Human Pose Estimation using a Joint Pixel-wise and Part-wise Formulation

L'ubor Ladický
ETH Zürich, Switzerland
lubor.ladicky@inf.ethz.ch

Philip H.S. Torr
Oxford Brookes University, UK
philiptorr@brookes.ac.uk

Andrew Zisserman
University of Oxford, UK
az@robots.ox.ac.uk

Abstract

Our goal is to detect humans and estimate their 2D pose in single images. In particular, handling cases of partial visibility where some limbs may be occluded or one person is partially occluding another.

Two standard, but disparate, approaches have developed in the field: the first is the part based approach for layout type problems, involving optimising an articulated pictorial structure; the second is the pixel based approach for image labelling involving optimising a random field graph defined on the image.

Our novel contribution is a formulation for pose estimation which combines these two models in a principled way in one optimisation problem and thereby inherits the advantages of both of them. Inference on this joint model finds the set of instances of persons in an image, the location of their joints, and a pixel-wise body part labelling.

We achieve near or state of the art results on standard human pose data sets, and demonstrate the correct estimation for cases of self-occlusion, person overlap and image truncation.

1. Introduction

The objective of this work is to detect and estimate the pose of humans in single images. This problem has a long history in computer vision, and a dominant method is tree-structured pictorial structures [11, 12, 13, 16, 31]. These proceed by searching for the most probable location of body parts, estimating a per pixel cost for each part, and combining the costs using dynamic programming over the tree structure graph. Whilst pictorial structures have enabled significant progress they have several problems, including: (a) failing when there is more than one person in the scene, if those people are overlapping; (b) over-counting of evidence [26] – pixels can contribute more than once to the cost function and hence multiple parts can explain the same image area; (c) failing to model the background, resulting in evidence being ignored as only pixels which are covered by the model contribute to the overall probability of a given limb configuration; (d) failing due to self or other types of occlusion.

Recent work has attempted to overcome these problems, for example by enforcing consistency of ensembles of parts [15, 23, 29] or eschewing the pictorial-structure formulation by directly learning poselets for human parts tightly clustered in both appearance and configuration spaces [3]. However although these approaches allow for more accurate localization of joints, and deal to some extent with occlusion, they do not deal with problems (a) and (c). In order to deal with (c), and also with self-occlusion, the work of [6] introduced a weak background model combined with a tight model of the human foreground. The resulting method is one of the first to deal convincingly with the problem of self occlusion and clearly demonstrates the benefit of a background model. Others have proposed modelling dependencies and relationships between multiple people [9], which addresses problem (a), and methods for efficiently sampling from pictorial structures [6, 11, 19].

We take inspiration from these approaches and also leverage recent work on semantic segmentation [18], in particular where the semantic classes correspond to human body parts (arms, torso, etc) [24], to make the following contributions: (i) a global energy formulation that combines the advantages of the flexible mixtures-of-parts model [31] with those of pixel-wise labelling methods [17, 24, 25] to explain the background and foreground together (section 2); (ii) an efficient algorithm for proposing candidate human poses in an image, ensuring both coverage and diversity (section 3); and (iii) a formulation of the energy potentials that internally performs non-maxima suppression, induces layout consistent solutions, and can deal with partial occlusion or self-occlusion (section 4).

The outcome is the set of instances of persons in an image with the location of their joints, *and* the pixel-wise labelling (segmentation) of each of their body parts. Our work is a continuation of the theme of combining segmentation and human pose estimation [5, 12, 28].

