Improved Human Parsing with a Full Relational Model

Duan Tran and David Forsyth

University of Illinois at Urbana-Champaign, USA
{ddtran2,daf}@illinois.edu

Abstract. We show quantitative evidence that a full relational model of the body performs better at upper body parsing than the standard tree model, despite the need to adopt approximate inference and learning procedures. Our method uses an approximate search for inference, and an approximate structure learning method to learn. We compare our method to state of the art methods on our dataset (which depicts a wide range of poses), on the standard Buffy dataset, and on the reduced PASCAL dataset published recently. Our results suggest that the Buffy dataset over emphasizes poses where the arms hang down, and that leads to generalization problems.

1 Introduction

In human parsing, we have a region of interest (ROI) containing a person, perhaps produced by a detector, and we must produce an accurate representation of the body configuration. This problem is an important part of activity recognition; for example, the ROI might be produced by a detector, but we must know what the arms are doing to label the activity. The representation produced is usually a stick figure, or a box model, but may be image regions or joint locations. All representations encode the configuration of body segments.

It is usual to represent pairwise spatial relations between locations structured into a kinematic tree, so that dynamic programming can be used for inference [10,6]. The joint relations encoded by the kinematic tree model are important, but there are other important relations. Limbs on the left side of the body usually look like those on the right. This cue should be important, because limbs are genuinely difficult to detect, particularly in the absence of an appearance model. Even the strongest recent methods have difficulty detecting forearms (e.g. [1], 32%, p8). Inference difficulties occur when one encodes relations between all pairs of segments, because finding the best parse now becomes max-cut. Approximate inference on sets of extended image segments can produce good parses for difficult images [16]. However, there is no evidence comparing the benefits of a full model against the cost of approximate inference.

In this paper we explore the advantages of representing a full set of relations for human parsing. We show strong quantitative evidence that the advantages of representing a full set of relations between segments outweigh the costs of approximate inference and approximate learning. We concentrate on upper body parsing, and show results on Buffy and Pascal dataset [9], and on a new dataset where the prior on body configuration is quite weak.

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part IV, LNCS 6314, pp. 227–240, 2010.

Fig. 1. A tree model of the upper body represents only relations that can be organized as a tree (always the kinematic relations in the natural tree, **top left**). By doing so, it omits relations that are important; for example, the left arm typically looks like the right arm. A full model (**bottom left** — we have indicated only some of the relations omitted by the tree model) encodes these relations, at the cost of approximate inference and approximate training. There is qualitative evidence in the literature that full models can yield good parses [16]; in this paper, we show quantitative evidence on two datasets that a full model offers strong advantages over a tree model. On the **right**, we show parses derived from a tree model (**top**) and a full model (**bottom**); note that the appearance constraints in the full model often help arms to be in the right place. This is confirmed by our quantitative data.

1.1 Related Work

For a tree structured kinematic model, and segments with known appearance, inference is by dynamic programming (the "pictorial structure model" [5]). A similar approach can be applied to informative local patches [15], or to joint locations using velocities at the joints, assuming known activity [25]. If segment appearance is unknown, it can be recovered from motion sequences [19], or by an iterative procedure of estimating appearance, then configuration, etc. [18]. The iterative procedure can produce good parses, but can fail if the search starts poorly. These methods can be costly, because the search space is a discretization of all possible segment configurations. Improvements result from estimating head location, then pruning the search space [9]; and from tuning the initial appearance model with spatial priors on segment locations and respecting

likely interactions between segment appearance models (the upper arm is often the same color as the upper body) [3]. Alternatively, local segment scores can be computed by appearance independent segment detectors; edge based limb detectors give limited performance [21], but a combination of HOG and color segmentation features beats the original iterative process [14].

There has been considerable experimental tuning of various tree models. A ten-body segment model (head, upper body, two for each arm and two for each leg) is now universal. The standard scoring procedure regards a prediction correct if its endpoints lie within 50% of the ground truth segment length from the true positions; this score is fairly generous, because body segments are long. Nonetheless, high scores are difficult to get. Ramanan [18] has published a widely used dataset for full body localization, on which the best method (Andriluka *et al.*, 2009 [1]) gets 55.2%. Ferrari *et al.* [9] published another for upper body localization, on which the best method (Eichner *et al.*, 2009 [3]) gets 80.3% in its best configuration on Buffy dataset. On a selected subset of PASCAL challenge images, this method gets 72.3%. The strongest methods all use carefully constructed and tuned segment detectors, with search space pruning.

