Part-Based Pose Estimation with Local and Non-Local Contextual Information

Ming Chen[a], Xiaoyang Tan[a,*]

[a]*Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China*

Abstract

In this work we propose a new method for part-based human pose estimation. The key idea of our method is to improve the accuracies for leaf parts localizations - an issue that is largely ignored by previous work - by incorporating both local and non-local contextual information into the model. In particular, we use the local contextual information to reduce or eliminate the influences of the noises, while the non-local contextual information helping to improve the detection accuracies of the leaf parts. Since more accurate parts localizations usually mean a more reasonable active set of spatial constraints, this potentially enhances the effectiveness of the subsequent optimization procedure. Furthermore, we keep the basic structure of tree-based model, hence taking the advantage of its conceptual simplicity and computational efficient inference. Our experiments on two challenging real-world datasets demonstrate the feasibility and effectiveness of the proposed method.

Keywords: part-based mixture model, human pose estimation, contextual information

1. Introduction

The human pose estimation is a very popular problem in computer vision. It has many practical applications in intelligent surveillance, image understanding, action analysis and intelligent human-computer interface. The pose estimation task is to identify the human in an image and to determine each body part's position and orientation relative to some coordinate system. However, due to the following reasons, the pose estimation problem becomes challenging: (1) variability of human visual appearances; (2) intense changes in human body postures; (3) variability of background in images; (4) occlusions due to self-articulation and layering of objects.

There are many approaches [1, 2, 3, 4, 5] working on this task. They can be roughly divided into two categories: the exemplars based methods and the part-based methods. The exemplars based methods [3, 4] learn the human pose exemplars from the training sets and do pose estimations by matching these exemplars to the input images. But due to the fact that the number of human poses is tremendous, the amount of exemplars needing to learn is enormous. Hence their performance tends to be limited by the number of the learned exemplars or the size of the training set. On the other hand, the part-based methods [1, 5, 6] bypass these difficulties by decomposing the whole human body into many parts (the feet, hands, arms, legs, torso, etc.) and model each part separately. The spatial constraints between each part are then utilized to filter the noisy responses of the individual part detectors, and it is therefore critical to design good spatial constraints so that it can capture a versatile and plausible set of poses. The part-based methods can characterize much more human poses than the exemplars based methods from the limited training samples, and are useful to deal with the curse of dimensionality or the occlusion problem. Due to these advantages, they have been widely used in the field of computer vision[7, 5, 8].

The part-based models (PBM) can be dated back to the generalized cylinder model of Binford [7]. Among others, one of the most influential works is given by Felzenszwalb and Huttenlocher [5], in which a tree model is adopted for efficient inference. However the tree model could be plagued by the well-known phenomena of double-counting, where the two estimated limbs cover the same region of the image. In addition, they may also be affected by the noises in background, and it cannot model the changes of postures as much as we may think. The loopy models [9] are extensions of the tree model. They apply more structural constraints on the parts, which help to address the double-counting problem and the occlusion problem. Since the resulting relational graph of the model no longer satisfies the tree constraints, less efficient inference strategies, such as loopy belief propagation [10], importance sampling [5, 11], or the iterative approximations [12] have to be adopted. Alternatively, the constellation models [13, 14, 15] take a strategy of using a sparse set of parts defined at the location of keypoints and modeling their relationship very loosely. Body plans [8] are another representation, in which particular geometric rules are encoded to shape the concept of valid deformation of the local templates.

Recent works have suggested that mixtures of part models are promising ways to enhance the capability of PBM to represent human body with various poses [12, 16, 17]. For example, a global mixture pictorial structure is adopted in [17] and it is shown that a rich number of postures can be reliably captured using this. However, such improvement is at the cost of the in-

*Corresponding author: Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Yudao Street 29, Nanjing 210016, China. Tel: +86-25-8489-2956; Fax: +86-25-8489-2452;

Email addresses: chenming@nuaa.edu.cn (Ming Chen), x.tan@nuaa.edu.cn (Xiaoyang Tan)

creased number of model parameters, which implies the need of either a large enough training dataset or a good part sharing mechanism to ensure an acceptable generalization capability of the model, and that the computational cost to estimate them could be very expensive as well.

To hand these issues, a method named flexible mixtures-of-parts (FMP) is proposed [6], in which each part rather than the entire body is modeled by a mixture. This provides a simple way to characterize a large amount of poses. However, its performance can be easily affected by the noises inherent in the images or the low accuracies of leaf parts detectors. As shown in Figure 2, many failures of this model are due to the incorrect localizations of leaf parts. Actually, as illustrated in Figure 4, the leaf parts detectors usually have the most unreliable outputs compared to other body parts due to the complexity of appearance and the limited information available in a local region. Indeed, the failure detections of leaf parts may make the pairwise constraints between parts fail to capture the complex spatial characteristics in the pose space. Alternatively, [18] incorporates the hierarchical structure with latent nodes and mixture parts to introduce high-order relationship among parts into the model, which allows it to capture an exponential number of plausible poses. But this model is still a tree model, and has the same defects as [6]. The method in [19] adopts the contour-based features with the spatial relationship to encode the structural constraints, which allow model to be able to evaluate quality of the part connectivity and hence being helpful to filter the local noisy responses.

Based on the above observations, in this work, we propose an extension to the tree model of [6] for pose estimation. The key idea of our method is to improve the accuracies for leaf parts localizations - an issue that is largely ignored by previous work - by incorporating both local and non-local contextual information. In particular, we use the local contextual information to reduce or eliminate the influences of the noises, and use the non-local contextual information to improve the detection accuracies of the leaf parts. Since more accurate parts localizations usually mean a more reasonable active set of spatial constraints, this potentially enhances the effectiveness of the subsequent optimization procedure. Additionally, although many work mentioned above try to improve the performance of pose estimation by adopting various complex models, we keep the basic structure of [6], hence taking the advantage of its conceptual simplicity and computational efficient inference. Our experiments on two challenging real-world datasets demonstrate the feasibility and effectiveness of the proposed method.

In what follows, we first give a brief review of the part-based model in Section 2, then we detail the proposed method in Section 3. The experimental results are presented in Section 4, and we conclude this paper in Section 5.

2. A Brief Review on the Part-Based Model

In this section, we will give a brief review on the part-based model with some discussion.

2.1. Linearly Parameterized Spring Model

Assume we have a K-parts model with T_i mixtures for each part i. Let us write the location of i-th part as $\boldsymbol{l_i} = \{x_i, y_i\}$, and its "type" (e.g., a particular orientation) as t_i ($t_i \in \{1, ..., T_i\}$), and hopefully we would have a corresponding mixture component for it. Then $\boldsymbol{L} = \{l_1, ..., l_K\}$ and $\boldsymbol{T} = \{t_1, ..., t_K\}$ respectively denote a particular configuration of the K parts and their corresponding types. Furthermore, let E denote the edge set of the relational graph $G = (V, E)$ of the K parts, which model the pairwise spatial relationship between parts locations. Note that these notations are consistent with those of [6].

Given an image $\boldsymbol{I}$, we hope to find a measure criteria for each possible configuration which tells us the probability of the configuration to be the real locations of the parts. According to the idea of the part-based model, this mainly includes two parts: the metric for the appearance, and the metric for the deformation or spatial constraints.

Metric for appearance The metric for the appearance measures the similarity of the appearances between the given image patches and the templates of the parts, which can be defined as the following scoring function:

$$A(\boldsymbol{I}, \boldsymbol{L}, \boldsymbol{T}) = \sum_{i=1}^{K} A_i(\boldsymbol{I}, \boldsymbol{l_i}, t_i) = \sum_{i=1}^{K} \boldsymbol{w_i^{t_i}} \cdot \phi(\boldsymbol{I}, \boldsymbol{l_i}) \tag{1}$$

where $\phi(\boldsymbol{I}, \boldsymbol{l_i})$ is the descriptor for part i extracted from a fixed size image patch from location $\boldsymbol{l_i}$ in image $\boldsymbol{I}$, and $\boldsymbol{w_i^{t_i}}$ is HOG filter for part i. $\boldsymbol{w_i^{t_i}} \cdot \phi(\boldsymbol{I}, \boldsymbol{l_i})$ computes the local score of appearance for part i at location $\boldsymbol{l_i}$ by detector t_i. This local scoring is akin to a linear template filter.

