [15]) and tracking edge contours over time. In contrast our approach uses a generative model consisting of parts and relations between parts. Our method works on a single frame, whereas the contour tracking approaches generally require motion between successive frames.

The approaches most closely related to ours are those which model the 2D projection of the human form in terms of rectangular parts with spring-like constraints between those pairs of parts that are connected by joints. The cardboard people model of [7] uses a kinematic chain, and subsequent work by [4, 6, 9, 11] uses a full kinematic tree. Such tree-based methods often use statistical sampling methods to esimate mulitple possible poses, and then select among hypotheses based on other criteria. While sampling techniques could be used with the model developed here, instead we investigate the power of a more constrained model to find the best pose via MAP estimation.

The work of [14] is similar to ours in that it considers graphical models with more constraint than a tree model, but which still allow for efficient estimation algorithms. Their approach is to use a form of triangulated graph that has cliques of size at most 3 (as recall the complexity of estimation is exponential in the clique size). In contrast, we stay with a tree-structured model where estimation can be done in linear time, but then augment that tree with latent variable(s) that must be explicitly optimized over. Another difference in the approaches is their work uses local point feature detectors as opposed to the limb-sized part models that we employ.

3. Trees and their Limitations

Consider an object with n parts, where each part is represented by a vertex $v_i \in V$, and there is an undirected edge $e_{ij} = (v_i, v_j)$ between each pair of vertices that has an explicit spatial dependency. Let l_i be a random variable representing the location of part v_i, and $L = (l_1, \ldots, l_n)$ be the overall spatial configuration of the model. Following the work of [4] and [6] the location of each part is parameterized by $l_i = (x, y, s, o)$ where (x, y) is the location of a reference point on the part, s is a scale factor that corresponds to foreshortening, and o is the part orientation. We use the notation L_S to denote the locations for $S \subset V$, for example if $S = \{v_i, v_j, v_k\}$ then $L_S = (l_i, l_j, l_k)$.

First consider a graph with edges E_T that forms a tree $T = (V, E_T)$. For such a model the prior over location, or spatial model, $p(L)$ factors into products involving the edges and the nodes,

$$p(L) = \frac{\prod_{(v_i,v_j)\in E_T} p(l_i, l_j)}{\prod_{v_i \in V} p(l_i)^{d(v_i)-1}},$$

where $p(\cdot)$ is the marginal probability of its arguments, and $d(v_i)$ is the degree of the vertex v_i.

Following [4] we assume that there is no meaningful prior on the location of an individual part. The spatial relations are all on relative locations of parts rather than on absolute location. In this case the prior can be rewritten as a product over the edges. In general we use potential functions $\phi(\cdot)$ rather than distributions to avoid normalization computations, yielding,

$$p(L) \propto \prod_{(v_i,v_j)\in E_T} \phi_{ij}(l_i, l_j), \tag{1}$$

where ϕ_{ij} is a potential function over the clique (pair of nodes) v_i and v_j.

As is common, we use a spring-like model for the connection between limbs. Thus the clique potential for a pair of parts connected by an edge e_{ij} is of the form,

$$\phi_{ij}(l_i, l_j) = N(T_{ij}(l_i) - T_{ji}(l_j), 0, \Sigma_{ij}),$$

where N is an (unnormalized) Gaussian with mean zero. T_{ij} and T_{ji} are linear transformations that bring l_i and l_j to an ideal relative orientation and scale about a connected pivot point, and Σ_{ij} is a covariance matrix. Conceptually this corresponds to a simple spring model of a revolute joint that connects two parts, where T_{ij} and T_{ji} encode the mean or ideal relative position of the joint and Σ_{ij} encodes the degree of flexibility in the joint.

For a human body, the variance in the relative orientations of two connected limbs is generally quite high compared to the other location parameters of position and scale. Another alternative would be to represent the orientation using a Potts-like model which specifies an allowable range of orientations rather than using a high variance Gaussian. We have found little difference in practice, and thus use a Gaussian for consistency with the other parameters.

