Articulated Pose Estimation using Discriminative Armlet Classifiers

Georgia Gkioxari[1], Pablo Arbeláez[1], Lubomir Bourdev[2] and Jitendra Malik[1]
[1]University of California, Berkeley - Berkeley, CA 94720
[2]Facebook, 1601 Willow Rd, Menlo Park, CA 94025
{gkioxari,arbelaez,malik}@eecs.berkeley.edu,lubomir@fb.com

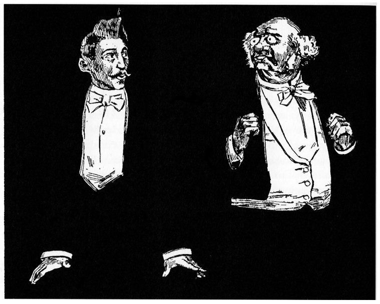

Figure 1. An example of an image where part detectors based solely on strong contours and edges will fail to detect the upper and lower parts of the arms.

Abstract

We propose a novel approach for human pose estimation in real-world cluttered scenes, and focus on the challenging problem of predicting the pose of both arms for each person in the image. For this purpose, we build on the notion of poselets [4] and train highly discriminative classifiers to differentiate among arm configurations, which we call armlets. *We propose a rich representation which, in addition to standard HOG features, integrates the information of strong contours, skin color and contextual cues in a principled manner. Unlike existing methods, we evaluate our approach on a large subset of images from the PASCAL VOC detection dataset, where critical visual phenomena, such as occlusion, truncation, multiple instances and clutter are the norm. Our approach outperforms Yang and Ramanan [26], the state-of-the-art technique, with an improvement from 29.0% to 37.5% PCP accuracy on the arm keypoint prediction task, on this new pose estimation dataset.*

1. Introduction

Suppose our goal is to find the arms and hands of the two gentlemen in Fig 1. We might aim to do so by fitting a stick figure model in one of its numerous manifestations [1, 7, 9, 19, 20] by detecting rectangles in the upper and lower parts of the arms. But are there any contours there? It seems clear enough that how we detect the arm configurations of these people is from the position of the head and the hands. And that information is enough to generate a very crisp prediction of the various joint locations.

This perspective, pose estimation as holistic recognition, can be found in papers from nearly a decade ago. Mori and Malik [16] matched whole body shapes of figures to exemplars using shape contexts and then transferred keypoint locations from exemplars to the test image. Shakhnarovich *et al.* [22] recovered the articulated pose of the human upper body by defining parameter-sensitive hash functions to retrieve similar examples from the training set. Our position is that these approaches were philosophically correct, but their execution left much to be desired. The classifiers used were nearest neighbors and we have evidence that discriminative classifiers such as support vector machines (SVMs) typically outdo nearest neighbor approaches [15]. The features used were simple edges, and again over the last decade we have seen the superiority of descriptors based on Histograms of Oriented Gradients (HOG) [5] in dealing with clutter and capturing discriminative information while avoiding early hard decisions.

We therefore revisit pose estimation as holistic recognition using modern classifier and feature technology. Fig. 2 presents an overview of our approach. During training, we partition the space of keypoints and train models for arm configurations, or armlets, using linear SVMs. Given a test image, we apply the trained models and use the mean predictions of the highest scoring activation to estimate the location of the joints. To train the armlets we extract features that could capture the necessary cues for accurate arm keypoint predictions. In Fig. 3 we show our choice of features. To capture the strong gradients in the image, we construct HOG features from local gradient contours [5] and gPb contours [2]. Another significant cue is the position, scale and orientation of the hands and of the rigid body parts, such as head, torso and shoulders. We cannot assume that we have perfect body part predictions. However, poselets [4] provide a soft way of capturing the contextual information of