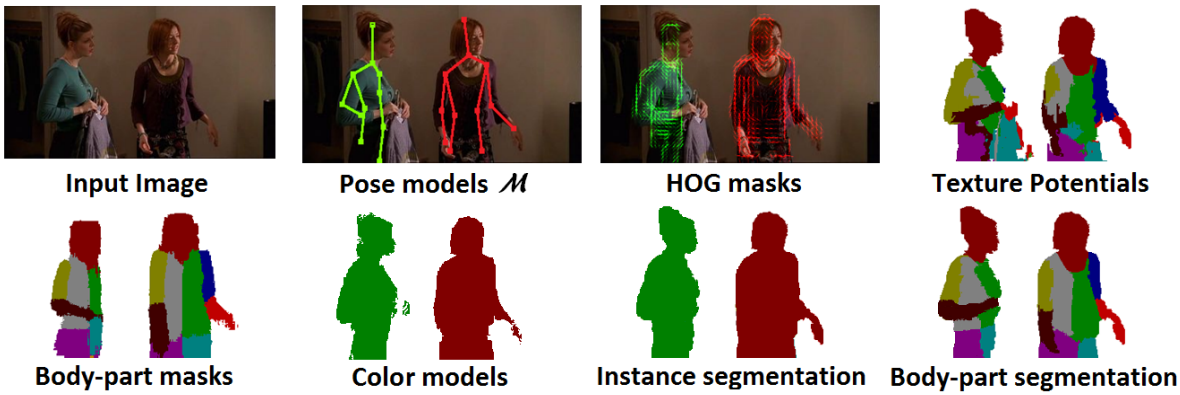

Figure 1. **The output of each component of the energy for an optimal Solution.** The mixture-of-parts model is used to produce the set of candidates. This figure shows two candidates that end up in the final solution, their HOG masks, body-part masks, instance color models, the result of the texture component and final instance and body-part segmentation. Candidates consist of 16 joints (some of them could be invisible) and the pixel-wise labelling of 11 body-part labels and background. The body-part color coding is described in Figure 2. **Best viewed in color.**

2. A joint pixel-wise and part-wise model

In this section we introduce and overview the model, which is specified by a single energy function.

Our goal is three fold: to assign every pixel to a body part of an instance of a person or to the background; for each instance, to estimate the body layout in terms of joint positions and body parts and specify their visibility; and thirdly, to determine the number of instances. The first goal is close to the traditional labelling problem of semantic segmentation, the second is close to the pose estimation which typically uses tree-structured pictorial structures.

In more detail, the method should predict a set of instances (subset of the set of candidates), consisting of their pose (joint positions and body parts). Correspondingly, a pixel in the image $\mathbf{x}$ is labelled by the instance and the body part that overlaps it. The notation $x_j = (x_j^{inst}, x_j^{part})$ indicates that the pixel x_j has two labels: one, $x_j^{inst} \in \mathcal{M} \cup \{B\}$, labelling the instance number from the set of estimated instances $\mathcal{M}$ or background B, and the other $x_j^{part} \in \mathcal{L}_{part} \cup \{B\}$, labelling which part from the set of parts $\mathcal{L}_{part}$ or background B overlaps it.

The model is fitted by minimizing an energy consisting of four components:

$$\begin{aligned} E(\mathcal{M}, \mathbf{x}) &= E^{HOG}(\mathcal{M}, \mathbf{x}^{inst}) + \lambda^{MASK} E^{MASK}(\mathcal{M}, \mathbf{x}) \\ &+ \lambda^{COL} E^{COL}(\mathcal{M}, \mathbf{x}^{inst}) + \lambda^{TEX} E^{TEX}(\mathbf{x}^{part}). \end{aligned} \quad (1)$$

where λ^{TEX}, λ^{MASK}, λ^{COL} are scalar weights.

We describe the components and their computation in detail below (section 4), but for the moment first illustrate their roles. The HOG component $E^{HOG}(.)$ models the contribution of a pixel to an instance of an articulated model. For a non-overlapping set of models this component corresponds to the negative sum of responses of a tree-structured mixture-of-parts pictorial structure model of [31]. The Body Part Mask component $E^{MASK}(.)$ links the model with its body-part labelling: given a set of joint positions from an instance of the model M, the body part mask is a (soft) assignment of pixels to mattes of the body parts, e.g. specifying the position and width of the arm. The Color Model component $E^{COL}(.)$ is a Gaussian mixture color model for the foreground and background built by thresholding the body part masks for the model. Up to this point the formulation is similar to PoseCut [5]. The final term, the Texture component $E^{TEX}(.)$, is a semantic segmentation of the image into body parts and background. It is computed independently of the instances and contributes information on the appearance and shape of body parts. optimization labels the pixels (with the instances and parts) and takes account of the costs from the four terms. The outputs of the components are illustrated in figure 1, where it can be seen how the texture component can contribute additional information over that provided by the instance segmentation from the color model.

The optimization proceeds by first proposing a number, N, of *candidates* for the instances using the mixture-of-parts model of [31]. Some of these candidates will survive and appear in the final solution, and the ones that do will have led to the minimal energy when all the components and interactions between instances have been taken into account (during inference).