It is useful to explicitly consider the random variable

$$Y_{ij} = T_{ij}(l_i) - T_{ji}(l_j), \tag{2}$$

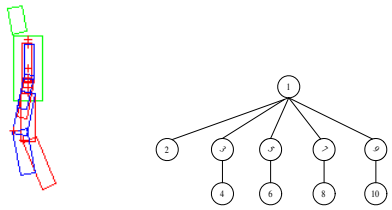

Figure 1: *Tree model for a side-view of a walking person, and a visualization of the model's default pose configuration.*

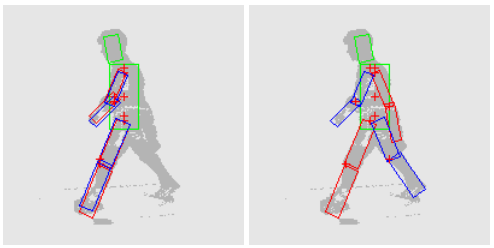

Figure 2: *Two pose configurations: 1) legs and arms all to the same side of the torso, 2) legs and arms symmetric about the torso. Both configurations have almost the same probability using a kinematic tree model.*

which measures the deviation of two parts from their ideal relative location in (x, y, s, o) space. This simplifies the expression for ϕ_{ij},

$$\phi_{ij}(l_i, l_j) = N(Y_{ij}, 0, \Sigma_{ij}),$$

making explicit that the clique potential has the form of a Gaussian distribution over the relative locations of the two parts Y_{ij}.

Figure 1 shows an example of a tree-structured model for a side view of a person walking. This model was learned from labeled training data using the method in [4] which is based on finding a minimum spanning tree (MST). The connections between the parts as well as the potential functions for each edge were learned from the data. Note that the tree captures the kinematic structure because the parts connected by joints are the most highly correlated, not because any specific tree was imposed a priori. The left side of the Figure shows the parts of the model at the mean relative configuration with respect to the root part, which is the torso. The right side of the Figure shows the tree structure.

Not only does this tree model naturally capture the kinematic structure, the factorization of the prior into a product of pairwise clique potentials in equation (1) allows for inference to be done in time linear in the number of nodes. Moreover, the form of the clique potentials allows the methods in [4] to be used to perform estimation in $O(nh)$ rather than $O(nh^2)$ time, where h is the number of discrete locations for each of the n parts.

However this model of a person also illustrates some limitations of tree-structured models. The locations of sibling parts are independent when conditioned on their parent. For instance, given a location for the torso, the locations of the upper arms and legs are independent. In general for an undirected graphical model, conditional independence is equivalent to reachability in the graph (e.g., see [5, 16]). If we remove node 1, the torso, then its children are all unreachable from one another and thus conditionally independent. Therefore it is not possible to directly represent coordination between limbs that are not connected by joints, such as the fact that in a side view the arms and legs should be symmetric about the torso.

Figure 2 illustrates this limitation. The Figure shows two configurations of a side-viewed body model. Clearly the one on the left is un-natural, as the parts are all on one side of the torso, whereas the one on the right is natural. However these two poses have essentially the same probability with regard to the tree-structured model in Figure 1 because it only encodes kinematic relations. It should be noted that this is a limitation in the spatial model, $p(L)$, which is missing important information about limb coordination. For instance, while other information such as a richer part appearance model than a silhouette might also be useful, even then one could not in general distinguish the two legs from one another. Thus we turn to additional sources of spatial constraint by considering the residual coordination between limbs given the tree model.

4. Residual Covariance Analysis

In order to identify possible additional spatial relations among parts we consider the residual covariance of the parts given the locations of their parents. Parts whose locations are highly correlated given the locations of their parents are parts that violate the conditional independence assumption of the tree model, and are thus good candidates for additional spatial constraints in the model. The simplest case is for parts that share a common parent and we use that case to illustrate the approach. In the human body model in Figure 1 the only nodes with a common parent are the upper arms, upper legs and head (nodes 2,3,5,7,9) which all have the torso (node 1) as parent. Thus we consider which of these nodes, if any, have substantial correlation in their locations given a fixed location of the parent.