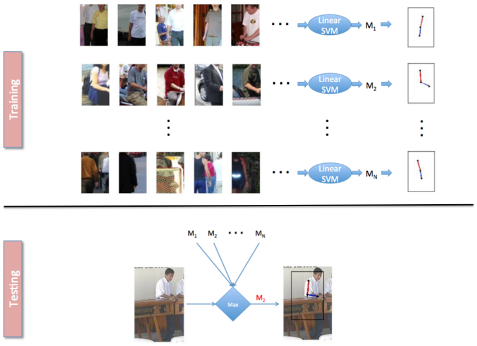

Figure 2. Our approach during training and testing. **Top**. We partition the keypoint space into arm configurations. For each configuration, we extract features and train linear SVMs using negative examples members of other configurations. The output of the training procedure, which we call armlets, consists of the SVM weights and predictions of the mean relative locations for all the keypoints. **Bottom**. Given the trained armlets and an image, we consider the highest scoring armlet activation assigned to each instance. From the predictions computed during training, we estimate the locations of the arm keypoints.

where the head or torso are in the image. Skin colored pixels are an additional cue to typically unclothed parts of the body, such as face and hands. The combination of all those four features gives an accuracy of 48.9% for upper arm and 23.6% for lower arm, according to the PCP metric [11], on a new dataset collected from the PASCAL VOC dataset [8] and the H3D dataset [4]. More details about the dataset can be found in Section 3. To further refine an armlet's joint location prediction we train shoulder, elbow and wrist detectors that are able to localize the joints more concretely, conditioned on an armlet activation. This improves performance, resulting in 49.7% PCP accuracy for upper arm and 25.2% for lower arm. Our approach is almost trivially simple compared to the complexity that one finds in the most recent elaborations of the stick figure fitting/pictorial structure paradigm. Yet our results are state-of-the-art, significantly outperforming Yang and Ramanan (Y&R) [26], the leading technique, who achieve 37.9% for upper arm and 20.1% for lower arm. We also tested the same PASCAL trained model "out-of-the-box" on the LSP dataset of sports figures [13, 14], where it performs creditably but is not the best, which is not suprising because of the dataset bias [24].

2. Related Work

The direction of representing the human body pose using stick figures was initially explored by Nevatia and Binford [18], where the body parts were modelled using generalized cylinders. Fischler and Elschlager [12] were the first ones to introduce pictorial structures for vision tasks while Felzenszwalb and Huttenlocher [9] presented a probabilistic framework for the same problem, which they called Pictorial Structure Model (PSM). In its original formulation, body parts (limbs) were represented as rectangles and their spatial relations were captured by tree-structured graphs. The tree-structure of PSM makes inference on location, scale and orientation of the body parts exact and efficient. However, the naive appearance model requires prior knowledge of the color of the parts, making PSM as formulated in [9] not applicable to real images. In subsequent work, more sophisticated appearance models have been explored. Ramanan [19] iteratively exploits image specific cues based on color and edge information. Andriluka *et al.* [1] build stronger generic part detectors based on shape context, while Eichner *et al.* [7] build stronger appearance models and exploit the similarity in appearance between parts.

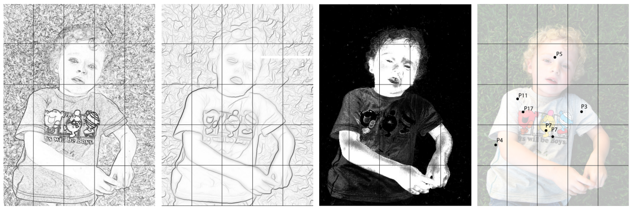

Figure 3. Our features. **Far left:** Each cell is HOG with local gradient contours. **Left:** Each cell is HOG with gPb. **Right:** We compute the average value of the skin classifier at each cell. **Far right:** Our context feature; at each cell we have a poselect activation feature vector. For each poselet type, we put the maximum of the scores of all poselet activations of that type whose center falls in the cell, or zero if no activations are present.

More recent work extends both the appearance models and the training framework. Ramanan and Sminchisescu [20] explore the benefits of using discriminative models, in particular Conditional Random Fields, compared to the generative framework used in PSM. Yang and Ramanan [26] augment PSM by defining mixtures of templates for each part, for capturing relations between them. Desai and Ramanan [6] enhance the model in [26] by training models for occluded parts. Johnson and Everingham [14] replace a single PSM with a mixture model of PSMs, for capturing pose-specific appearance terms corresponding to more informative pose priors. Sapp *et al.* [21] allow for richer appearance models, including contour and segmentation cues, by learning a cascade of pictorial structures of increasing pose resolution, which progressively filter the pose state space. Tiang *et al.* [23] explore a hierarchical model using mixture of parts and intermediate latent nodes to capture spatial relationships among them.