In the following section we describe the method for generating the candidate instances, and then give the details of the component computation and inference method in section 4.

3. Efficiently Generating Pose Candidates

We would like to find a set of candidates – local optima, such that they cover all the persons in the image and all their possible poses. A proposal algorithm for this task has already been given in [19] for the mixture-of-parts model which iteratively estimates the (approximately) next best solution by examining the max-margin tables and then restricting the search space for the next iteration. However, the complexity of this procedure grows at least linearly with the number of estimated candidates. In practice if there are e.g. 5 people present, to capture all possible poses a large number, N, of candidates is required ($N = 200$ in our experiments) and the running time of this method is too large. In contrast we propose a method which takes only slightly more time to find a large number of candidates than to find just the best one per root node.

In [19], two poses are considered different if at least one part is sufficiently far from the corresponding part of the other pose. We take an alternative approach; because the relative location and orientation of joints is modelled in the mixture-of-parts model using T *types* of joints and the search space for a given type is restricted only to a small neighbourhood, we relax the matching constraint and consider two detections different if they either differ in scale, at least one part-type, or their root-nodes are sufficiently far from each other. To increase the chance of capturing all instances we restrict ourselves to the search for the best solutions with at most K candidates ($K = 8$ here) with the same root node.

Typically the inference methods for graphs with the structure of a tree are solved using dynamic programming [31]. Starting from the leaf nodes going towards the root node for each location and type of the part the best locations and types of its children are estimated. To find the best K candidates differing by at least one type of a child, we need to estimate for each location and type of the part its top K constellation (types and locations) of all its children. In the first step we find the best location of a child for each type of a child, and take the top K solutions for this location and type. This step is only approximate; these K solutions are only a good approximation of the real top K solutions, which can be obtained by merging all lists for each location given the type of a child. Thus, we get T lists (one for each type) with K ordered solutions each. These lists are merged and top K of these TK solutions are kept. This can be done in $O((K + T) \log T)$ by iteratively estimating the next top solution and keeping the set of T lists ordered by their top solution. In the second step the parent has to merge its response for each type with the top K solutions of each child. Thus, we need to find the top K sums from K lists of K items, taking one item from each list. This can be solved [2] in $O(P_c K \log K)$, where P_c is the number of children of the parent. Each node remembers the *back tracks* by remembering the indexes of the subtrees of each child without any need of copying whole trees of solutions. The final set of candidates is obtained by merging whole trees of solutions of suppressed root nodes.

In practice for $T = 5$ and $K = 8$ (as used in the experiments) the running time of brute force search for the best location of a child of each type is much more expensive (especially in case of the sub-cell accuracy) than the sorting and merging steps together and the algorithm takes only $1.5\times$ more than the search for just one best solution per root node. This is mainly because the pairwise deformation costs are calculated on the fly and not kept in memory.

To obtain also candidates with hidden parts, the set of types is altered with an additional $hidden$ type, corresponding to the invisible joint whose children are also hidden. Its response is a constant and takes no deformation cost. Using this hidden type allows for candidates which have certain joints either outside of an image, occluded by another object or person, or self-occluded.

4. Implementation Details

In this section we describe how each component of the model energy (1) is computed, and then the inference method for the model.

4.1. The HOG component

Our formulation is built upon the state-of-the-art mixture-of-parts model [31]. The classifier takes the form:

$$H(\mathbf{P}, \mathbf{t}) = \sum_{p \in \mathbf{P}} \sum_{c \in p} \mathbf{w}_c^{(p,t_p)} \cdot \mathbf{h}(c) + R(\mathbf{P}, \mathbf{t}), \qquad (2)$$

where $\mathbf{h}(c)$ is the HOG feature vector for a cell c, $\mathbf{P}$ is the set of joints (parts) p, $\mathbf{t}$ is the vector of the *types* of joints modelling orientations, where for each part p and its type (orientation) t_p there is a different weight vector $\mathbf{w}^{(p,t_p)}$, and $R(\mathbf{P}, \mathbf{t})$ is the layout consistency term, modelling the likelihood of combinations of types and their relative locations. The weight vectors $\mathbf{w}^{(p,t_p)}$ and the parameters of the layout consistency term are learnt using a linear SVM.