The random variable Y_{ij} defined in equation (2) measures the degree to which two parts are at their mean relative locations (i.e. larger values correspond to more deformation). Thus given a parent node r and its children $u_1, \ldots, u_k$ we compute the covariance matrix Σ of the Y_{ri} for each u_i and then consider the correlation coefficients for

that covariance matrix. Note that in the more general case of multiple parents, this can be done with respect to all of the parents rather than a single parent.

As an illustration we consider the case of side-views of a person walking. We use 240 labeled silhouette images as training examples. Given the tree-model learned in the previous section, as shown in Figure 1, the correlation matrix is computed for the five children of the torso. The correlations of the position and scale parameters for these parts are not statistically significant (that is, conditional independence is a reasonable assumption). However the correlations of the orientation parameters are highly statistically significant for the four upper limbs (that is, conditional independence is a poor assumption). The portion of the correlation matrix related to the orientation variable is shown in Table 1.

	Head	L. Arm	L. Leg	R. Arm	R. Leg
Head	1.00	0.00	-0.00	-0.06	0.00
L. Arm	0.00	1.00	-0.58	-0.83	0.67
L. Leg	-0.00	-0.58	1.00	0.61	-0.43
R. Arm	-0.06	-0.83	0.61	1.00	-0.59
R. Leg	0.00	0.67	-0.43	-0.59	1.00

Table 1: *The correlation coefficients of the orientation parameters for the 5 parts connected to the torso from 240 side-view images of a person walking.*

All of the entries in this table are highly statistically significant except those for the head. If one were to encode these relations as additional constraints in the graphical model, one would end up with the structure in the top of Figure 3, where the nodes for the upper arms and legs are connected to one another. In the more general case for nodes that do not have a common parent, all the parents would be part of the clique.

When there is a common parent the graph $G = (V, E_G)$ is triangulated (there are no minimum cycles of length more than 3). In the general case the graph can easily be triangulated, if it is not already, by adding edges to the subtree beneath their common ancestor. For a triangulated graph $p(L)$ can be factored into the ratio of a product over maximal cliques and a product over separators (for more details see, for example, [5, 16]). Note that the maximal cliques are those cliques that cannot be made any larger by adding more nodes. The separators are the nonempty intersections between pairs of maximal cliques. If $\mathcal{C}$ denotes the maximal cliques of G and $\mathcal{S}$ the separators then $p(L)$ factors as,

$$p(L) \propto \frac{\prod_{C \in \mathcal{C}} \phi_C(L_C)}{\prod_{S \in \mathcal{S}} \phi_S(L_S)}, \tag{3}$$

where $\phi_C(L_C)$ are clique potentials for the cliques, $\phi_S(L_S)$ are clique potentials for the separators, and the potential functions are properly defined as proportional to the marginal probability of the corresponding clique or separator. Recall from above that the notation L_C denotes the location variables of the nodes $C \subset V$.

As was the case for the tree models, the denominator of (3) can dropped because the separators all contain a single node and the priors over individual nodes are uniformly distributed (uninformative), yielding

$$p(L) \propto \prod_{C \in \mathcal{C}} \phi_C(L_C). \tag{4}$$

Moreover as in the tree model, each clique potential is naturally defined in terms of the relative locations of the parts. We use the tree structure to provide a parameterization, where part locations are expressed relative to the parent. Let $C = \{r, u_1, \ldots, u_k\}$ where $u_1, \ldots, u_k$ are all children of r. Then the clique potential can be defined over the cross product of the domains of all pairwise random variables Y_{ri}, $1 \leq i \leq k$,

$$\begin{aligned} \phi_C(L_C) &= N((Y_{r1}, \ldots, Y_{rk}), 0, \Sigma_C) \\ &= N(Y_C, 0, \Sigma_C), \end{aligned} \tag{5}$$

where Y_C is shorthand for $(Y_{r1}, \ldots, Y_{rk})$, and Σ_C is the covariance for Y_C. Note that for a two-clique this is the same as the edge potential $\phi_{ij}(l_i, l_j)$ of the tree model used above.