Metric for deformation The metric for the deformation measures the distribution of the relative spatial position of the parts. This can be decomposed into a set of spatial constraints on pairs of parts. For example, normally the hands should be next to the elbows but not between elbows and shoulders.

$$D(\boldsymbol{I}, \boldsymbol{L}, \boldsymbol{T}) = \sum_{(i,j)\in E} D_{ij}(\boldsymbol{I}, \boldsymbol{l_i}, \boldsymbol{l_j}, t_i, t_j) = \sum_{(i,j)\in E} \boldsymbol{w_{ij}^{t_i,t_j}} \cdot \Psi(\boldsymbol{I}, \boldsymbol{l_i}, \boldsymbol{l_j}) \tag{2}$$

where the shape encoder $\Psi(\boldsymbol{I}, \boldsymbol{l_i}, \boldsymbol{l_j}) = [dx\ dx^2\ dy\ dy^2]$ (with $dx = x_i - x_j$ and $dy = y_i - y_j$). This models the negative spring energy associated with pulling part i from a typical relative location with respect to part j. The parameters $\boldsymbol{w_{ij}^{t_i,t_j}}$ specify the rigidity of the rest locations of the spring, e.g., some parts may be easier to shift horizontally versus vertically. By assuming a Gaussian distribution of the relative location of two parts, one can describe the relative locations between two neighboring parts by the mean of the Gaussian, and the rigidity between them is specified by the covariance of the Gaussian.

Note that the relative location of part with respect to its parent is only dependent on its part type but not on its parent's type. For example, the hand should lie below the elbow, regardless of the orientation of the upper arm. This observation helps to simplify the above equation (2) to be,

$$D(\boldsymbol{I}, \boldsymbol{L}, \boldsymbol{T}) = \sum_{(i,j)\in E} D_{ij}(\boldsymbol{I}, \boldsymbol{l_i}, \boldsymbol{l_j}, t_i) = \sum_{(i,j)\in E} \boldsymbol{w_{ij}^{t_i}} \cdot \Psi(\boldsymbol{I}, \boldsymbol{l_i}, \boldsymbol{l_j}) \tag{3}$$

Based on the above two metrics, the final scoring function of an arbitrary configuration can be written as follows:

$$S(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) = A(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) = \sum_{i=1}^{K} A_i(\boldsymbol{I},\boldsymbol{l_i},t_i) + \sum_{(i,j)\in E} D_{ij}(\boldsymbol{I},\boldsymbol{l_i},\boldsymbol{l_j},t_i)$$
$$= \sum_{i=1}^{K} \boldsymbol{w}_i^{t_i}\cdot\phi(\boldsymbol{I},\boldsymbol{l_i}) + \sum_{(i,j)\in E} \boldsymbol{w}_{ij}^{t_i}\cdot\Psi(\boldsymbol{I},\boldsymbol{l_i},\boldsymbol{l_j}) \tag{4}$$

2.2. Inference

The score function (4) can be compactly rewritten as a function of the part appearance and the spatial parameters:

$$S(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) = \boldsymbol{w}\cdot\Phi(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) \tag{5}$$

where $\boldsymbol{w}$ is a concatenated vector of templates $\boldsymbol{w_i}$ and the spatial parameters $\boldsymbol{w_{ij}}$, while $\Phi(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T})$ is a concatenated sparse vector of the HOG descriptors $\phi(\boldsymbol{I},\boldsymbol{l_i})$ and the relative offsets $\Psi(\boldsymbol{I},\boldsymbol{l_i},\boldsymbol{l_j})$. Assuming that we have learnt the parameters $\boldsymbol{w}$ from the annotated training samples (using a large margin criterion), for a test image $\boldsymbol{I}$, we must infer the locations of each part and its type from it. This corresponds to the maximization of equation (4) over $\boldsymbol{L}$ and $\boldsymbol{T}$.

Fortunately, when the relational graph $G = (V, E)$ is a tree, this inference can be done efficiently with dynamic programming. Write $kids(i)$ as the set of children of part i in G. One can use the following equations to compute the message part i passes to its parent part j:

$$score_i(\boldsymbol{I},\boldsymbol{l_i},t_i) = \boldsymbol{w}_i^{t_i}\cdot\phi(\boldsymbol{I},\boldsymbol{l_i}) + \sum_{k\in kids(i)} m_k(\boldsymbol{I},\boldsymbol{l_i},t_i) \tag{6}$$

$$m_i(\boldsymbol{I},\boldsymbol{l_j},t_j) = \max_{t_i}(\max_{\boldsymbol{l_i}}(score_i(\boldsymbol{I},\boldsymbol{l_i},t_i) + \boldsymbol{w}_{ij}^{t_i}\cdot\Psi(\boldsymbol{I},\boldsymbol{l_i},\boldsymbol{l_j}))) \tag{7}$$

The equation (6) computes the local score of part i at pixel position $\boldsymbol{l_i}$ by collecting the massage passed from its children. The equation (7) computes the message passed from part i to pixel position $\boldsymbol{l_j}$ of part j type t_j, and the parameters of the maximization indicate the best scoring location of part i respect to position $\boldsymbol{l_j}$ of part j. Once the message is passed to the root part (j=1), $score_1(\boldsymbol{I},\boldsymbol{l_1},t_1)$ represents the scores of the best scoring configuration for each root position and type. After this, the best scoring locations and types for each part can be collected by backtracking. The overall pipeline of inference is illustrated in Figure 1.

This model is very successful on several challenging datasets [6] but it still could be confused by the noises, occlusions, and so on. Figure 2 illustrates some typical failure cases among others based on the implementation of [6]:

1. The influence of noises. The estimation of the parts can be easily affected by the noises in the image. As shown in the images of the second row of Figure 2(a), some of the parts are estimated inaccurately and sometimes the whole body could be completely drift to the background.
2. Suboptimal estimation. As shown in Figure 2(b), poses estimated with the highest score may not be the best one. That's to say, the suboptimal estimation could be actually better. This implies that the tree-structured model is oversimplified since any high-order interleaving among parts is ignored completely.

(a) (b) (c)

Figure 2: Illustration of the typical failure cases of the mixtures of parts model [6]: (a) the influence of noises - the two images in the first row show the ground truth parts locations, and the two images in the second row give the corresponding estimation by the model; (b) suboptimal estimation - the pose estimation by the image shown in the first row is better than that in the second row, but with a lower score of (−0.85), compared to a score of (−0.71) of the second row; (c) double counting.

3. Double counting. Another common limitation of the tree models is the double-counting phenomena, as shown in Figure 2(c). That is because the parts are located independently, and the appearances between the left and the right limbs are not discriminative enough. As a result, two body parts are located into the same image area.

All the above issues are more or less related to the message passing mechanism of the tree model. As one can see from Figure 1, almost all the information that can be used for pose estimation is from the response maps of each part detectors, while the tree model passes the detection information of lower body parts to upper parts, and uses this information to improve the detection performance of upper parts. However, since the information of the lower parts is inherently inaccurate, the error could be accumulated during the propagation procedure and hurts the performance in the end. We try to address this issue by improving the localization accuracies for leaf parts through incorporating both local and non-local contextual information into the model, as detailed in the next section.

3. The Proposed Method

As mentioned before, the specific information available to each part detector is actually very limited and thus is easily confused with that of other parts. An obvious way to compensate for this is to use contextual information, i.e., any information that is relevant to the part of interest. In this section, we will give a detailed description on the contextual information that we adopted.

3.1. Local Contextual Information

The first type of contextual information is based on the human kinematic principle. That is, the spatial relationship between some parts in a human body are always more stable than

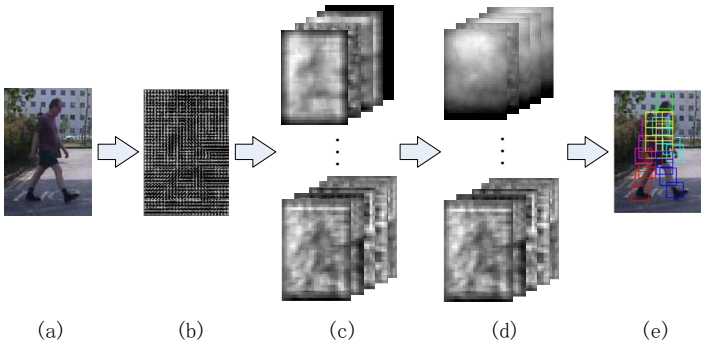


Figure 1: The pipeline of pose estimation of the part-based model. From left to right, for a given image (a), its feature map (b) is first extracted, then a sliding window method is applied on it to yield a set of response maps (c). Afterwards, based on a predefined tree structure, a message propagation algorithm is used to sharpen the response map (d) and give the final detection results (e).

others. For example, the hand, the central lower arm and the elbow are near to each other and move together. These three parts can be thought as a group which provides local contextual information to each of the component. One can use a new "lower arm" part detector to exploit this information.