The tree model presents real difficulties: limb occlusions seem to be correlated; tree models tend to prefer parses that superimpose both arms or both legs on one promising set of image segments; and a tree model cannot encode the tendency of the left arm (resp. leg) to look like the right arm (resp. leg). Fergus *et al.*were able to work with models encoding all relations between up to seven object parts using efficient pruning methods [8]. Tian *et al.*extend the tree model by inserting appearance constraints between segments (e.g. left lower leg looks like right lower leg); in this case, the tree model supplies a strong enough bound that exact inference using branch and bound is possible [28]. Sapp *et al.*demonstrate improvements in performance by making some terms in the tree model depend on an approximate estimate of global pose, obtained by matching the test image to the training dataset [22]. This suggests strong structural correlations appear in poses. Correlated limb occlusions can be dealt with using a mixture of trees without complicating inference [11]. Alternatively, Sigal *et al.*use a modification to the likelihood to avoid double counting in the case of occluded limbs [24]. Doubled limbs can be controlled with "repulsive" edges [3,13], or by converting the energy function into a posterior probability, drawing samples, and using some process to decide which is best (for example, rejecting parses where arms overlap) [5]. An alternative, which requires approximate inference, is to require that the model covers more of the image pixels [12].

Recent work has shown that human parses can be helped by identifying objects people might be using [2,32]. Yao *et al.*use a fixed set of quantized body poses, with exact segment locations depending probabilistically on (a) the type of the pose (and so on the activity label and nearby objects through this) and (b) other segment locations [32]. Their method can produce parses that are as good as, or better than, the state of the art, for a sports dataset of stylised poses.

Another important difficulty is determining which poses to work with. In our opinion, the performance of a parser should be as close to pose-independent as possible. That is, the parser should be tested (if not trained) on a dataset with a rich selection of poses at approximately even frequencies. This is complicated, because natural data

sources often have strong biases — as we shall see, TV actors in stills tend to have their arms by their sides. The result is very strong effects due to the prior, which can cause generalization problems. For these reasons, we have collected a further dataset that emphasizes rich upper body configurations.

2 Method

Our approach defines a search space in the image using a set of tuned body segment detectors. We then build an energy model that is regressed against actual loss for a set of parses of each training image. Our model scores appearance and spatial relations between all pairs of segments in the image. We then find the best parse by an approximate maximization procedure.

We have detectors for upper body, head and arm segments. Our detectors do not distinguish between upper and lower arms. We must choose a label (head, upper body, left/right upper/lower arm, null) for each response. For image $\mathcal{I}$, we build a scoring function $C(L;\mathcal{I})$ which evaluates a labelling L of the responses. We consider only labellings that are consistent, in the sense that we do not attempt to label head detector responses as upper bodies, etc. Write S_i for the i-th body segment in the model, D_j for the j-th detector response in the image, and $L(S_i)$ for the image segment labelled S_i by L. Our energy is a linear combination of unary and binary features, which we write as

$$C(L;\mathcal{I}) = \sum_{i \in \text{features}} w_i \phi_i(L;\mathcal{I}) = \mathbf{W}^T \Phi(L;\mathcal{I})$$

where each feature ϕ_i is either a unary feature (yielding $\phi_i = \phi_i(S_j, L(S_j);\mathcal{I})$) or a binary feature (yielding $\phi_i = \phi_i(S_j, S_k, L(S_j), L(S_k);\mathcal{I})$). We do *not* require that the set of binary features form a tree, but represent all pairs. Our features measure both spatial and appearance relations (section 2.4). The scoring function can be converted to an energy by $E(L) = -C(L)$; a probability model follows, though we do not use it.

2.1 Searching a Full Energy Model

Finding the best labelling involves solving a general zero-one quadratic form subject to linear constraints, and there is no exact algorithm. While approximate algorithms for MRF's could be applied, most labels are null and there is only one instance of each non-null label, meaning that expansion moves are unlikely to be successful. We use an approximate search procedure that relies on the proven competence of tree models.