Our goal here is to transform the cell-wise mixture-of-parts model to a pixel-wise formulation. We do this in two stages, by first rewriting the form of (2), and then using this to define the pixel wise energy. By extending the definition of the weight vectors as $\mathbf{w}_c^{(p,t_p)} = 0$ if $c \notin p$ the classifier response can be rewritten as:

$$H(\mathbf{P}, \mathbf{t}) = \sum_{c \in \bigcup \mathbf{P}} \mathbf{w}_c^{\mathbf{P},\mathbf{t}} \cdot \mathbf{h}(c) + b^{\mathbf{P},\mathbf{t}}, \qquad (3)$$

where $\mathbf{w}_c^{\mathbf{P},\mathbf{t}} = \sum_{p \in \mathbf{P}} \mathbf{w}_c^{(p,t_p)}$ and $b^{\mathbf{P},\mathbf{t}} = R(\mathbf{P}, \mathbf{t})$. Thus, given the set of latent parameters (locations and types of joints) the form of the classifier is the same as the form of

a standard HOG detector [7]. Each candidate model m is defined by the locations and types of parts $m = (\mathbf{P}, \mathbf{t})$ and has its own associated HOG weight vector $\mathbf{w}^m$ and bias b^m.

We now define the pixel-wise cost in terms of (3) as

$$E^{HOG}(\mathbf{x}^{inst}) = \sum_{j \in \mathcal{V}} \psi^{HOG}(x_j^{inst}) + \sum_{m \in \mathcal{M}} (-|c^m| b^m) \delta(m) \tag{4}$$

where $\mathcal{V}$ are the pixels, and $\delta(m)$ indicates the presence of a model m in the labelling (i.e. $\delta(m) = 1$ if $\exists j \in \mathcal{V}$, s.t. $x_j^{inst} = m$) and $\delta(m) = 0$ otherwise), $|c^m|$ is the size of the HOG cell in pixels and

$$\psi^{HOG}(x_j^{inst}) = \begin{cases} -\mathbf{w}^m_{c^m(j)} \cdot \mathbf{h}(c^m(j)) & \text{if } x_j^{inst} = m \in \mathcal{M} \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $c^m(j)$ is the corresponding cell of a model m for a pixel j. Note, the first term of (4) transfers from HOG cells of the candidate models to the pixels they cover, and the second term adds a contribution to the cost if the model covers any pixel in the image.

To understand the design of these costs, note that for an unoccluded person this energy is $-|c^m|(\mathbf{w}^m \cdot \mathbf{h}(\mathbf{c}^m) + b^m)$. We restrict the weights to be non-negative, so a candidate will appear in the solution if the mixture-of-parts classifier response is positive w.r.t. the bias b^m (note, the bias is negative in general as it has to prevent false negative human detections across the image). Indeed, the HOG model can be interpreted as the scalar product for each cell, $\mathbf{w}_c \cdot \mathbf{h}(c)$, providing evidence for the detection hypothesis, and the bias b as the threshold needed to accept the hypothesis. The evidence should not be taken into account from occluded parts of the object, and thus a significantly occluded object would be unable to provide sufficient evidence, larger than the threshold. The restriction, that each pixel may belong to at most one candidate, leads to a non-maxima suppression of the candidates.

4.2. The Body part mask component

Given the location of the joints, it is possible to predict the body part segmentation to a good approximation. We achieve this by learning a classifier to predict whether each pixel belongs to each body part given the location of all joints. The output is a soft body-part likelihood for each model m. For a given model m the multi-class classifier $H^m_{part}(j)$ predicts the likelihood that each pixel j has a body-part label $b \in \{B\} \cup \mathcal{L}_{part}$.