Figure 3 (a) gives all the augmented local contextual parts used in this work. Totally 11 new "bigger parts" are added to the original tree model, such as the head, upper arms, lower arms, and so on. Let V_{ext} denote this set of "bigger parts" and A_{ext} their scores. Then our scoring function can be written as:

$$\begin{aligned} S(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) =& A(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + \epsilon_1 \cdot A_{ext}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) \\ =& \sum_{i=1}^{K} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i) + \sum_{(i,j)\in E} D_{ij}(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j,t_i) + \epsilon_1 \cdot (\sum_{i\in V_{ext}} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i)) \\ =& \sum_{i=1}^{K} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i) + \sum_{(i,j)\in E} \boldsymbol{w}_{ij}^{t_i} \cdot \Psi(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j) + \epsilon_1 \cdot (\sum_{i\in V_{ext}} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i)) \end{aligned} \tag{8}$$

where $\epsilon_1 (0 \le \epsilon_1 \le 1)$ is the noise suppression parameter (set as $\epsilon_1 = 0.5$ in our experiments).

We can further write the equation (8) in the form of (5), and hence the same strategy can be used to learn the parameters. For testing, we first fuse the responses of the "bigger parts" detectors with that of the corresponding parts, and then use the dynamic programming method described in Section 2.2 for inference.

3.2. Non-Local Contextual Information

The second type of contextual information is inspired by the massage propagation mechanism discussed in Section 2.2. As Figure 4 shows, the accuracy of each part is improved significantly after the message propagation. Actually, the message propagation along the tree model can be thought of as an accumulation of contextual information from leaves to root: the higher the parts are in the model, the more information they can use. This partly explains why the parts in the higher levels of the model have higher accuracies than those in the leaves, as revealed in Figure 4 (b).

However, this mechanism of contextual information propagation doesn't help the parts in the lower levels as much as those in the higher levels due to the predefined message passing route.

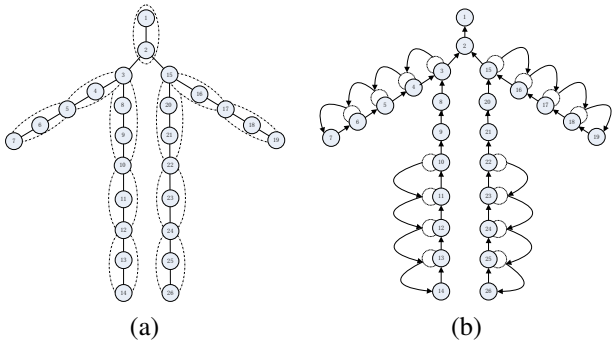
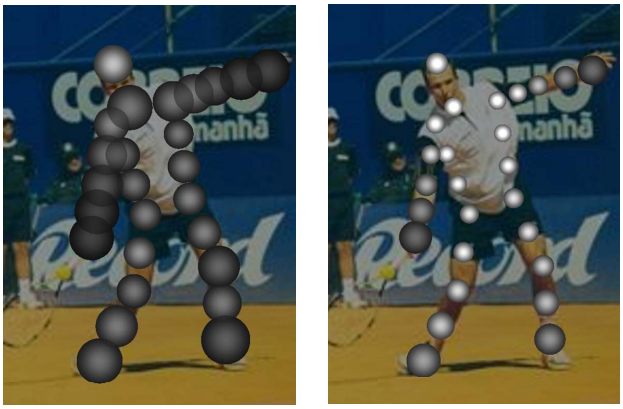


Figure 3: (a) The body parts used in our model - besides those parts in [6] (marked as circles), we also add extra local contextual parts (marked as dotted ellipses). (b) The path of the message transmission used in our model. The dotted circles denote the virtual parts, and the directed arcs represent the directions of the message transmission.

In addition, errors in the lower parts can be propagated to the upper parts and thus hurts the overall performance. Therefore, it would be helpful if we can improve the performance of the leaf nodes before the normal message propagation starts.

To this end, we add a new message passing route for each leaf node that collects non-local contextual information. Figure 3(b) illustrates the idea, where several virtual parts (marked as dotted circles) which mirror their corresponding parts are used to pass their detection information to the leaf parts. The relational graph of our model is still a tree, and we can use the same inference as [6]. Let us denote V_{vir} as the set of the virtual parts and E_{vir} as the set of edges connected to the virtual parts. The scoring function of our model is as follows:

$$\begin{aligned} S(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) =& A(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + \epsilon_2 \cdot (A_{vir}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D_{vir}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T})) \\ =& \sum_{i=1}^{K} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i) + \sum_{(i,j)\in E} D_i(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j,t_i) + \epsilon_2 \cdot (\sum_{i\in V_{vir}} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i) + \sum_{(i,j)\in E_{vir}} D_{ij}(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j,t_i)) \\ =& \sum_{i=1}^{K} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i) + \sum_{(i,j)\in E} \boldsymbol{w}_{ij}^{t_i} \cdot \Psi(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j) \\ &+ \epsilon_2 \cdot (\sum_{i\in V_{vir}} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i) + \sum_{(i,j)\in E_{vir}} \boldsymbol{w}_{ij}^{t_i} \cdot \Psi(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j)) \end{aligned} \tag{9}$$



(a) (b)

Figure 4: Visualization of the detection accuracy of each part using in the FMP model [6]: a ball is put on each part's location with its size in proportion to the variance of error and lightness to the average error (i.e., a small and light ball indicates a good part detector). The original accuracies of the body part detectors are shown in (a) and the accuracies after message propagation is shown in (b).

while $\epsilon_2(0 \leq \epsilon_2 \leq 1)$ is the noise suppression parameter. And when training, we set $\epsilon_2 = 1$ via cross validation. Again, this score function can be reformulated in the same form as (5) for parameters learning.

3.3. The Algorithm

Combining the local and non-local contextual models described in the previous two sections, we get the following model:

$$\begin{aligned} S(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) =& A(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + \epsilon_1 \cdot A_{ext}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + \epsilon_2 \cdot (A_{vir}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T}) + D_{vir}(\boldsymbol{I},\boldsymbol{L},\boldsymbol{T})) \\ =& \sum_{i=1}^{K} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i) + \sum_{(i,j)\in E} D_i(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j,t_i) + \epsilon_1 \cdot (\sum_{i\in V_{ext}} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i)) \\ &+ \epsilon_2 \cdot (\sum_{i\in V_{vir}} A_i(\boldsymbol{I},\boldsymbol{l}_i,t_i) + \sum_{(i,j)\in E_{vir}} D_{ij}(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j,t_i)) \\ =& \sum_{i=1}^{K} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i) + \sum_{(i,j)\in E} \boldsymbol{w}_{ij}^{t_i} \cdot \Psi(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j) + \epsilon_1 \cdot (\sum_{i\in E_{ext}} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i)) \\ &+ \epsilon_2 \cdot (\sum_{i\in V_{vir}} \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i) + \sum_{(i,j)\in E_{vir}} \boldsymbol{w}_{ij}^{t_i} \cdot \Psi(\boldsymbol{I},\boldsymbol{l}_i,\boldsymbol{l}_j)) \end{aligned} \tag{10}$$

To implement this model, we add two processing steps between step (c) and (d) of the flowchart in Figure 1, and use the same method of [6] for training and inference. In particular, the local contextual information from the responses of the "bigger parts" detectors is firstly used to revise the estimation of the corresponding part model, and then the non-local contextual information is exploited to enhance the responses of leaf parts. Except these, the remaining parts of our model are kept the same as those of [6] (i.e., still a tree-structured model) and hence the inference can be done very efficiently. The overall inference procedure is summarized in Algorithm 1.