The upper body detector is quite reliable, so there are relatively few false positives. This means we can search for a configuration at each upper body, then take the overall best configuration. Because tree models are quite reliable, we can use specialised tree models to produce arm candidates on each side of each given upper body, then evaluate all triples of right arm-torso-left arm. Finally, we use a local search to improve segments.

Obtaining arm candidates: We need to obtain candidates for left (resp. right) arm that have a good chance of being in the final configuration. We can do so by simplifying the

cost function, removing all terms apart from those referring to upper body, left (resp. right) upper arm and left (resp. right) lower arm. The resulting simplified cost function is easily maximised with dynamic programming. We keep the top 300 candidates found this way for each side.

Building good triples: We now have 300 candidates each for left (resp. right arm), and a set of head candidates. We obtain the top five triples by exhaustive evaluation of the whole cost function.

Polishing with local search: Limb detectors chatter, because they must respond to contrast at limb edges and limbs are narrow; it is usual to see more than one response near a segment. To counteract this effect, we polish each of the top five triples. We repeatedly fix five segments and search for the best candidate for the sixth, stopping when all segments have been visited without change. We now report the best of the polished five triples for each upper body.

Detection: We report the parse associated with the best upper body detector response. In principle, one could parse multiple people in an image by keeping all such parses, applying a threshold to the cost function, and using non-maximum suppression (to control chatter at the upper body detector); since most images in evaluation datasets contain single people, and since our focus is on "hard parses", we have not investigated doing so.

Complexity: With 6 human parts in the model, the exact solution will cost $O(T * H * LUA * LLA * RUA * RLA)$ where T, H are torso and head detections, LUA, LLA and RUA, RLA are left upper, lower arms (resp. right upper and lower arms) detections. While T and H are small (less than 10 each), LUA, LLA, RUA, RLA are quite large (normally more than 100 each after pruning by the closeness to the torso), this complexity is practically intractable. However, our approximate solution has complexity $O(T * H * LA * RA) - -LA, RA$: numbers of full left (resp. right arms) that we keep top 300 for each). This complexity is tractable, and though it is an approximation, it still proves its benefit of improving the performance. In fact, Our implementation in C just takes around 5 seconds for one parsing on a computer of Xeon 2.27HGz.

2.2 Training a Full Energy Model

We wish to train the energy model so that detections using that model are as good as possible. **Structure learning** is a method that use a series of correct examples to estimate appropriate weightings of features relative to one another to produce a score that is effective at estimating configuration (in general [26,27]; applied to parsing [29]). For a given image $\mathcal{I}$ and known $\mathbf{W}$ the best labelling is

$$\underset{L \in L(\mathcal{I})}{\arg\max} \mathbf{W}^T \Phi(L; \mathcal{I})$$

though we cannot necessarily identify it. We choose a loss function $\mathcal{L}(L_p, \hat{L})$ that gives the cost of predicting L_p when the correct answer is $\hat{L}$. Write the set of n examples as

$\mathcal{E}$, and $L_{p,i}$ as the prediction for the i'th example. Structure learning must now estimate a $\mathbf{W}$ to minimize the hinge loss as in [20,30,26]

$$\min \lambda \frac{1}{2} \parallel \mathbf{W} \parallel^2 + \frac{1}{n} \sum_{i \in examples} \xi_i$$

subject to the constraints

$$\forall i \in \mathcal{E},\ \mathbf{W}^T \Phi(\hat{L}; \mathcal{I}_i) + \xi_i \geq \max_{L_{p,i} \in L(\mathcal{I}_i)} (\mathbf{W}^T \Phi(L_{p,i}; \mathcal{I}_i) + \mathcal{L}(L_{p,i}, \hat{L}_i))$$
$$\xi_i \geq 0$$

At the minimum, we can choose the slack variables ξ_i to make the constraints equal. Therefore, we can move the constraints to the objective function, which is:

$$\lambda \frac{1}{2} \parallel \mathbf{W} \parallel^2 + \frac{1}{n} \sum_{i \in examples} \max_{L_{p,i} \in L(\mathcal{I}_i)} \left(\begin{array}{c} \mathbf{W}^T \Phi(L_{p,i}; \mathcal{I}_i) + \\ \mathcal{L}(L_{p,i}, \hat{L}_i) - \mathbf{W}^T \Phi(\hat{L}; \mathcal{I}_i) \end{array} \right)$$