The main drawback of this factorization is that computing the potential function for the 5-clique is not practical, because it involves the cross-product space of four Y_{ri}'s. A common approach is to approximate the computation using pairwise potentials for the edges and loopy belief propagation (LBP). We contrast that approach with ours in the experimental section, and find that LBP does not seem to work well for this problem. We suspect this is because the clique is quite large compared to most problems where LBP has been applied successfully (such as a four-connected grid graph where the maximal cliques are still pairs).

5. Factor Analysis

In this section we investigate the use of factor analysis to identify common factors that account for the residual correlations among parts. Such factors can be added to the graphical model as latent variables, rather than adding explicit dependencies between nodes as described in the previous section. Our main goal is to represent the important information about additional spatial relations between parts in a more computationally efficient manner. As with the kinematic tree model, we want to learn this kind of relation from data rather than imposing it, both as a means of setting parameter values and as a means of validating the underlying intuition.

Consider a clique $C = (r, u_1, \ldots, u_k)$ with the clique potential defined in equation (5), where Σ_C is the covariance of Y_C. We now investigate applying factor analysis to the covariance matrix Σ_C to look for an underlying (hidden) factor that explains the covariance. Factor analysis is a common statistical tool for modeling covariance structure. Similar to principal components (PCA), it uses a small number of variables to model high dimensional data and its covariance matrix. However, PCA only reduces the dimension, whereas factor analysis further tries to explain the correlation between different components using a common factor.

The model usually used in factor analysis is

$$Z = \mu + AX + e,$$

where Z is a n dimensional observation vector, X is a m dimensional ($m < n$) vector of unobservable variables called the common factor, and A is a $n \times m$ matrix of factor loadings (parameters). The common factor X is assumed to be independently and identically distributed as $N(X, 0, I_m)$(I_m is a m dimensional identity matrix here), independently of the errors e, which are assumed to be independently and identically distributed as $N(e, 0, D_e)$, where $D_e = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$. The σ_i^2 are called the uniquenesses. Thus conditional on $X = x_0$, the random variable Z is independently distributed as $N(Z, \mu + Ax_0, D_e)$. Unconditionally, Z is independently distributed according to a normal distribution with mean μ and covariance matrix $AA^T + D_e$. From now on, we will assume $\mu = 0$ and ignore it, since we can shift the data to zero its mean position.

A set of parameters $\{A, D_e\}$ needs to be learned for the factor analysis, given the training data set $\{z_1, z_2, ...z_t\}$. There exists an EM algorithm to learn these parameters, where $\{z_1, z_2, ...z_t\}$ is considered the incomplete data set, and $\{z_1, x_1; z_2, x_2; ...z_t, x_t\}$ is considered as the complete data set (for more details see [8]).

Given the factor analysis for a clique $C = (r, u_1, \ldots, u_k)$, we know that for a particular value $X = x_0$ of the common factor, the random variable Y_C is independently distributed as a Gaussian function $N(Ax_0, 0, D_e)$, where D_e is a diagonal matrix. That is, conditional on the common factor, the high dimensional multivariate Gaussian clique potential $\phi_C(L_C)$ can be factored into a product of independent Gaussian functions over the Y_{ri}'s,

$$p(L_C|X) \propto \phi_{C|X}(L_C) = \prod_{i=1}^{k} N(Y_{ri} - \lambda_i X, 0, \sigma_i^2), \quad (6)$$

where λ_i is the i^{th} row vector of factor loading matrix A.