We calculate multi-scale features for a given input image and through the above steps, we get a max scoring estimation for each scale, and the one with max score is chosen from all the estimations as the final result.

Algorithm 1 Algorithm for pose estimation of our model

Input:

The image $\boldsymbol{I}$ to be estimated.

Output:

The estimated bounding boxes $\boldsymbol{B}$ for each body part and its corresponding maximum score *score*.

1: Extract the feature maps $\phi(\boldsymbol{I})$ from the given image $\boldsymbol{I}$.
2: For each part detector $\boldsymbol{w}_i^{t_i}$, search the feature map to get the response maps $\boldsymbol{R}_i^{t_i}$, $\boldsymbol{R}_i^{t_i}(\boldsymbol{l}_i) = \boldsymbol{w}_i^{t_i} \cdot \phi(\boldsymbol{I},\boldsymbol{l}_i)$;
3: Use responses $\boldsymbol{R}_i^{t_i\prime}$ of the extend parts to correct the responses $\boldsymbol{R}_i^{t_i}$ of their corresponding body parts, $\boldsymbol{R}_i^{t_i} = \boldsymbol{R}_i^{t_i} + \epsilon_1 \cdot \boldsymbol{R}_i^{t_i\prime}$;
4: Collect the message $\boldsymbol{m}$ passed from the neighbors of the leaf parts and use that to correct the responses of the leaf parts, $\boldsymbol{R}_l^{t_l} = \boldsymbol{R}_l^{t_l} + \epsilon_2 \cdot \boldsymbol{m}(t_l)$;
5: Pass the message from the leaf parts to the root part, and use the information to correct the responses of parts passed by;
6: Choose the max score from the responses of the root part, $score = \max \boldsymbol{score}_{root}$.
7: Backtrack the maximum parameters to get the max scoring estimation $\boldsymbol{B}$.
8: **return** $\boldsymbol{B}$ and *score*;

4. Experiments

In this section, we will evaluate the performance of our model on the two different benchmark datasets.

4.1. Data Sets and Experimental Settings

Two datasets, i.e., the PARSE [1] dataset and UIUC People [20] dataset, are used in this work for performance evaluation. We follow the commonly used evaluation protocols in these two datasets with the standard data partitions. In particular, the PARSE dataset contains 100 training images (the first 100 images in the dataset) and 205 test images (the rest 205 images), while the UIUC People dataset contains 346 training images (the first 346 images in the dataset) and 247 test images (the last 247 images). For both datasets, one person in each image is manually annotated with 14 body parts. We draw our negative training images from INRIA dataset [21] as many other works do, and restrict our deformation constraints to capture the co-occurrence relations between pairwise adjacent parts. As mentioned before, the multi-scale HOG features [21] are adopted as our appearance descriptors, yielding a feature pyramid for each image.

As shown in Figure 3, we use 26 body parts (the same as those defined in [6]) for pose description, 11 bigger parts for local contextual information and 16 virtual parts for non-local information in the model. The annotations for the extra parts are derived from the existing annotations in the datasets. For each body part, 5-6 mixtures are used for modeling their appearance variations with each mixture type obtained by k-means clustering over joint locations respect to their parents, while for

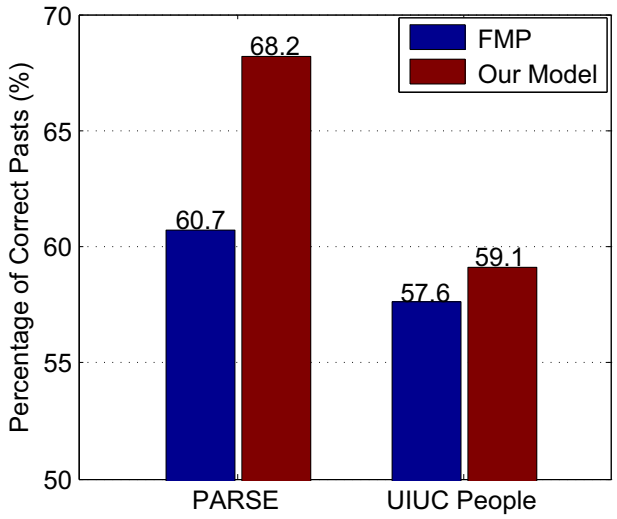


Figure 5: Comparative performance of the FMP model [6] and our model on the two datasets.

each leaf part, 4 virtual parts are assigned to collect the non-local contextual information (i.e., the long-range spatial constraints). The noises suppression parameters in Eq.(10) are set to be $\epsilon_1 = 0.5, \epsilon_2 = 0.4$, respectively in testing, which are a bit smaller than in training. Since the whole model is still a tree, we use the structural SVM code in [6] to train our model.

Since this work mainly focuses on the task of single human pose estimation in static images, we take the max scoring estimations given by the model as the final results. The Percentage of Correct Parts (PCP) metric [6, 22, 23] is adopted for performance measure. The PCP metric gives the percentage of the correctly localized body parts, and it can be calculated as follows: $PCP = \frac{nCorrect}{nLimbs \times nImages}$, where the *nCorrect* is the number of all correctly localized body parts in all images, *nLimbs* the number of body parts to be estimated in each image, *nImages* the size of the test set or the number of images in which the human subject is correctly localized according to a ground truth bounding box. It is worth mentioning that the definition of *nImages* and the correctly localized part is slightly different from work to work, and here we follow the criterion advocated in [22]: a part is localized correctly only if both the distances of the endpoints from their respective ground truth endpoints are less than a fraction (usually set as 0.5) of the part length. With this, the percentage of correct parts can be calculated for each image and then be averaged across all images.

4.2. Comparison with the Baseline Method

Since our work can be thought of as an extension to the Flexible Mixtures of Parts (FMP, [6]) model by incorporating local and non-local contextual information for more accurate parts detections, in this first series of experiments, we take the FMP as our baseline algorithm that serves to evaluate the effectiveness of the proposed method. Figure 5 gives the results. The figure clearly shows that our model outperforms the FMP on both datasets, with 7.5% and 1.5% improvement in performance on the PARSE dataset and the UIUC People dataset, respectively. This indicates that our method is effective by exploiting contextual information for more accurate pose estimation.

Figure 6 illustrates some cases in which the FMP model (the upper row) fails while our model (the bottom row) somewhat succeeds. One can see from the figure that our model improves upon the baseline method significantly in these cases, and as we will see later, this can be partly explained by the higher detection accuracies of our part detectors enhanced with the contextual information.

Figure 7 shows several typical failure cases of our model - some of them (the first two columns) are apparently due to the confusing edges (e.g., the arm-like horizontal shadow in the second column); and some of them (the third and fourth column) are possibly due to the rare poses which are not properly covered by the training data, while others (the last two columns) are likely caused by the local minimum problem during the part detection. The figure highlights the need for further research on these challenging situations.

4.3. Comparison with Other Similar Methods

Next we compare our method with several part-based pose estimation methods closely related. In particular, 1) the work of [1] uses a coarse part-based model for pose estimation, which has become the popular benchmark on the PARSE dataset; 2) the work of [24] makes a combination of the strong discriminatively trained appearance models and the kinematic tree prior on the configurations of body parts; 3) the work of [25] combines multiple cues (the gradient and color segmentation cues) to capture the coherent appearance properties of a limb; 4) the work [16] also uses the mixture of parts model for pose estimation, but the mixtures are mainly used for hypothesis voting and the relations between mixtures of different parts are not modeled; 5) the work of [6] extends [16] by adding the relation between mixtures of different parts into the model, which makes the mixtures more useful; 6) the work of [22] decomposes the body part into smaller parts and models them with a multi-layer composite model; 7) the work of [9] introduces a method of hierarchical poselets to extend the part-based model for human parsing, in which a loopy model is built with various constraints imposed on the part appearance. All these methods are evaluated by the percentage of correct parts metric, thereby facilitating our comparison.

Table 1 and Table 2 summarize some of the comparative results on the PARSE dataset and the UIUC People dataset respectively. It can be seen that our method is the best performer on both datasets, with a PCP value of 68.2% and 59.1%, respectively. It is worth mentioning that on the two parts that we argue are of particular importance to the overall performance, i.e., the lower legs and the lower arms, our detection rates reach 68.6% and 41.2%, respectively, higher than 65.6% and 36.6% of the previous best performer of [22], and also higher than 63.9% and 35.4% of our baseline method [6]. Similar observations can also be made on the more challenging UIUC People dataset. In the next section, we will give more experimental results which are helpful to explain why our method works well in these cases.