Notice that this function is convex, but not differentiable. We use the cutting-plane method of [30], as implemented in SVM-Struct package[1]. Our approach is:

Start: we initialize $\mathbf{W}$, and prepare a pool of candidate labellings $\mathcal{C}_i^{(0)}$ for each example image using the search of section 2.1. Then, iterate multiple rounds of

1. for each example, compute the best (most violated constraint) labelling $L_{p,i} = \arg\max_{L \in \mathcal{C}_i^{(k)}} \mathbf{W}^T \Phi(L; \mathcal{I}_i) + \mathcal{L}(L_{p,i}, \hat{L}_i)$.
2. pass these labellings to SVM-Struct to form cutting planes to update $\mathbf{W}$.

This procedure will stop until there are no violated labelling found (in this case, the ground truth labelling is the highest score) or no significant change in the objective value when updating $\mathbf{W}$. We observe that the learning converges after 50-60 iterations.

Table 1. Summary of part detectors. Equal error rates (EER) are computed with 5-fold cross validation. The lower arm detector is not comparable to others as it tends to be dataset dependent. We operate the detectors at 92% recall and given a upper body candidate we keep the 300 best lower arm responses.

Detector	Size	Features	EER
Upper body	80x80	HOG	0.096 +/-0.005
Head	56x56	HOG	0.123+/0.012
Lower arm	30x30	HOG, SSIM	0.249+/-0.068

[1] http://svmlight.joachims.org/svm_struct.html

2.3 Part Detectors

We have detectors for upper body, head and arm segments, but do not distinguish between upper and lower arms for a total of three part detectors. The detectors are oriented, and we use a total of 25 orientations of $(-180^o..180^o)$ for arm detectors and 13 orientations of $(-90^o..90^o)$ for head detector and upper body detector. We use code from

Table 2. This table shows pairwise features (undirected links) to be computed. [D]: distance binning, [A]: appearance difference, [N]: angle, [O]: overlap

Parts	Upper body	Head	LUA	LLA	RUA
Upper body	-	-	-	-	-
Head	D,A,N,O	-	-	-	-
LUA	D,A,N,O	A,O	-	-	-
LLA	D,A,N,O	A,O	D,A,O	-	-
RUA	D,A,N,O	A,O	A,O	A,O	-
RLA	D,A,N,O	A,O	A,O	A,O	D,A,O

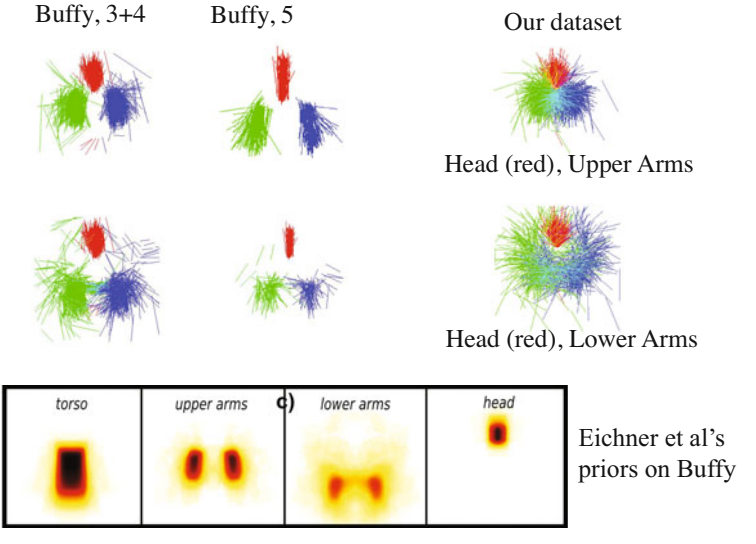

Fig. 2. In the Buffy dataset, upper arms hang by the sides of the body, and lower arms mostly do so as well. This very strong spatial prior can overcome contributions by other parts of a model, but impedes generalization. **Above:** Scatter plots of head and upper arm (**top row**) or lower arm (**bottom row**) sticks with respect to fixed upper body position for the Buffy 3 and 4 ground truth, Buffy 5 ground truth, and our ground truth. Notice how compact the prior configuration is for the Buffy datasets. Our dataset emphasizes a wide range of body configurations. **Below:** Part of figure 1, from [3], for reference, showing the location priors derived in that paper for the Buffy dataset; again, note the highly compact prior.