The approach presented in this work revisits the holistic recognition paradigm, initially explored by Mori and Malik [16, 17] and Sakhanarovich *et al.* [22], but using modern feature design and learning methods. Specifically, we build on poselets introduced by Bourdev *et al.* [3, 4]. Poselets are detectors corresponding to relatively large body parts (e.g. head & shoulders, torso) and capture contextual relationships between them, such as orientation, scale and aspect.

Wang *et al.* [25] use a hierarchy of poselets for human parsing. Their model consists of 20 different poselets and relates the parts in a graph with loops, thus making exact inference intractable. Our work is close to [25], since we also partition the keypoint configuration space to extract parts that are subsequently trained and used for recognition. However, we propose an augmented feature representation including richer information than standard HOG. In addition, our method replaces inexact inference on a graph with a simple but powerful feed forward network.

3. Datasets

Commonly used datasets for human pose estimation from 2D images, the Parse dataset [19], the Buffy dataset [11] and the PASCAL stickmen dataset [7], suffer from two significant problems: size and limitation of annotations. The Parse dataset contains around 300 instances, Buffy around 1000 and PASCAL stickmen around 500. These are relatively few examples given the range of human pose variation. The other fundamental problem with these datasets is that the joints are annotated in the image coordinate system, meaning that a joint is labeled as ‘left’ if it is leftmost in the image and not if it is the left joint of the person in question. The algorithms evaluated on those datasets are not required to discriminate between frontfacing and backfacing instances, which is another challenge in the task of human pose estimation.

In view of the great popularity of the PASCAL VOC challenge [8], which has driven contemporary research on classification, detection and segmentation, we thought that the collection of people images in the PASCAL dataset would constitute a very representative set for training and testing pose estimation algorithms. Our training set consists of the PASCAL VOC 2011 main dataset for the person category (excluding val ’09), the PASCAL VOC action recognition dataset as well as the H3D dataset [4]. Our training set consists of 5208 images with 9593 instances. We use the validation set of VOC 2009 as our test set, which consists of 1446 images with 2996 instances. The keypoints are annotated in the object coordinate system, requiring discrimination between frontfacing and backfacing poses. There are on average 2 instances per image, with a maximum of 18 instances per image, and 22.4% of the instances in the test set are non frontfacing. The dataset is publicly available.

We regard our dataset as complementary to the other “big” dataset for human pose estimation, the Leeds Sports Pose dataset [13, 14]. The extended LSP dataset contains 10000 training images of people performing sports, such as parkour, gymnastics and athletics, with one annotated instance per image. The test set consists of 1000 images with the same properties.

The algorithm developed in this paper is oriented towards the PASCAL dataset. Here we have people in relatively stereotyped poses but with significant amount of occlusion from other objects, people etc. The LSP dataset on the other hand has people performing athletic activities and tackles strong variations in pose. The person is typically fully visible but the poses are very unusual. It is not clear that the same techniques will perform equally well on both of these different datasets which pose different challenges.

4. Training armlets

In this section, we describe the procedure for selecting and training highly discriminative poselets to differentiate arm configurations. We also explain the choices of features. Fig. 2 shows our approach during training and testing.

4.1. Partioning of the Configuration Space

We create lists of positive examples by partitioning the arm configuration space. The space consists of the keypoint configuration of one arm, as well as the position of the opposite shoulder. This configuration space captures both the arm configuration as well as the 3D orientation of the torso. For example, an arm stretched downwards can be described by the location of the arm keypoints and the relative location of the opposite shoulder captures whether the person is front or back facing.