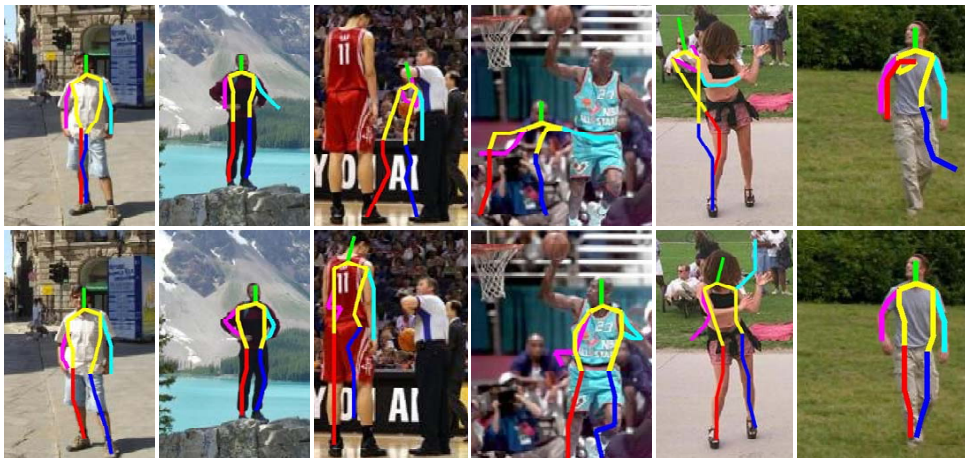


Figure 6: Illustration of some examples resulted from the FMP model [6] (the top row) and from our model (the bottom row).

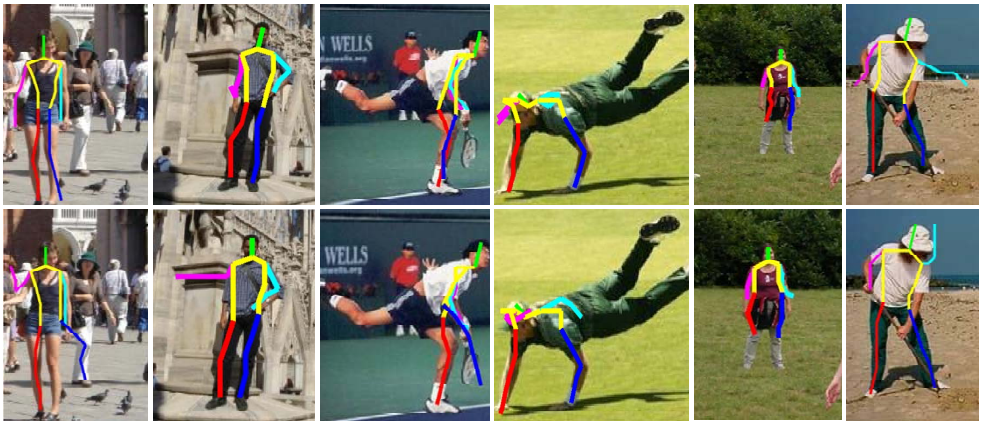


Figure 7: Illustration of the failure cases of our model (the bottom row), compared with results of the FMP model [6] (the top row).

4.4. The Usefulness of the Contextual Information

4.4.1. The Importance of Leaf Parts

The key idea of our work is based on the assumption that the accuracies of leaf parts have significant impact on the final performance of the whole model since they are the starting points of the message passing routes. Here by leaf parts we mean those parts like hands, foots that are positioned at the leaf nodes of the tree model. To verify the validation of this assumption in practice, a comparison is made on the performance of the FMP model with and without manually annotated leaf parts as ground truth (GT).

For performance evaluation, we calculate the error $\mu \pm \sigma^2$ (μ is the mean and σ^2 is the variance) for the normalized error distances of each parts, which is the distance between the estimated part position and its respective ground truth position, normalized by the average pairwise length of adjacent parts in the given image. Figure 8 gives the results. It can be seen that if we know the ground truth of leaf parts, the average part localization error of the FMP model could be reduced by as much as 60.0% (from 1.05 to 0.45), and the variance by as much as 87.6% (from 1.86 to 0.23). This is a significant improvement in performance and shows us the possible benefits we may have only if the leaf parts could be accurately localized.

4.4.2. Alternative Topologies for Message Transmission

In this section, we explore two other possible topologies for non-local contextual information collection. In particular, as illustrated in Figure 9, the first one (denoted as T1, see Figure 9(a)) is a simple variant of the originally proposed scheme (Figure 3(b)) by passing message from each neighbor node directly to the leaf node of interest, rather than passing through the intermediate nodes, while the second one (denoted as T2, see Figure 9(b)) is actually a non-tree model which collects information not only from nodes of the same limb but also from those on the other side as well - we conjecture that this may be useful since in some cases (e.g., walking) the poses of the left leg and the right leg can be closely correlated.

Table 3 shows the results on the PARSE dataset, where the influence of each topology on the performance both over the four leaf nodes and over all the nodes of the model is given. We have several observations from this table. First, both topologies (T1 and T2) improve the performance upon the baseline method, which indicates the benefit of non-local contextual information

Method	Torso	Head	Upper legs	Lower legs	Upper arms	Lower arms	Total
Ramanan [1]	52.1	13.6	37.5	31.0	29.0	17.5	27.2
Andriluka [24]	81.4	75.6	63.2	55.1	47.6	31.7	55.2
Johnson [25]	77.6	68.8	61.5	54.9	53.2	39.3	56.4
Singh [16]	**91.2**	76.6	71.5	64.9	50.0	34.2	60.9
Yang [6]	82.9	77.6	69.0	63.9	55.1	35.4	60.7
Duan [22]	85.6	80.4	71.7	65.6	57.1	36.6	62.8
Ours	90.2	**84.9**	**78.1**	**68.6**	**65.9**	**41.2**	**68.2**

Table 1: Comparison of various part-based pose estimation methods on the PARSE dataset.

Method	Torso	Head	Upper legs	Lower legs	Upper arms	Lower arms	Total
Ramanan [1]	44.1	30.8	9.5	25.3	11.1	25.5	21.8
Andriluka [24]	70.9	59.1	36.5	22.9	26.2	10.1	32.1
Wang [9]	86.6	68.8	56.3	50.2	30.8	20.3	47.0
Yang [6]	85.0	83.4	63.6	56.3	48.8	34.6	57.6
Ours	**87.9**	**85.4**	**64.2**	**57.5**	**49.2**	**38.3**	**59.1**

Table 2: Comparison of various part-based pose estimation methods on the UIUC Peoples dataset.

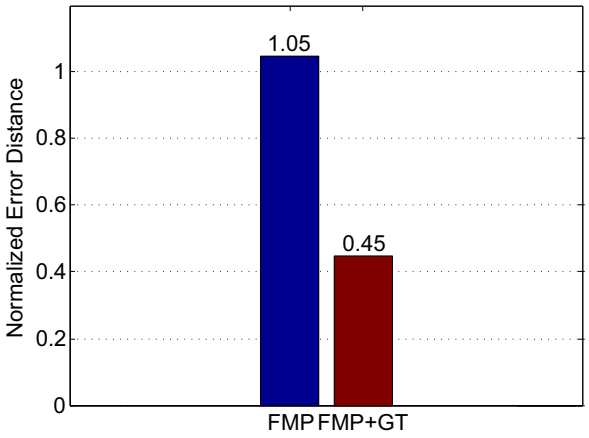


Figure 8: The effect of leaf parts on the overall performance.