Table 3. Average PCP over body segments for a full model; for a tree model; and for Eichner and Ferrari [3], who use a tree model with a location prior to recover appearance. Performance of the full model is much better than performance of tree models, except for the model of Eichner and Ferrari applied to Pascal or to buffy_s256. However, for all models, training on Buffy_256 leads to strong generalization problems (performance on Buffy_256 is much better than performance on other test sets), most likely because of the quite strong bias in arm location. We believe that the very strong performance of Eichner and Ferrari on Buffy_s256 should be ascribed to the effects of that bias. Buffy_s5e3&Pascal appears to contain a similar, but smaller, bias (compare training on this and testing on Buffy_s256 with training on this and testing on our_test). We do not have figures for Eichner and Ferrari's method trained on Buffy_s2to6 and tested on Buffy_s256. Note that: Buffy_s5e256_sub and Buffy_s5e3&Pascal_sub are subsets of 150 examples randomly chosen for each dataset.

Model		Test set		
		our_test	Buffy_s5e256	Pascal
	Train set			
Full model	our_train	0.663	0.623	0.627
	Buffy_s5e256_sub	0.583	0.676	0.625
	Buffy_s5e3&Pascal_sub	0.613	0.628	
Tree model	our_train	0.608	0.552	0.565
	Buffy_s5e256_sub	0.545	0.629	0.599
	Buffy_s3&Pascal_sub	0.565	0.596	
Eichner&Ferraris	Buffy_s5e2to6	0.557		0.675
	Buffy_s5e34&Pascal	0.559	0.801	

Felzenszwalb[2] to compute HOG features [7] for upper body and head detectors. Arm segment detectors use HOG features and self-similarity features (from [23], using the implementation of V. Gulshan). This detector does not distinguish between upper and lower arms, because locally they are similar in appearance. Upper arms can be difficult to detect, because there may be little contrast between the segment and the body. To overcome this difficulty, we also use a *virtual* upper arm detector, obtained by joining points nearby the top of the upper body segment to the elbows of nearby lower arm segments.

Lower arms can be strongly oriented (i.e. long and thin), and our arm detector may respond more than once to a lower arm in a lateral view. Extending the support of the detector does not help, unless one searches an impractical number of orientations. We deal with this by expanding the search space: we add new lower arms to the pool of detector responses, made by fusing nearby arm detections at the same orientation.

All part detectors are linear SVM trained on cropped parts from our dataset and from some of Buffy_s5e3 dataset. We bootstrap upper body and head detectors on a subset of background images, and lower arm detector on subset training images (regions outside the subject box). Table 1 summarizes part detector parameters.

[2] http://people.cs.uchicago.edu/~pff/latent/

Fig. 3. Examples of stick-figure, upper body parses of figures in our dataset produced by the full model trained on our dataset **top** row, our tree model **top-center** and the code of Eichner *et al.***bottom center** (trained on buffy_2to6) and **bottom** (trained on buffy_3&4 and pascal), all applied to our dataset. Red: upper body; Green: head; Blue-Purple: left upper/lower arm; Green-Yellow: right upper-lower arm. Note doubled arms produced by the tree model and a tendency for Eichner *et al.*to produce hanging arms, most likely a result of the strong geometric prior in their training datasets.

2.4 Features

We use a binning scheme, after [18]. Binning takes a feature such as distance and quantizes the range to a set of discrete bins, then sets the bin into which a value falls to be one and all others zero. We find it helpful to antialias by splitting the vote among nearby bins.

Unary features are the detector score at the detection labelled with a part label (converted to a probability using the method of [17]), and a binned vector representing the part length. For virtual upper arms, we have no detector score and instead use the value of the detector response at the lower arm used to create the virtual upper arm.

Binary features are different for different pairs of parts. We use six parts: upper body (from chest to navel), head (from top forehead to chin), left upper arm (LUA), left lower arm (LLA), right upper arm (RUA), and right lower arm (LLA). For each pair, we compute features from *distance*, *appearance*, *angle*, or *overlap*, according to the scheme of table 2.