When the clique involves children of a common parent we can rewrite this clique potential as a product over certain edges of the original tree $T = (V, E_T)$, by noting that each term of the product in (6) corresponds to an edge between a child u_i and the parent r, and moreover

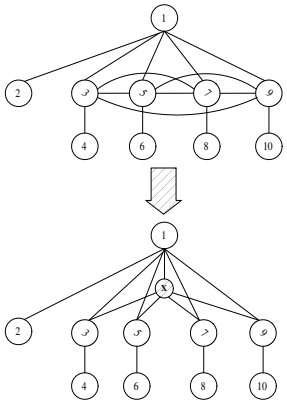

Figure 3: *Introducing the common factor variable into the graphical model breaks the large clique into several 3-cliques by augmenting the original tree with a common factor vertex (labeled "X").*

these are all edges of E_T. Denote this set of edges by $E_C = \{(v_i, v_j) \in E_T | v_i, v_j \in C\}$.

Then Equation (6) can be rewritten in factorized form based on the tree edges in E_C,

$$p(L_C|X) \propto \prod_{e_{ij} \in E_C} \phi_{ri|X}(l_i, l_j), \quad (7)$$

where

$$\phi_{ri|X}(l_r, l_i) = N(Y_{ri} - \lambda_i X, 0, \sigma_i^2). \quad (8)$$

Letting $E_{\bar{C}} = E_T - E_C$ be the tree edges not in the clique C, we can write the conditional probability $p(L|X)$ in a tree-factored form by considering the partition of E_T into E_C and $E_{\bar{C}}$. Substituting equation (7) into the factorization of $p(L)$ in (4) yields,

$$p(L|X) \propto \prod_{e_{ij} \in E_C} \phi_{ij|X}(l_i, l_j) \prod_{e_{ij} \in E_{\bar{C}}} \phi_{ij}(l_i, l_j). \quad (9)$$

In other words, for a fixed $X = X_0$ the corresponding graphical model is simply the original tree T, however the clique potentials on the edges of E_C are different from the original problem. They are $\phi_{ij|X}$ defined in (8). Adding a latent node for the common factor X into the tree T requires connecting that new node to every node that was in the clique C, because in the factorization in equation (7) there is a dependency between X and every node in C (or more precisely between X and every pair of nodes corresponding to an edge in E_C). We call the resulting graph the common-factor graph $F = (U, E_F)$, where $U = V \cup \{X\}$ and $E_F = E_T \cup \{(X, v_i) | v_i \in C\}$. This replacement of the clique in G by the latent variable structure in F is illustrated in Figure 3.

6. Inference Methods

The posterior distribution $P(L|I)$ of object configurations given an image I is commonly used for estimating the pose of a model with respect to the image. By Bayes' rule

$$P(L|I) \propto P(I|L)P(L),$$

which is the product of the likelihood of observing the image given location L and the prior over locations. It is generally assumed that the likelihood factors into a product of functions, one for each part of the model,

$$P(I|L) \propto \prod_{v_i \in V} \psi_i(I, l_i).$$

We use a simple likelihood model from [4] that measures the degree to which each part overlaps the binary silhouette data.

In this paper we have considered three forms of prior $P(L)$, the tree in equation (1), the graph with the large clique in equation (4) and the common-factor graph in equation (9). For each of these three factorizations of the posterior $P(L)$, we consider the problem of finding an optimal configuration of the parts by MAP estimation,

$$L^* = \arg\max_L \prod_{v_i \in V} \psi_i(I, l_i)P(L).$$

The computational difficulty of this MAP estimation problem depends on how the prior factors. For the tree-structured graph, T, the MAP estimation problem can be solved in $O(nh)$ time, where n is the number of parts and h is the number of possible locations of each part (using the methods in [4]). For the graph with a 5-clique, exact solutions to the MAP estimation problem are prohibitively slow taking time $O(nh^5)$ (see [5, 16]). However it is common to do approximate inference on graphs with cycles using loopy belief propagation (LBP). This has been done for object recognition (see [2]).

For the common-factor graph, F, an optimal configuration is given by

$$\langle L, X \rangle^* = \arg\max_{\langle L,X \rangle} p(L, X)p(I|L)$$

Since $p(L, X)$ factors into $p(X) \cdot p(L|X)$ we can compute

$$\arg\max_L p(L|X)p(I|L) \tag{10}$$

for each X and then maximize over X. Moreover, from equation (9), $p(L|X)p(I|L)$ factors into a tree, so standard dynamic programming methods can be used to efficiently compute (10). The maximization over X simply involves trying the h_x possible discrete values of X, for an overall running time of $O(h_x nh)$. Note that for the models developed here X ranges over possible orientations $[-\pi, \pi]$, and thus a reasonable discretization results in values of h_x that are fairly small.