By defining a distance function $d(p, q)$ for p, q in the configuration space, we can quantitatively measure the similarity of two arm configurations. We define our distance function to be the euclidean distance of the centered and normalized x, y-positions of the keypoints for the two configurations, i.e. if $p = \{(x_i^p, y_i^p), i = 1, ..., K\}$ and $q = \{(x_i^q, y_i^q), i = 1, ..., K\}$, where K is the number of keypoints that define the configuration space, then

$$d(p,q) = \sqrt{\sum_{i=1}^{K} (\hat{x}_i^p - \hat{x}_i^q)^2 + (\hat{y}_i^p - \hat{y}_i^q)^2} \quad (1)$$

$$(\hat{x}_i^p, \hat{y}_i^p) = \big((x_i^p, y_i^p) - (\bar{x}^p, \bar{y}^p)\big)/\sigma_p \quad (2)$$

$$\sigma_p = \sqrt{\frac{1}{K}\sum_{i=1}^{K} (x_i^p - \bar{x}^p)^2 + (y_i^p - \bar{y}^p)^2} \quad (3)$$

where $(\bar{x}^p, \bar{y}^p)$ is the point around which we center the keypoints in p. Eq. 2 and Eq. 3 hold similarly for q.

We partition the configuration space in a greedy fashion. We iteratively pick a configuration p from the training set. If it falls within ϵ distance from the center of a configuration component, i.e. $d(p, center_i) < \epsilon$ for some i, then p is assigned to the i-th component. If no such i exists, then p forms the center of a new component. This is repeated until all the training instances have been picked. After the partitions have been formed, we reinitialize the center of each

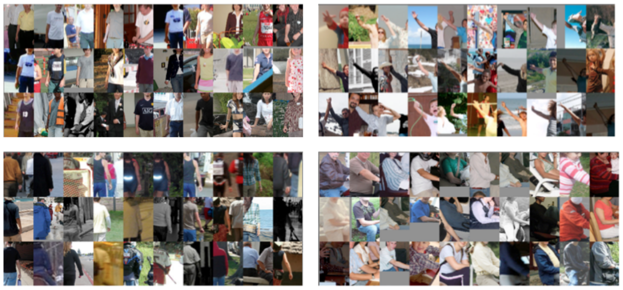

Figure 4. Examples of four different arm configurations resulting from the partitioning of the right arm configuration space.

component by the configuration that minimizes the distance to all other members in that component.

We also considered clustering techniques, such as agglomerative clustering. The results of those methods were very similar to our partioning, as expected since the configuration space is rather continuous. In addition, our technique is $O(N^2)$ which is faster than agglomerative clustering, which scales as $O(N^2 logN)$, where N is the number of training examples.

We collect patches of arm configurations by partioning the configuration space as described above. We center the patch of a configuration p around the location of the elbow and scale it by $2\sigma_p$, where σ_p is defined in Eq. 3. We sort the examples in each armlet according to the distance function in Eq. 1 having as reference the center of the configuration cluster, which we call the seed patch.

We obtain 25 different arm configurations for each arm after partitioning the corresponding configuration space. Fig. 4 shows four right arm configurations out of the 25.

4.2. Features

In this subsection, we explore the various choices of features, as illustrated in Fig. 3.

HOG with local gradient contours. Histograms of oriented gradients after convolving the image with tap filter [5] captures high frequency information. The HOG of local gradient contours captures the orientation of gradients while allowing for small deformations. We choose 16×16 pixel blocks of four 8×8 pixel cells, and 9 bins for orientation. For a 96×64 pixel patch, the dimensionality of the HOG-tap feature is 2772.

HOG with gPb contours. Histograms of oriented gradients of the gPb output of an image [2] will capture only the contours that emerge from strong brightness, color and texture gradients. We choose 16×16 pixel blocks of four 8×8 pixel cells, and 8 bins for orientation. For a 96×64 pixel patch, the dimensionality of the HOG-gPb feature is 2464.

Skin color. Information about the location of skin in the image is an important cue for arm-specific detectors. Strong responses of a skin detector indicate where the head and the hands of a person in an image are likely to be located and thus eliminate the large number of possible arm configurations. Our skin detector is a Gaussian Mixture Model with five components which operates in LAB space. We generated our training data from skin patches using the H3D dataset [4]. We bin the skin information using 8×8 pixel cells, where each cell contains the average probability of existence of skin inside the cell. For a 96×64 pixel patch, the dimensionality of the skin color feature is 308.