Distance features for a pair of segments consist of a binned vector representing distance between endpoints, concatenated with the actual distance. The **comparative appearance feature** is formed from a set of appearance vectors. The appearance vectors consist of normalized color histograms, normalized Gabor filter histograms [4], and a histogram of textons [31]. For each type of appearance vector, we compute the χ^2 distance between the vectors corresponding to the two segments to be compared. For speed, integral images of appearance features are precomputed over reoriented images. **Angle features** are given by a binned angle vector representing signed angle from segment 1 to segment 2 in the range $(-90^o..90^o)$ for the head-torso pair, and $(-180^o..180^o)$ for all others. **Overlap features** give the ratio of endpoint distances to segment length, with the ratio computed for each segment. There are a total of 707 features.

3 Experimental Results

We compare a full model to a tree model on three datasets, described below. The full model is trained as above. The tree model is trained in the same way, but with the weights of features representing relations not in the tree clamped at zero. Inference (and so training) of the tree does not require a polishing step, because dynamic programming is exact. The tree is the usual kinematic tree (figure 1).

3.1 Dataset

We describe results on three datasets. The first is the Buffy dataset of [9], in various partitions. This dataset has little variation in layout (figure 2). The second is the subset

Fig. 4. Examples of stick-figure, upper body parses of figures in the buffy produced by the full model trained on ours **top** row, and our tree model trained on ours **bottom**. Red: upper body; Green: head; Blue-Purple: left upper/lower arm; Green-Yellow: right upper-lower arm. Note doubled arms produced by the tree model, and the strong tendency for poses to have hanging arms.

Fig. 5. Examples of stick-figure, upper body parses of figures in our dataset produced by the full model trained on our dataset **top** row, our tree model **top-center** and the code of Eichner *et al.***bottom center** (trained on buffy_2to6) and **bottom** (trained on buffy_3&4 and pascal), all applied to our dataset. Red: upper body; Green: head; Blue-Purple: left upper/lower arm; Green-Yellow: right upper-lower arm. Note doubled arms produced by the tree model and a tendency for Eichner *et al.*to produce hanging arms, most likely a result of the strong geometric prior in their training datasets.

of Pascal images marked up and released by [3]. Human parsing results are usually intended to drive activity recognition, which is at its most interesting when the body takes unusual postures. Methods that work well on a dataset with a strong spatial bias may do so because (say) they are particularly good at some common poses; such methods may not be useful in practice. For this reason, we have created a third dataset of 593 images (346 training, 247 test), marked up with stick figures by hand. This dataset is built to have aggressive spatial variation in configuration (figure 2).

3.2 Results

We follow convention and measure performance with PCP (Percentage of Correctly estimated body Parts). In this method, a segment is correct if its endpoints lie within 50% of the length of the ground truth from the annotated location [9]. Since our method produces one parse for each upper body detector response, we apply non-maximum suppression to the score, to prevent effects from multiple nearby upper body detector

responses. As in Eichner *et al.* [3], we evaluate PCP only for stickmen whose upper body response overlaps the correct upper body.

On Buffy and Pascal, our method obtains 62.3% and 62.7%, respectively (compare 80.3% and 72.3%, Eichner *et al.* [3]). However, there are two difficulties with these dataset (especially for Buffy), both potentially quite serious. First, there is little variation in pose. Figure 2 shows a scatter plot of ground truth head and arm segments for overlaid upper body segments. Head, upper arm and lower arm segments all have relatively little scatter — most figures are upright. Second, the contrast for many frames is relatively low. Both issues suggest that careful detector engineering will produce improvements in performance by overcoming contrast difficulties. Detector engineering is a valuable contribution which is responsible for all advances on the buffy dataset, but it will not identify better or worse modelling techniques. Because the spatial configuration varies so little in the Buffy dataset, comparisons of modelling techniques on this dataset should be approached with caution.

On all three datasets, the full model significantly outperforms the tree model (table 3). This is most likely because appearance consistency constraints between upper arms help overcome relatively low contrast at the boundary. Typical results suggest that improvements occur because consistency in appearance (the left arm must look like the right) is a cue that helps parse, and possibly because the model is spatially more rigid than a tree model (figure 5). The value of these cues outweighs the cost of approximate inference and approximate learning. Our parser can be configured as a detector by applying non-maximum suppression to the parse score and thresholding.