7. Experimental Results

We learned common-factor models for three different kinds of images, one for a side-view of a person walking, one for a 45-degree view of a person walking, and one for a person dancing. In each case the (labeled) training data for learning the model was a different set of images than the ones used for doing pose estimation. For each of the three kinds of images we learned three different types of models, one using a tree, one using a common-factor graph and one using a graph with a large clique. Our main interest in these experiments is in comparing the pose estimation accuracy of these three types of models for various kinds of images. Thus we use a simple appearance model that measures the degree of overlap of a part with silhouette data.

As described above the learning process consists of three stages. First a kinematic tree structure is learned using the minimum-spanning tree method of [4], illustrated in Figure 1. Second, covariance analysis is used to identify parts that violate the independence assumption of the tree model. These parts form the model with a large clique (where approximate pose is estimated using max-product LBP). Third, factor analysis is used to find a common factor that models the clique, and a latent variable is introduced in place of the clique, as illustrated in Figure 3.

For the side-view walking model we trained the model using 240 labeled frames from CMU's HumanId side-view walking sequence. For testing we used 50 frames from the Brown sequence in [13], for which there is ground truth from motion capture. The ground truth gives the location of markers, which can easily be related to the parts of our models because they are at joints between parts or part centers. To generate silhouettes all images are background subtracted and normalized to a size of 200 by 200. For this data the large clique consists of the upper arms and legs and the resulting common factor X is a 1D random variable in the orientation dimension with Gaussian distribution $N(0, 1)$, and with the loading vector as $A = (0.9426, -0.8055, -0.9432, 0.8152)^T$. The common factor has a simple interpretation as the "swing" angle of the arms and legs during walking, and the loading matrix reflects the use of the limbs for balance related to that variable.

We used the three types of side-view model (common-factor, tree and large clique) learned from the CMU data to estimate poses for the first 50 frames of the Brown data. Note that the model is quite generic, being generated with data from a different person than appears in the test set. For each test image and for each of the three models we find the MAP estimate of the pose. While the exact (discrete) MAP estimate is computed for the tree and the common-factor models, for the model with the large clique only an approximate estimate can be computed (using max-product LBP). To evaluate the accuracy of the models we used the 15

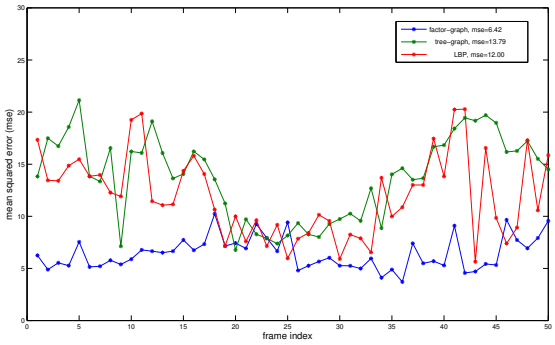

Figure 4: *Mean squared error of the joint location for each frame for the three estimation techniques, compared to the ground truth MOCAP data.*

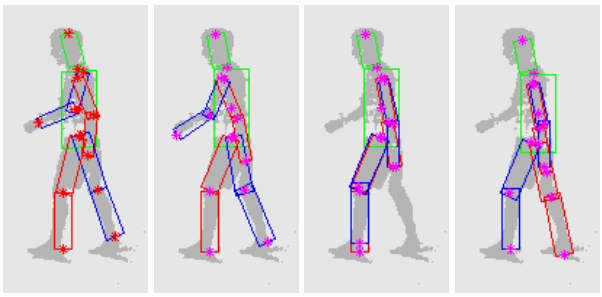

Figure 5: *Illustration of MAP pose estimation accuracy for the three methods: 1) ground truth (MOCAP), 2) common-factor model, 3) tree model, 4) LBP for model with large clique.*

	shoulder	elbow	wrist	hip	knee	ankle
Factor	4.8	5.5	8.6	4.2	4.4	5.4
Tree	9.1	11.1	19.4	6.4	6.6	28.6
LBP	9.9	11.9	20.5	6.4	5.3	20.5