Context. The location of the head, the torso, their orientation and scale is significant in detecting an arm configuration. For example, it is much easier to detect where the right arm is if we know where the head is, where the torso is and whether the person is facing front or back. To encode that information, we use generic poselets [4], trained for the purpose of person detection. We will call them detection-specific poselets. For our purpose, we use N detection-specific poselets (N=30). For each 8×8 pixel cell, we define a N-dimensional activation vector that contains in its i-th entry the score of the i-th detection specific poselet, if the center of the activation is located within radius r (r=8) from the center of the cell, and 0 otherwise. For a 96×64 pixel patch, the dimensionality of the context feature is 2310.

Fig. 3 shows an example of our features. We show the local gradient and the gPb contours used to construct the HOG features, the output of the skin detector and detection-specific poselet activations used to encode context.

4.3. Classifier Training

The top panel of Fig. 2 describes the pipeline of our training procedure.

We construct the feature vector for every patch and train linear SVM classifiers. Since we want our armlets to be discriminative and fine-grained, we use negative images coming from people but with different arm configurations. An instance with keypoint configuration q is considered as a negative example for an armlet α with a seed patch of configuration $center_\alpha$, if $d(center_\alpha, q) > 2 \cdot \max_{i\in\{1,\dots,N_\alpha\}} d(center_\alpha, p_i)$ where p_i is the i-th member of armlet α consisting of N_α members.

For each armlet α, we model the distribution of the location of each joint J by fitting a gaussian. The distribution of the location $\mathbf{x}$ for joint J conditioned on an activation α_i of armlet α is given by

$$P_m(\mathbf{x}|\ \alpha_i) = \mathcal{N}\big(\mathbf{x}\,|\,\mu_J^{(\alpha)},\,\Sigma_J^{(\alpha)}\big) \qquad (4)$$

where $\mu_J^{(\alpha)}$ is the mean location of J and $\Sigma_J^{(\alpha)}$ is the covariance matrix, conditioned on activation α_i. Both parameteres are ML estimates from the positive examples of α.

4.4. Keypoint prediction at test time

To get the armlet activations for an input image, we apply the trained model at multiple scales and keep the activations with non negative scores. For the task of keypoint prediction, we cluster the activations to the instances in the image. We assume that the torso bounds of all instances in the image are known. We associate an activation to the instance with the biggest overlap with the predicted torso bounds and if that is greater than 0.4. Subsequently, for every instance in the image we consider the activation with the highest score assigned to that instance and use its mean prediction for the location of the arm keypoints. In other words, if β_i^* is the activation with the highest score, which is of armlet type β, then the final prediction for joint J is given by $\mu_J^{(\beta)}$.

5. Results using armlets

In this section we report the performance of armlets and compare it with Yang and Ramanan [26]. For performance evaluation, we use the PCP metric [11], which is the most commonly used metric in the literature for reporting results on the pose estimation task. According to this metric, a part of the stick figure is predicted correctly if the predicted locations of its endpoints are within 0.5 of the part length from the ground truth locations of the corresponding endpoints.

Feature evaluation. We present an ablation study to compare the performance of our method with respect to the different features used to train armlets. In Table 1 we present our results according to the PCP metric on the PASCAL VOC dataset for different combination of features. For computational efficiency, we perform non max suppression on the activations.

The performance of our complete system (LSCG) is 47.8% for the Upper Arms and 23.0% for the Lower Arms, compared to 44.5% and 19.8% respectively for standard HOG (L). It is clear that our approach of using gPb contours, skin and detection-specific poselets for context leads to a significant improvement over the standard HOG.

One can get additional insight from the ablation study, where we removed each of the cues in turn from the full system, as shown in Table 1.

Due to the combination of all the features, we get sparser activations and thus we can remove NMS. The performance of our algorithm with and without NMS is shown in Table 2.

Comparison with baseline. We compare our method with the state of the art method by Y&R [26]. To ensure a fair comparison, we gave Yang and Ramanan our data and asked them to train on our training set and evaluate on our test set. See Table 2. Note that even with the vanilla HOG detector (column L in Table 1) we achieve 32.1% PCP accuracy which outperforms Y&R, who achieve 29.0%. If we give Y&R the benefit of using image coordinates, their performance goes up slightly (41.1% for Upper Arm and 21.5% for Lower Arm) but is still below us.