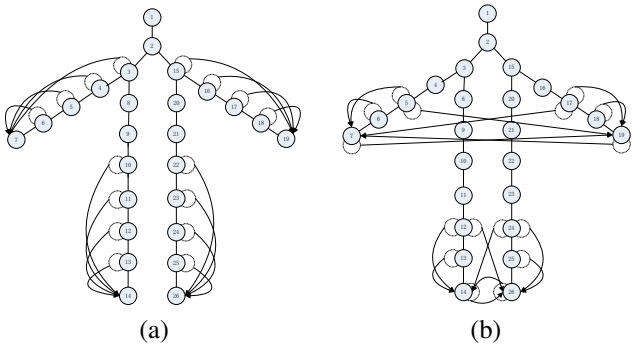


Figure 9: Alternative topologies for non-local contextual information collection (Left: T1; Right: T2).

for pose estimation even though it is exploited in different ways. Despite this, the table clearly shows that both alternative topologies give inferior performance compared to the proposed one of Figure 3(b). For T1, this can be partly explained by the fact that the spatial constraints tend to be weaker and less reliable when the contextual parts are far away from the leaf node. A natural way to address this is to reduce the weight of those nodes according to their distances to the leaf node of interest. Here we use the negative exponential as the weight function, and hence the messages passed from the virtual parts to the leaf part l become:

$$\boldsymbol{m}_l = \sum_{i \in V_{vir}(l)} exp(-(|i-l|-1)/\sigma) \cdot \boldsymbol{m}_i \tag{11}$$

where $V_{vir}(l)$ is the set of parts near to the leaf part l , and m_i is the message passed from part i to leaf part l, and σ is a parameter which is chosen carefully to control shape of the negative exponential function such that it will not decline too fast. One can see from Table 3 that this does improve the performance but is still slightly worse than our originally proposed scheme shown in Figure 3(b).

The model of T2 (Figure 9(b)), on the other hand, takes the spatial constraints between different limbs into consideration. From Table 3 we know that the performance of this model is even worse than that of model of T1 due to the unstableness of the spatial regularity it tries to capture. However, it can be conjectured that the spatial relationship between different limbs can be useful under certain circumstances, such as when people walk. To verify this, we evaluate T2 on the "Walking" subset of action dataset collected by Ikizler-Cinbis [26], where people in images of this subset are walking with various backgrounds. Table 4 gives the results. We can see that the performance of T2 on this dataset is much closer to that of the proposed model than that on the general dataset (PARSE), which confirms our previous speculation.

4.4.3. Improving the Localization Performance for Leaf Parts

Next we investigate the question of whether various types of contextual information are useful in improving the localization performance of leaf parts. Table 5 gives the results, in which the localization accuracy of each leaf part is measured by the average and the variance of the normalized error distance described in the previous section. One can see from the table that the local context (LC) information slightly improves the localization performance of 3 leaf parts (i.e., left and right hands, and right foot)

Method	Left Hand	Left Foot	Right Hand	Right Foot	Average Error
Before Message Propagation					
Baseline (FMP[6])	5.82 ± 9.70	3.50 ± 10.35	5.72 ± 9.20	4.14 ± 11.77	3.67 ± 8.43
Topology T1	4.18 ± 9.50	2.06 ± 5.63	4.23 ± 9.44	2.30 ± 6.37	3.20 ± 7.82
T1(weighted)	4.03 ± 9.55	2.09 ± 6.33	4.16 ± 8.29	**2.22 ± 5.72**	3.19 ± 7.76
Topology T2	4.31 ± 6.84	1.94 ± 6.17	4.49 ± 9.09	2.38 ± 5.62	3.24 ± 7.74
The proposed topology	**3.82 ± 10.22**	**1.80 ± 5.34**	**3.75 ± 9.94**	2.27 ± 7.18	**3.13 ± 7.94**
After Message Propagation					
Baseline (FMP[6])	1.98 ± 3.52	1.32 ± 3.81	1.94 ± 2.90	1.43 ± 3.81	1.05 ± 1.86
Topology T1	1.89 ± 2.96	1.21 ± 3.50	1.75 ± 2.40	1.36 ± 3.09	0.90 ± 1.50
T1(weighted)	1.84 ± 2.71	**1.21 ± 3.35**	**1.71 ± 2.43**	1.36 ± 3.01	0.90 ± 1.47
Topology T2	2.06 ± 2.97	1.26 ± 3.37	1.84 ± 2.98	1.40 ± 2.95	0.97 ± 1.59
The proposed topology	**1.75 ± 2.73**	1.26 ± 3.60	1.72 ± 2.57	**1.34 ± 3.13**	**0.89 ± 1.47**

Table 3: Performance comparison of various message passing topology schemes in terms of mean normalized error distance and variance ($\mu \pm \sigma^2$) on the PARSE dataset, where the rightmost column - "Average Error" - is calculated based on the error distances of all the body parts (26 parts) rather than the leftmost four column.

Method	Left Hand	Left Foot	Right Hand	Right Foot	Average Error
Before Message Propagation					
Baseline (FMP[6])	3.14 ± 3.60	**0.95 ± 1.06**	2.76 ± 2.57	**1.16 ± 1.06**	2.17 ± 3.33
T1(weighted)	2.01 ± 2.37	1.02 ± 1.24	1.95 ± 2.45	1.19 ± 1.22	1.97 ± 2.99
Topology T2	2.36 ± 2.71	1.14 ± 1.27	2.27 ± 2.81	1.23 ± 1.24	2.03 ± 3.17
The proposed topology	**2.00 ± 2.50**	1.02 ± 1.24	**1.85 ± 2.21**	1.20 ± 1.20	**1.96 ± 3.08**
After Message Propagation					
Baseline (FMP[6])	1.55 ± 1.41	1.02 ± 1.22	1.60 ± 1.38	1.21 ± 1.27	0.96 ± 0.77
T1(weighted)	1.51 ± 1.41	**1.00 ± 1.26**	1.49 ± 1.24	1.16 ± 1.23	0.89 ± 0.72
Topology T2	1.53 ± 1.41	1.07 ± 1.29	1.49 ± 1.34	1.14 ± 1.22	0.90 ± 0.74
The proposed topology	**1.51 ± 1.40**	1.01 ± 1.24	**1.45 ± 1.28**	**1.12 ± 1.23**	**0.88 ± 0.73**

Table 4: Performance comparison of various message passing topology schemes in terms of mean normalized error distance and variance ($\mu \pm \sigma^2$) on "walking" subset, where "Average Error" column is calculated based on the error distances of all the body parts (26 parts) rather than the leftmost four column.

among the four parts. However, when the non-local context information is added, the localization performance of all the leaf parts is improved significantly. In particular, we tested various ranges of non-local context information depending on how many neighboring virtual nodes are added for message passing for the given leaf nodes (c.f., Figure 3(b)): i.e., short range (2 neighbors), middle range (4 neighbors), and long range (6 neighbors). One can see from Table 5 that the scheme of long range context combined with local context performs best - it reduces the average normalized error by 57.5% for the left hand, 64.0% for the right hand, 52.3% for the left foot, and 50.7% for the right foot, respectively. For comparison, we also give the results assuming that the exact locations of these supporting nodes are known (GT). Note that in this later cases (GT), the small range context combined with local context wins.

Furthermore, the table reveals that different range of context information has different influence on the localization accuracies of leaf parts. Intuitively, by taking more neighboring nodes into account for a given leaf node, we have more contextual information for reference in locating that leaf part, hence its localization accuracy is expected to be improved. Unfortunately, that's not the whole story - remember that we have to propagate such location information of leaf parts upward to help determining the positions of other parts. It should be noted that the noise from the context information could be accumulated during that message passing procedure, and as we will see later, it is very difficult to suppress such noise. In other words, the final performance of the system could be hurt if the context information is not used properly. We will present more analysis on this in the next section.

4.4.4. The Impact of Leaf Parts on the Final Performance

We show in the previous section that contextual information is helpful in improving the localization accuracies of leaf parts, and thus it will be interesting to know how such improvement could be turned into the final performance gain of the whole system after the message propagation. In this section, we try to answer this question experimentally.

Figure 10 gives the final performance with various context information incorporated into the leaf parts localizations. It can be seen that by adding the local context information, the mean normalized error distance reduces by 13.3% from 1.05 to 0.91, and further 2.0% of reduction could be gained if the non-local context is applied on. Actually, totally 26.7% reduction in error could be obtained in the ideal case (with ground truth), as indicated in the figure.

It should be noted that in Figure 10, the middle range of non-local information is used. However, from Table 5, we know that middle range context is not the best performer in terms of localization accuracies of leaf parts. To investigate this, Figure 11 gives the detailed final performance of the system after message propagation with different ranges of non-local context applied on the leaf parts. The figure shows that compared with the long range and the short range, the middle range context performs the best.