4 Discussion

We have shown quantitative evidence that a full relational model of the body performs better at upper body parsing than the standard tree model, despite the need to adopt approximate inference and learning procedures. We have obtained our results on a new dataset where there is extensive spatial variation in body configuration. Our results suggest that appearance consistency constraints help localize upper arms better. Our method extends to a full body parse.

Acknowledgements

This work was supported in part by the National Science Foundation under IIS - 0534837 and in part by the Office of Naval Research under N00014-01-1-0890 as part of the MURI program, and in part by the Vietnam Education Foundation through a fellowship to Duan Tran. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the NSF, the ONR, or the VEF.

References

1. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: CVPR (2009)
2. Desai, C., Ramanan, D., Fowlkes, C.: Discriminative models for static human-object interactions (2010)

3. Eichner, M., Ferrari, V.: Better appearance models for pictorial structures. In: British Machine Vision Conference (2009)
4. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Learning to describe objects. In: CVPR (2009)
5. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. IJCV 61(1), 55–79 (2005)
6. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient matching of pictorial structures. In: CVPR (2000)
7. Felzenszwalb, P.F., McAllester, D.A., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR (2008)
8. Fergus, R., Perona, P., Zisserman, A.: Object Class Recognition by Unsupervised Scale-Invariant Learning. In: CVPR (2003)
9. Ferrari, V., Marin, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR (2008)
10. Ioffe, S., Forsyth, D.: Finding people by sampling. In: ICCV, pp. 1092–1097 (1999)
11. Ioffe, S., Forsyth, D.: Human tracking with mixtures of trees. In: ICCV, pp. 690–695 (2001)
12. Jiang, H.: Human pose estimation using consistent max-covering. In: ICCV (2009)
13. Jiang, H., Martin, R.: Global pose estimation using non-tree models. In: CVPR (2008)
14. Johnson, S., Everingham, M.: Combining discriminative appearance and segmentation cues for articulated human pose estimation. In: MLVMA 2009 (2009)
15. Mori, G., Malik, J.: Estimating human body configurations using shape context matching. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2352, pp. 666–680. Springer, Heidelberg (2002)
16. Mori, G., Ren, X., Efros, A.A., Malik, J.: Recovering human body configurations: Combining segmentation and recognition. In: CVPR (2004)
17. Platt, J.: Probabilities for sv machines. In: Advances in Neural Information Processing (1999)
18. Ramanan, D.: Learning to parse images of articulated bodies. In: Advances in Neural Information Processing (2006)
19. Ramanan, D., Forsyth, D., Barnard, K.: Building models of animals from video. PAMI 28(8), 1319–1334 (2006)
20. Ratliff, N., Bagnell, J.A., Zinkevich, M.: Subgradient methods for maximum margin structured learning. In: ICML 2006 Workshop on Learning in Structured Output Spaces (2006)
21. Ronfard, R., Schmid, C., Triggs, B.: Learning to parse pictures of people. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2353, p. 700. Springer, Heidelberg (2002)
22. Sapp, B., Jordan, C., Taskar, B.: Adaptive pose prior for pictorial structure. In: CVPR (2010)
23. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: CVPR (2007)
24. Sigal, L., Black, M.J.: Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In: CVPR (2006)
25. Song, Y., Feng, X., Perona, P.: Towards detection of human motion. In: CVPR, pp. 810–817 (2000)
26. Taskar, B.: Learning Structured Prediction Models: A Large Margin Approach. PhD thesis, Stanford University (2004)
27. Taskar, B., Lacoste-Julien, S., Jordan, M.: Structured prediction via the extragradient method. In: Neural Information Processing Systems Conference (2005)
28. Tian, T.-P., Sclaroff, S.: Fast globally optimal 2d human detection with loopy graph models. In: CVPR (2010)

29. Tran, D., Forsyth, D.: Configuration estimates improve pedestrian finding. In: Advances in Neural Information Processing (2007)
30. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. Journal of Machine Learning Research (JMLR) 6, 1453–1484 (2005)
31. Varma, M., Zisserman, A.: A statistical approach to texture classification from single images. Int. J. Computer Vision 62(1-2), 61–81 (2005)
32. Yao, B., Fei-Fei, L.: Model mutual context of object and human pose in human-object interaction activities. In: CVPR (2010)