Table 2: *Average error by marker (see text).*

marker positions indicated by stars in Figure 5 to compare the pose estimation results for each model with the ground truth (MOCAP). Figure 4 shows the mean squared error of the estimated marker locations compared to the true locations for each image frame. Note that for single-viewpoint silhouette data there is an unresolvable left-right ambiguity, so we switch the left and right limbs and use the one with smaller error in each case.

Overall the average errors are 6.42, 13.79 and 12.00 pixels for the common-factor, tree and large clique models (with standard deviations of 1.55, 3.99 and 3.99 respectively). Thus we see that the common-factor model has about half the pose estimation error of the other two models. These results support the assumption that coordination between limbs beyond the kinematic structure is highly important for pose estimation. Figure 5 shows one of the image frames with the ground truth joint positions and the results from the three models, illustrating some typical pose errors. Note that between frames 20 and 30 the three methods have almost the same performance. This is not surprising because these are the frames where the arms and legs overlap and are nearly vertical, where the kinematic tree model works quite well.

Approximate inference for the large-clique model (using max-product LBP) yields results more similar in accuracy to the tree model than to the common-factor model. This is in contrast to many other applications of LBP, including for recognition (e.g., [2]). One difference that may explain the relatively poor performance of LBP is that here the model has one quite large clique, whereas in other applications the cliques are quite small. For instance in [2] the clique size is two.

We can also consider how the error varies by body part. Table 2 shows the mean error for twelve of the fifteen markers, averaging errors for the left and right sides together. This illustrates that the largest improvement comes from the extremities (wrist and ankle). In fact for the three markers not shown in the table, the torso, neck and head, the error of all three methods is similar. Intuitively, the extremities have the most positional freedom in a tree model. The common-factor model constrains the upper limbs more tightly, and this in turn increases the accuracy in the extremities.

In addition to the Brown sequence, we consider some data without ground truth to provide a qualitative evaluation of the common-factor model. The second dataset is a 45-degree view of a walking person from CMU's humanID database. We show some results using the common-factor model in Figure 6. The other two models produce similar kinds of pose estimation errors to that seen in the side-view walking sequence in Figure 5.

The third dataset contains snapshots from a frontal-view

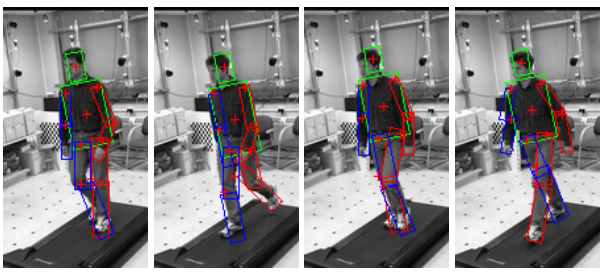

Figure 6: *MAP results for the common-factor model on several images of a 45-degree view of a person walking.*

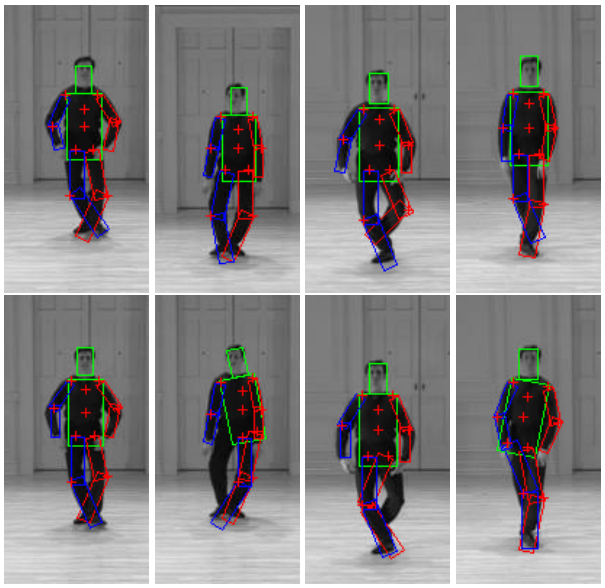

Figure 7: *Comparison of pose estimation results for a dance image: (Top) common-factor model, (Bottom) tree model.*