The intuitive way to incorporate this potential in the random field framework would be to add it as a simple unary potential, assigning a cost $H^m_B(j) - H^m_{x_j^{part}}(j)$ if the pixel j takes a model label m and the body-part label x_j^{part}. However, that would lead to an undesired bias, caused by each pixel, where the most probable label is not background. Suppose we use only the HOG and body part mask components. Because the mask is independent of the image data, it should not change the ordering of the HOG candidates. In the other words, if the labelling agrees with the body-part mask prediction, the energy for each candidate should be 0. Thus, we need to balance the bias of all foreground pixels and the unbiased potential takes the form:

$$E^{MASK}(\mathcal{M}, \mathbf{x}) = \sum_{j \in \mathcal{V}} (H^m_B(j) - H^m_{x_j^{part}}(j)) + \sum_{m \in \mathcal{M}} C(m) \delta(m), \tag{6}$$

where $C(m)$ is defined as:

$$C(m) = \sum_{i \in \mathcal{V}} \max_{p \in \{B\} \cup \mathcal{L}_{part}} (H^m_p(j) - H^m_B(j)). \tag{7}$$

If the final labelling agrees with the most probable body part mask, then it sums up to zero; if some pixels do not agree, they are penalized based on the difference of body-part likelihoods for the estimated and present label.

The classifier is based on the standard multi-class boosting approach of [27], where the classes here corresponds to the set of body-part label and background. The feature vector computed for each training pixel i consists of signed distances $d(i)$ in x and y from each joint, and signed distances from each limb and axis of a limb. All distances are relative to the size of the object determined by the longest limb (all limbs are about the same size). Because the joints (and limbs) may be occluded, we double the number of the decision stumps $d(i) \geq \theta$ and $d(i) < \theta$ used as the weak features, where both conditions are by definition not satisfied for an occluded part or limb. For the weak classifier, the tests include, e.g. is there a shoulder further than θ from this point, and is there a shoulder closer than θ from this point, so that the algorithm can distinguish between the cases, when the *shoulder* is visible and when it is not. In the evaluation stage, the classifier predicts the likelihood of a pixel taking each body-part label or background.

4.3. The Color component

Color component ensures the solutions, where the color models of the foreground and the background are different, are preferred. It is self-trained for each instance using Gaussian mixture model [21] initialised using the mask estimated as in section 4.1. The associated energy takes the form:

$$E^{COL}(\mathcal{M}, \mathbf{x}) = \sum_{j \in \mathcal{V}} \psi^{COL}(x_j^{inst}) + \sum_{m \in \mathcal{M}} C(m) \delta(m), \tag{8}$$

where

$$\psi^{COL}(x_j^{inst}) = \begin{cases} -\log \frac{p_j^m(F)}{p_j^m(B)} & \text{if } x_j^{inst} = m \in \mathcal{M} \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

$p_j^m(F)$ and $p_j^m(B)$ are the probabilities of foreground respectively background, and $C(m) =$

$\sum_{j \in \mathcal{V}} \max(\log(p_j^m(B) - \log(p_j^m(F)), 0)$ is the balancing term, removing the self-training bias as in section 4.1.

4.4. The Texture component

The Texture component consists of potentials used for the semantic segmentation problem, the multi-feature TextonBoost [17, 25], the body-part super-pixel terms as in [17] and the pair-wise term is the usual contrast dependent Potts model. Even though the performance of this component is not very high on its own, it can reliable distinguish between torso and arms and resolve several ambiguities of the mixture-of-parts model.

It consists of potentials used for the semantic segmentation problem: the multi-feature TextonBoost [17, 25], the body-part super-pixel terms as in [17] and the pair-wise term is the usual contrast dependent Potts model.

4.5. Inference

We wish to minimize the energy (1) in order to determine: the set of instances $\mathcal{M}$ with their layout (joints, parts), as well as a pixel-wise labelling of the image according to whether the pixels belong to a part (e.g. lower arm of instance 1) or background. The optimization cannot be carried out directly, and we proceed in two stages: first, finding the number and joint position of the instances; and second, with this restricted set of label possibilities, determining the best pixel labelling (i.e. assigning the pixels to parts or background).

In the first stage we have N human pose candidates (obtained as described in section 3). For each candidate we compute the potentials E^{HOG}, E^{MASK} and E^{COL}. The potential E^{TEX} is independent of the candidates and is evaluated once for the entire image. We start by labelling everything as background and iteratively adding the next best candidate. The quality of each candidate is determined by calculating the energy after α-expansion [4] over the 11 body-part labels and background (i.e. 12 labels in total). The first stage is complete once no more instances can be added to the solution without increasing its energy.

In the second stage the optimization is over all the selected candidates to refine the solution. Again, α-expansion is used, but now over a remaining label set of size $1 + 11\times$ the number of selected candidates. The α-expansion is repeated unil convergence. This procedure can determine partially overlapping instances and self-occlusion.