To gain further understanding on this issue, we compare the results of Figure 10 with those of Table 5. Actually, we see from Table 5 that, although the model with long range contextual information do have the most accurate leaf part detectors, it is not accurate enough to make the overall performance improved as a whole, and that's why the subsequent message propagation

Method	Left Hand	Left Foot	Right Hand	Right Foot
Baseline (FMP[6])	5.82 ± 9.70	3.50 ± 10.35	5.72 ± 9.20	4.14 ± 11.77
Local Context (LC)	5.28 ± 11.04	3.79 ± 11.34	5.35 ± 12.05	3.82 ± 9.40
Non-Local Context				
ShortRange(2 neighbors)+LC	5.24 ± 11.16	2.44 ± 8.55	4.95 ± 10.21	2.65 ± 7.77
MiddleRange(4 neighbors)+LC	3.82 ± 10.22	1.80 ± 5.34	3.75 ± 9.94	2.27 ± 7.18
LongRange(6 neighbors)+LC	**2.46 ± 7.26**	**1.67 ± 5.31**	**2.06 ± 3.63**	**2.04 ± 5.64**
Non-Local Context with Ground Truth (GT)				
ShortRange(2 neighbors)+LC+GT	**1.17 ± 1.18**	**0.71 ± 0.85**	**1.04 ± 0.98**	**0.88 ± 1.00**
MiddleRange(4 neighbors)+LC+GT	1.70 ± 2.52	0.94 ± 1.95	1.62 ± 2.30	1.12 ± 1.86
LongRange(6 neighbors)+LC+GT	1.84 ± 3.18	1.26 ± 3.21	1.73 ± 2.46	1.32 ± 2.91

Table 5: Mean normalized error distance and variance ($\mu \pm \sigma^2$) for the leaf parts with different contextual information used in our model on the PARSE dataset.

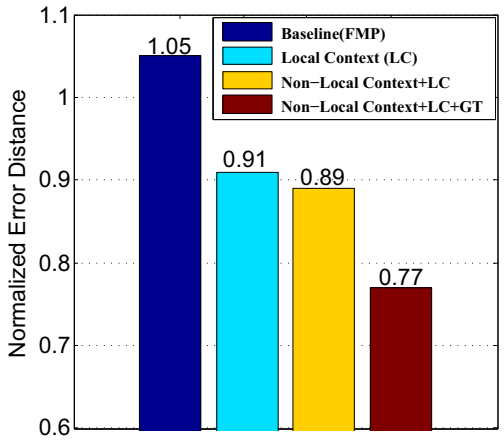


Figure 10: The effect of incorporating various context information in the leaf parts localizations on the final performance of the system, in terms of normalized mean error distance μ.

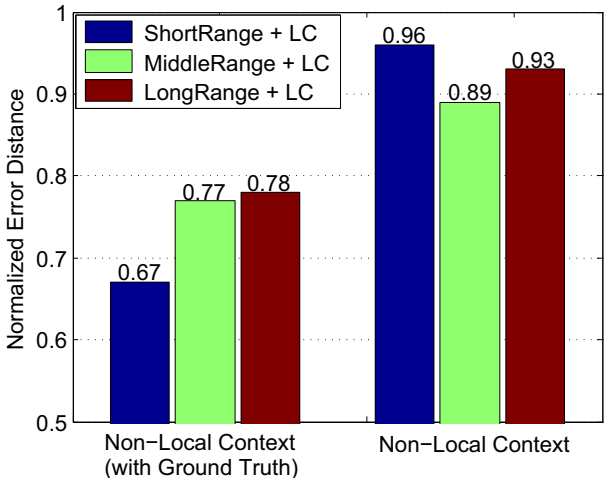


Figure 11: Comparative performance of various ranges of non-local context information.

process is needed. In this process, a patch with the maximum response will be chosen for propagating its localization information. However, in a model with long range contextual information incorporated, the patch with the maximum response tends to dominate the response values among others such that there are few chances for other patches to compete with it. This overconfident choice actually weakens the effects of the message passing and backtracking, and thereby reducing the performance. On the other hand, the model with middle range context effectively avoids the above problem by reducing the number of neighboring nodes for context information passing, which leads to much better performance (c.f.,Fig. 11).

4.4.5. On Over-fitting

As we can see, the number of parameters in our model increases to about 3 times of that in the FMP model due to the extra contextual nodes introduced. It is well known that more parameters usually lead to higher danger of over-fitting (i.e., the model fits the training samples well but cannot generalize to the unseen samples). To investigate this, we conduct a series of experiments to see how the test performance changes with the varying number of training samples. In particular, we generate a new training set with 1000 samples based on the original images in the PARSE dataset by rotating (in four angles(-16, -8, 4, 12 degrees)) or flipping them. Then we randomly sample varying number of images from this pool of training samples and train the models respectively. We repeat this procedure for 10 times and report the performance of our model and the baseline FMP model in terms of average normalized error distance.

Figure 12 gives the results [1] and Table 6 details the range of performance variations accordingly. From the figure one can see a common trend for both our method and the FMP method that with the growing number of training images, the generalization error reduces gradually. In addition, it's interesting to see that our method behaves much better than the FMP model in terms of generalization error and training error when we have a few hundreds of training samples, and that when the number of training samples is below 30, the performance of our method is worse than that of the FMP model. In other words, only when the training samples are extremely limited, our model is more likely to be over-fitting than the baseline model, otherwise our model actually performs well. This can be partly explained by the regularization mechanism adopted in the structured-output SVM that penalizes those models fitting data well but with high model complexity.

[1] We only show the performance curves with the number of training samples up to 400 since the curves tend to be steady beyond that number.

Categories	Training Error of FMP Method			Training Error of Our Method			Test Error of FMP Method			Test Error of Our Method		
	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
10 Samples	0.12	0.17	0.1464	0.14	0.16	0.1524	1.37	2.28	1.8100	1.45	2.41	1.8984
30 Samples	0.15	0.18	0.1628	0.15	0.17	0.1626	1.22	1.51	1.3654	1.21	1.39	1.2953
50 Samples	0.17	0.20	0.1765	0.16	0.18	0.1704	1.11	1.32	1.2345	1.08	1.27	1.2001
70 Samples	0.18	0.22	0.2019	0.17	0.20	0.1818	1.11	1.22	1.1812	1.04	1.24	1.1359
100 Samples	0.20	0.28	0.2431	0.18	0.22	0.1961	1.04	1.20	1.1081	1.01	1.10	1.0592
200 Samples	0.32	0.39	0.3549	0.23	0.28	0.2510	0.99	1.08	1.0377	0.89	1.02	0.9695
300 Samples	0.36	0.48	0.4309	0.27	0.31	0.2936	0.97	1.06	1.0086	0.89	1.01	0.9527
400 Samples	0.41	0.58	0.5148	0.29	0.36	0.3257	0.96	1.03	0.9912	0.89	0.98	0.9324

Table 6: The range of performance variations of the FMP model and our method in terms of mean normalized error distance (μ) with different number of training samples.

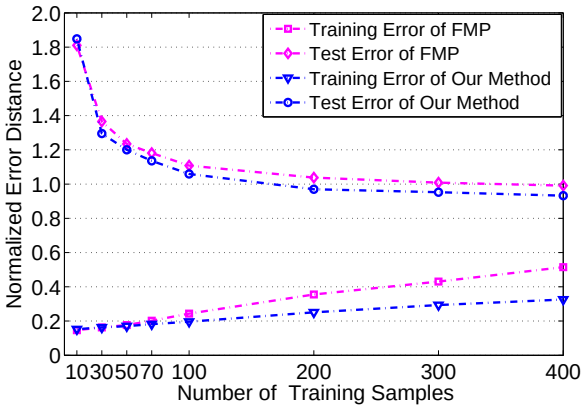


Figure 12: Performance comparison of our model and the FMP model with varying number of training samples.

4.5. Discussions

As mentioned before, the contextual information from the adjacent parts contains both useful information and the noise due to the inaccurate locations of these parts. Hence it is interesting to investigate the methods to suppress the noise while keeping the useful information. We tested the following strategies in this work:

- Reweighting: In E.q. (10), the reweighting parameter ϵ_2 is used to suppress the noises in the contextual information $\boldsymbol{m}$ used by the leaf parts' responses $\boldsymbol{R}_l$: $\boldsymbol{R}'_l = \boldsymbol{R}_l + \epsilon_2 \cdot \boldsymbol{m}$. Note that the value of ϵ_2 is set by cross validation in our work, but it could suppress both the noise and the useful information at the same time since it cannot differentiate between them.