sequence of a person dancing, again without ground truth. Figure 7 shows some frames contrasting the common-factor model and the tree model. The top row displays the pose estimation results using the common-factor model, while the bottom shows the results on the same frames using the tree model. The balance constraint is quite different here than for the side or 45-degree walking views, but there is a similar improvement in results for the common-factor model compared to the tree model.

To help visualize the 2D pose estimation results we have also composed some videos showing the MAP pose estimates that were computed for each frame. These videos are included in the supplemental materials. For the dance sequence there is no ambiguity about the left vs. right side, whereas for the other sequences there is an ambiguity. This ambiguity is resolved by simple temporal continuity, choosing the left vs. right configuration that is most consistent with the previous frame. The videos suggest that with some extensions this technique could also be used for person tracking. However it would be important to add some temporal constraints (e.g., using a linear dynamical system) in order to smooth out the estimated part locations on successive frames.

8. Conclusion

In this paper we extend tree-structured kinematic models so as to model residual correlations in locations of the parts, thereby capturing constraints such as balance and coordination of the limbs. To achieve tractable inference we use factor analysis to model the covariance matrix for limbs whose orientations are highly correlated after accounting for kinematic constraints. By introducing the common factor as a latent variable into the graph we are able to apply standard dynamic programming techniques to perform exact (discrete) inference with relatively low computational cost.

The additional spatial constraints allow us to better capture the dependency among parts. The experimental results illustrate that the common-factor model yields better pose results than a tree, more than halving the estimation error for a set of images that have ground truth.

References

[1] C. Bregler and J. Malik. Tracking People with Twists and Exponential Maps. CVPR 1998.

[2] J.M. Coughlan and S.J. Ferreira. Finding Deformable Shapes Using Loopy Belief Propagation. ECCV 2002.

[3] J. Deutscher, A. Blake, and I. Reid. Articulated Body Motion Capture by Annealed Particle Filtering. CVPR 2000.

[4] P.F. Felzenszwalb and D.P. Huttenlocher. Efficient Matching of Pictorial Structures. CVPR 2000.

[5] C. Huang and A. Darwiche. Inference in Belief Networks: A procedural guide, Intl. J. Approximate Reasoning, 1996.

[6] S. Ioffe and D.A. Forsyth. Mixtures of Trees for Object Recognition. CVPR 2001.

[7] S. X. Ju, M. J. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated motion. 2nd Int. Conf. on Automatic Face- and Gesture-Recognition, 1996.

[8] G.J. McLachlan and T. Krishnan. The EM Algorithm and Extensions. Wiley Series in Probability and Statistics, 1997.

[9] D. Ramanan and D.A. Forsyth. Finding and Tracking People from the Bottom Up. CVPR 2003.

[10] D.M. Gavrila. The Visual Analysis of Human Movement: A Survey. CVIU 1999.

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

[12] G. Elidan, I. Nachman and N. Friedman, "Ideal Parent" Structure Learning for Continuous Variable Networks. UAI 2004.

[13] L. Sigal, S. Bhatia, S. Roth, M.J. Black, and M. Isard. Tracking Loose-Limbed People. CVPR 2004.

[14] Y. Song, L. Goncalves, P. Perona. Unsupervised Learning of Human Motion. PAMI, Volume 25, 2003.

[15] K. Toyama, A. Blake. Probabilistic Tracking in a Metric Space. ICCV 2001.

[16] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. UC Berkeley, Dept. of Statistics, Technical Report 649. September, 2003.