5. Model Evaluation

We performed experiments on two standard pose estimation data sets – the Buffy [12] and Image Parse [31] data sets.

5.1. Data and annotation

The Buffy data set. We use the standard Buffy data set of [12] consisting of 748 images from episodes $s5e2$ – $s5e6$, with episode $s5e3$ used for training, episode $s5e4$ for validation and episodes $s5e2$, $s5e5$ and $s5e6$ for testing.

For the purposes of training and evaluation, we add the following additional annotation to each image: all visible locations of 16 joints (head-top, head-bottom, left/right shoulders, elbows, wrists, waists, hips, knees and ankles) for *all* instances in an image; and a pixel-wise instance and 12-label body part (background, head including neck and hair, left/right torso, upper arms, forearms, thighs, lower legs) segmentations, i.e. the joints, instance and body-part pixel-wise labellings are provided for all images. Largely occluded persons are marked as hard and ignored for the evaluation. Figure 2 shows samples with this annotation.

Model training and execution. Each individual component of the model (the HOG, texture, color and body part mask potentials) is trained separately using this annotation. The HOG component is trained using the approach of [31], the texture component using the learning methods described in [17], the color model using a mixture of 10 Gaussians as in [21], and body part mask using the method described in section 4.1. The top 200 candidates are used in the experiments, 8 at most with the same scale and the same root node. The weights λ^{TEX}, λ^{COL} and λ^{MASK} are hand-tuned on the validation set using grid search. With these parameters the optimisation takes approximately 4 minutes per image on one 3 GHz core.

The Image Parse data set. The Image Parse [31] data sets consist of 305 images; each containing only one person and a labelling of 14 joints. 100 images are used for training and the others for testing. The data set does not contain pixel-wise labellings, so the texture and foreground mask potentials trained on the Buffy data set are also used for this data set.

5.2. Evaluation measures

We assess the performance of the algorithm using three standard performance measures: the PCP measure for human layout prediction as introduced by [12], this assesses the part-wise aspect of the model; and two measures of semantic segmentation accuracy to assess the pixel-wise aspect – these are the pixel-wise recall per class (i.e. the proportion of the pixels of each class that are correctly classified out of the pixels belonging to that class), and the intersection over union per class as used in the VOC challenge [10]. Since most pixels are background, the average of the pixel-wise recall over classes is more affected by mislabelling a body part pixel as background than the other

Figure 2. **Additional labelling provided for the Buffy data set.** Different colors correspond to different instances. In cases where the instances are highly occluded (such as C) or difficult to distinguish, the joints are not labelled, and the body-part pixels are labelled as hard (black) and ignored for training and evaluation. Some of joints are labelled as half-visible (sometimes because they are too close to the boundary) and ignored for evaluation too. Limbs with at least one such joint are shown white.

Method	Head	Torso	U Arms	L Arms	Overall
[8]	83.4	84.0	70.5	50.9	68.2
[22]	81.9	85.1	81.1	53.6	72.8
[15]	99.6	99.6	93.2	60.6	84.5
[31]	98.9	99.6	95.1	68.5	87.6
Our method (2 labels)	**100.0**	**100.0**	97.1	73.9	90.3
Our method (12 labels)	**100.0**	**100.0**	**97.5**	**75.4**	**90.9**

Table 1. **PCP performance on the Buffy data set**. A comparison with the state of the art. The joint formulation leads to a significant improvement mainly on the lower arms, where the method of [31] struggles to get a good score. The performance was evaluated using the code of [12] on the original upper-body ground truth.

way around. A consequence of this class bias is that over-segmenting the parts (e.g. people appear fatter than veridical) scores more highly than under-segmenting. The intersection over union does not suffer from this problem. Experiments on the Buffy data set were carried out for both 2 label (person & background) and 12 label (body parts & background).

5.3. Results

We compare our method to the state of the art methods in both the part-wise and segmentation aspects.

For the pictorial structure measures, the comparison to the state-of-the-art methods for the Buffy data set in the loose-PCP measure is given in table 1, and for the Image Parse data set for the strict- and loose-PCP measures in table 2. See further discussion on the evaluation measures in [20]. The incorporation of all components leads to a significant improvement on the Buffy data set, however, the method did not improve on the Image Parse data set, probably because the texture and mask potentials were trained on a different data set with different distribution of poses.