- Response Normalization: A linear transformation is applied to normalize the response values of leaf parts $\boldsymbol{R}_l$ into some fixed range: $\boldsymbol{R}'_l = a * (\boldsymbol{R}_l + \boldsymbol{m}) + b$, where $max(|\boldsymbol{R}'_l|) = max(|\boldsymbol{R}_l|)$, and a and b are two parameters. This will normalize the extra energy introduced by the contextual information but under the risk of weakening the role played by the leaf part itself.

- Using Max. Response: Only the maximum response of the leaf parts is used for message propagation: $\boldsymbol{R}'_l = max\_mask(\boldsymbol{R}_l + \boldsymbol{m})$, where max_mask is a function which only keeps the maximum response and set all the other responses to zeros.

Method	Total Error
Reweighting	**0.89 ± 1.49**
Response Normalization	0.93 ± 1.61
Using Max. Response	1.32 ± 2.79
NMS with radius=2 pixel	0.94 ± 1.64
NMS with radius=1 pixel	0.93 ± 1.62

Table 7: Comparative performance ($\mu \pm \sigma^2$) of various noise suppression strategies.

- Non-maximum suppression (NMS): i.e., the classical non-maximum suppression (NMS) method with suppression radius set to be 1 or 2 pixels.

Table 7 gives the results. One can see from this table that overall there is no obvious difference between these noise suppression strategies (we take the first one in this work). Actually, it is very difficult to tell noise from useful information in general, and various noise suppression strategies are taken with specific assumption about the properties of the noise, otherwise a compromise has to be made.

5. Conclusions

In this paper, we presented a simple but useful extension model with the contextual information to the tree-based mixture parts model. Specifically, we use the local context information to reduce the influence of noises in the image and improve the performance of part detectors in general. Furthermore, based on the observation that the performance of leaf parts detectors is essential to the overall performance of the tree-based model, we propose a novel way to improve the localization accuracies of leaf parts by incorporating non-local context information, and we show that this benefits the pose estimation through the message propagation mechanism. Last but not least, we investigate the influence of various range of non-local context information, showing that although long range context information is helpful to the detection performance of individual leaf part, the noise contained could be accumulated during the message passing procedure. Consequently in practice a tradeoff has to be made when considering how much contextual information to use.

We believe that this is the first work that enhances the tree-based mixture parts model by exploiting the message passing mechanism to pass down the non-local contextual information. Our experimental results on two challenging datasets verify the feasibility and effectiveness of the proposed method. As

mentioned before, pose estimation in the real world has wild range of applications but faces many difficulties needing to be addressed. Currently we are working on more effective ways to suppress the noise from the long range context information while keeping much useful information as possible.

Acknowledgements

The authors want to thank the editors and anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Science Foundation of China (61073112, 61035003, 61373060), Jiangsu Science Foundation (BK2012793), Qing Lan Project, Research Fund for the Doctoral Program (RFDP) (20123218110033).

References

[1] Ramanan D. Learning to parse images of articulated bodies. In: NIPS: the Twentieth Annual Conference on Neural Information Processing Systems; Vancouver, Canada, December 2006, pp.1129-1136.

[2] Rogez G, Rihan J, Ramalingam S, Orrite C, Torr PH. Randomized trees for human pose detection. In: CVPR2008: the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Anchorage, USA, June 2008, pp.1-8.

[3] Fathi A, Mori G. Human pose estimation using motion exemplars. In: CCV2007: the 2007 IEEE 11th International Conference on Computer Vision; Rio de Janeiro, Brazil, October 2007, pp.1-8.

[4] Junior J, Julio C, Jung CR, Musse SR. Skeleton-based human segmentation in still images. In: ICIP2012: the 19th IEEE International Conference on Image Processing; Lake Buena Vista, USA, September 2012, pp.141-144.

[5] Felzenszwalb PF, Huttenlocher DP. Pictorial structures for object recognition. International Journal of Computer Vision. 2005; 61(1), pp.55-79.

[6] Yang Y, Ramanan D. Articulated pose estimation with flexible mixtures-of-parts. In: CVPR2011: The 24th IEEE Conference on Computer Vision and Pattern Recognition; Colorado Springs, USA, June 2011, pp.1385-1392.

[7] Binford, O T. Visual Perception by Computer. In: Proceedings of the 30 IEEE Conference on Systems and Control; Miami, FL, 1971, pp. 262.

[8] Forsyth DA, Fleck MM. Body plans. In: CVPR1997: the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Juan, Puerto Rico, June 1997, pp.678-683.

[9] Wang Y, Tran D, Liao Z. Learning hierarchical poselets for human parsing. In: CVPR2011: The 24th IEEE Conference on Computer Vision and Pattern Recognition; Colorado Springs, USA, June 2011, pp.1705-1712.

[10] Sigal L, Black MJ. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In: CVPR2006: the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; New York, USA, June 2006, pp.2041-2048.

[11] Lee MW, Cohen I. Proposal maps driven mcmc for estimating human body pose in static images. In: CVPR2004: the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Washington DC, USA, June 2004, pp.334-341.

[12] Wang Y, Mori G. Multiple tree models for occlusion and spatial constraints in human pose estimation. In: ECCV2008: the 10th European Conference on Computer Vision; Marseille, France, October 2008, pp.710-724.

[13] Burl MC, Weber M, Perona P. A Probabilistic Approach to Object Recognition Using Local Photometry and Global Geometry. In: ECCV98: the 5th European Conference on Computer Vision; Freiburg, Germany, June 1998, pp.628-641.

[14] Fergus R, Perona P, Zisserman A. Object class recognition by unsupervised scale-invariant learning. In: CVPR2003: the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Madison, USA, June 2003, pp.264-271.

[15] Weber M, Welling M, Perona P. Unsupervised learning of models for recognition. In: ECCV2000: the 6th European Conference on Computer Vision; Dublin, Ireland, June 2000, pp.18-32.

[16] Singh VK, Nevatia R, Huang C. Efficient inference with multiple heterogeneous part detectors for human pose estimation. In: ECCV2010: the 11th European Conference on Computer Vision; Heraklion, Greece, September 2010, pp. 314-327.

[17] Johnson S, Everingham M. Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC2010: the 2010 British Machine Vision Conference; Aberystwyth, UK, August 2010, pp.1-11.

[18] Tian Y, Zitnick CL, Narasimhan SG. Exploring the spatial hierarchy of mixture models for human pose estimation. In: ECCV2012: the 12th European Conference on Computer Vision; Florence, Italy, October 2012, pp.256-269.

[19] Ukita N. Articulated pose estimation with parts connectivity using discriminative local oriented contours. In: CVPR2012: the 2012 IEEE Conference on Computer Vision and Pattern Recognition; Providence, USA, June 2012, pp.3154-3161.

[20] Tran D, Forsyth, A D. Improved human parsing with a full relational model. In: ECCV2010: the 2010 11th European Conference on Computer Vision; Heraklion, Greece, September 2010, pp.227-240.

[21] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: CVPR2005: the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; San Diego, USA, June 2005, pp.886-893.

[22] Duan K, Batra D, Crandall D. A Multi-layer Composite Model for Human Pose Estimation. In: BMVC2012: the 2012 British Machine Vision Conference; Surrey, UK, September 2012, pp.1-11.

[23] Pishchulin L, Jain A, Andriluka M, Thormahlen T, Schiele B. Articulated people detection and pose estimation: Reshaping the future. In: CVPR2012: the 2012 IEEE Conference on Computer Vision and Pattern Recognition; Providence, USA, June 2012, pp.3178-3185.

[24] Andriluka M, Roth S, Schiele B. Pictorial structures revisited: People detection and articulated pose estimation. In: CVPR2009: the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Miami, USA, June 2009, pp.1014-1021.

[25] Johnson S, Everingham M. Combining discriminative appearance and segmentation cues for articulated human pose estimation. In: ICCV2009: the 2009 IEEE 12th International Conference on Computer Vision Workshops; Kyoto, Japan, Sept. 2009, pp.405-412.

[26] Ikizler-Cinbis N, Cinbis RG, Sclaroff S. Learning actions from the web. In: ICCV2009: the 12th International Conference on Computer Vision; Kyoto, Japan, September 2009, pp.995-1002.