Method	Head	Torso	U Legs	L Legs	U Arms	L Arms	Overall
[1] (strict)	75.6	81.4	63.2	55.1	47.6	31.7	55.2
[14] (strict)	76.1	85.4	73.4	65.4	**64.7**	**46.9**	**66.2**
[31] (strict)	**77.6**	82.9	69.0	63.9	55.1	35.4	60.7
[20] (strict)	73.7	**88.8**	**77.3**	**67.1**	53.7	36.1	63.1
Our method (strict)	75.1	83.9	71.0	63.9	56.8	33.9	61.0
[31] (loose)	**93.2**	97.6	83.9	75.1	72.0	**48.3**	74.9
[20] (loose)	92.5	**98.9**	**90.1**	**79.6**	68.8	48.1	**76.5**
Our method (loose)	92.7	98.0	86.1	75.1	**72.9**	47.6	75.4

Table 2. **Strict- and loose-PCP performance on the Parse data set** (implementations of [20] and [31] respectively). The performance of our method is not significantly different from the state-of-the-art. There is less opportunity for the joint model to show its power as images do not contain overlapping or occluded people.

For the semantic segmentation performance, table 3 gives a quantitative comparison of our results on the Buffy data set for the two measures with the results from [17] and various subsets of the components. Our method significantly outperforms the baseline texture component alone, and any combination of components. The result obtained by the mixture-of-parts model mapped into segmentation (HOG + Mask) does not use the pixel-wise data and thus can not get the person boundaries exactly. Adding a color model resolves this problem. Texture potentials are good at distinguishing between limbs and torso, and thus help to resolve ambiguities in estimation of joints and their visibility. See the discussion about the pros, cons and failure cases in the caption of figure 3 and figure 4.

6. Conclusion

In this paper we have shown that, given appropriate training, it is possible to achieve Kinect style body part labelling and layout in color images (despite not having depth information). Furthermore, we have for the first time covered the case of multiple, possibly interacting, human instances in quite varied and unconstrained poses. The formulation of a joint model covering foreground and background has effectively dealt with all of the problems we listed in the introduction for pictorial-structures, e.g. over counting of data and failure to take account of background evidence. Furthermore our combined method provides state of the art results both for pose and segmentation. This richer output opens up new applications for human parsing and segmentation algorithms, e.g. for analysis of clothing [30].

Acknowledgements. We are grateful for financial support from ERC grant VisRec no. 228180.

	2 labels			12 labels												
method	Average	Background	Person	Average	Background	Head	R Torso	L Upper Arm	R Upper Arm	L Thigh	R Thigh	L Torso	L Forearm	R Forearm	L Lower Leg	R Lower Leg
Texture [17] (recall)	87.3	88.1	86.6	38.7	91.2	**77.0**	47.4	32.7	41.3	27.0	19.7	48.7	35.8	41.8	0.2	1.2
HOG + Mask (recall)	85.8	85.2	86.5	48.4	86.3	61.2	70.1	59.2	56.4	36.9	45.2	67.4	40.5	38.5	9.5	9.6
HOG + Mask + Color (recall)	88.4	90.3	86.5	54.8	91.4	61.4	**72.7**	63.0	61.7	**47.7**	**54.0**	**70.7**	43.9	41.9	**25.6**	**23.0**
All (recall)	**92.3**	**95.3**	**89.2**	**55.5**	**96.8**	64.5	71.9	**63.5**	**63.0**	46.5	53.6	70.6	**46.5**	**45.0**	23.3	20.4
Texture [17] (int / union)	72.5	85.0	60.0	26.2	86.3	57.5	25.0	23.0	22.2	16.4	13.2	28.8	20.1	20.5	0.2	1.1
HOG + Mask (int / union)	69.0	82.2	55.7	30.1	82.6	41.5	36.8	30.5	33.8	23.8	26.4	36.5	18.0	19.5	5.6	6.0
HOG + Mask + Color (int / union)	75.3	87.1	63.5	37.1	87.6	49.4	44.1	38.7	42.1	30.7	33.8	44.3	22.6	23.8	14.1	14.2
All (int / union)	**84.3**	**92.7**	**76.0**	**42.7**	**92.8**	**59.3**	**50.5**	**46.3**	**49.6**	**35.5**	**39.7**	**50.7**	**28.1**	**29.5**	**15.3**	**15.2**

Table 3. Pixel labelling performance on the Buffy data set. Results are given for both recall and intersection over union measures. The first 3 columns correspond to the performance on the 2-label problem, and the remainder on the 12-label problem. The incorporation of pose into the random field framework leads to a significant improvement of the performance. The weights are optimised for the intersection over union measure, which is more suitable for this data set because of a significant imbalance of the body part classes (dominated by background). Surprisingly, the incorporation of texture potentials improved the intersection over union measure also for the lower legs, even though there is insufficient training data to learn them well. The improvement is mainly because the texture potentials give a much better definition of the instance boundary.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 2009.

[2] A. Babenko and V. Lempitsky. The inverted multi-index. In *CVPR*, 2012.

[3] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 2001.

[5] M. Bray, P. Kohli, and P. H. S. Torr. Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *ECCV*, 2006.

[6] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman. Upper body detection and tracking in extended signing sequences. *IJCV*, 2011.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[8] M. Eichner and V. Ferrari. Better appearance models for pictorial structures. In *BMVC*, 2009.

[9] M. Eichner and V. Ferrari. We are family: Joint pose estimation of multiple persons. In *ECCV*, 2010.

[10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results, 2011.

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, 2000.

[12] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, 2008.

[13] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *Transactions on Computers*, 1973.

[14] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010.

[15] L. Karlinsky and S. Ullman. Using linking features in learning non-parametric part models. In *ECCV*, 2012.

[16] M. P. Kumar, P. H. S. Torr, and A. Zisserman. OBJ CUT. In *CVPR*, 2005.

[17] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical CRFs for object class image segmentation. In *ICCV*, 2009.

[18] L. Ladicky, C. Russell, P. Sturgess, K. Alahari, and P. H. S. Torr. What, where and how many? Combining object detectors and CRFs. *ECCV*, 2010.

[19] D. Park and D. Ramanan. N-best maximal decoders for part models. In *ICCV*, 2011.

[20] L. Pishchulin, A. Jain, M. Andriluka, T. Thormaehlen, and B. Schiele. Articulated people detection and pose estimation: Reshaping the future. In *CVPR*, 2012.

[21] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, 2004.

[22] B. Sapp, A. Toshev, and B. Taskar. Cascaded models for articulated pose estimation. In *ECCV*, 2010.

[23] B. Sapp, D. Weiss, and B. Taskar. Parsing human motion with stretchable models. In *CVPR*, 2011.

[24] J. Shotton, A. Fitzgibbon, M. Cook, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[25] J. Shotton, J. Winn, C. Rother, and A. Criminisi. *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, 2006.

[26] L. Sigal and M. J. Black. Predicting 3d people from 2d pictures. In *AMDO*, 2006.

[27] A. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, 2004.

[28] H. Wang and D. Koller. Multi-level inference by relaxed dual decomposition for human pose segmentation. In *CVPR*, 2011.

[29] D. Weiss and B. Taskar. Sidestepping intractable inference with structured ensemble cascades. In *NIPS*, 2010.

[30] K. Yamaguchi, H. Kiapour, L. E. Ortiz, and T. L. Berg. Parsing clothing in fashion photographs. In *CVPR*, 2012.

[31] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.

Figure 3. **Qualitative results on the Buffy data set.** The first three columns are the pictorial structure pose, instance segmentation and body-part segmentation obtained using our method; the last three columns are the corresponding ground truth. The method can naturally handle multiple instances. The algorithm successfully estimated the large majority of the instances and the locations of their joints. Furthermore, there are examples of partial occlusion by another person (A right, C left), a background object (C left) and self-occlusion (B right, C right). Some of the images also demonstrate failings, including: a phantom hallucinated instance (F left), a missed part in an uncommon location (F right), incorrect visibility of the part (B right, H right), limb assigned to a wrong person (D left / right), a missed instance due to a large occlusion (I middle) or an insufficient number of candidates for images with too many instances (K middle).

Figure 4. **Qualitative results on the Image Parse data set.** Due to an insufficient size of images we run the experiments using only 2 labels (person vs background) in the segmentation body-part domain. We restricted the algorithm to report only a pose of a single person.