PCP	L	LCG	LSC	LSG	SCG	LSCG
R_UpperArm	44.8	45.9	46.5	46.7	47.1	**48.1**
R_LowerArm	20.0	21.1	22.3	22.5	19.7	**23.2**
L_UpperArm	44.1	46.0	46.9	**47.7**	44.2	47.5
L_LowerArm	19.5	21.0	21.1	**23.6**	18.7	22.7
Average	32.1	33.5	34.2	35.1	32.4	**35.4**

Table 1. Part localization accuracy on the PASCAL VOC dataset. L is the standard HOG feature based on local gradient contours, G is the HOG feature based on the gPb countours, C is the context feature and S is the skin color feature (with NMS on the activations).

PCP	LSCG_NMS	LSCG_noNMS	Y&R
R_UpperArm	48.1	**49.4**	38.9
R_LowerArm	23.2	**23.5**	21.0
L_UpperArm	47.5	**48.3**	36.9
L_LowerArm	22.7	**23.7**	19.1
Average	35.4	**36.2**	29.0

Table 2. Part localization accuracy on the PASCAL VOC dataset with (first column) and without NMS (second column) using all the features (LSCG). The perfomance of Y&R [26] on our test set (third column).

We also evaluated our out-of-the-box model trained on PASCAL on the LSP test set, obtaining PCP accuracy of 35.6% and 19.2% for Upper and Lower Arms respectively. These numbers are below the state-of-the-art on LSP [14] 53.7% for Upper Arm and 37.5% for Lower Arm. This is not surprising in view of the dataset bias [24]; people in LSP are in unusual athletic poses compared to those in the PASCAL dataset.

6. Augmented Armlets

The armlets described above are trained to discriminate among different arm configurations. To capture the appearance of smaller areas around the joints we train three different poselets to detect the shoulder, the elbow and the wrist, specific to each of the 50 arm configurations. Below we explain how these models are trained and are used to make keypoint predictions.

6.1. Training augmented armlets

Each armlet can be considered as a root filter and the shoulderlet, the elbowlet and the wristlet are connected to

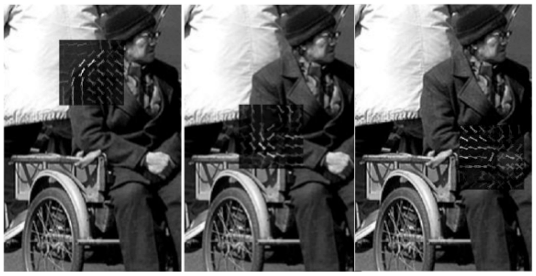

Figure 5. HOG templates for the shoulderlet, the elbowlet and the wristlet for armlet 3 superimposed on a positive example of that armlet.

the root forming a star model (tree of depth one), similar to [10]. However, the position of the corresponding joints w.r.t. the armlet activation is observed, in contrast to [10] where the location of the parts is treated as a latent variable.

We extract rectangular patches from the positive examples of each armlet type. The patches are centered at the keypoint of interest and at double the scale of the original positive example. Since the patches come from similar arm configurations, defined by the armlet type, they are aligned, allowing for the use of rigid features such as HOG.

Each patch is described by a local gradient HOG descriptor as well as the skin color signal. The patches are 64×64 pixels and we use 16×16 pixel blocks of 8×8 pixel cells for both features which are constructed as in Section 4.2. We use instance specific skin color models, which are GMMs with 5 components fitted on the LAB pixel values corresponding to the predicted face region of each instance as dictated by the detection specific poselet activations.

Subsequently, we learn a linear SVM using as negatives 64×64 sized patches coming from the positive armlet examples but not centered close to the joint in reference. After the first round of training, we re-estimate the positive patches by running the detector in a small neighborhood around the original keypoint location. This allows for some small variations in the alignment of the examples coming from the armlet clustering and results in a better alignment of the actual parts to be trained. A new model is trained with the improved positive examples. Fig. 5 shows an example of an armlet along with the HOG templates for the shoulderlet, the elbowlet and the wristlet.

6.2. Augmented armlet activations

We can use the activations of the shoulderlet, the elbowlet and the wristlet to rescore the original armlet activations. Strong part activations might indicate that the right armlet has indeed fired while weak part activations indicate a false positive activation.

Recall that for each armlet α, we computed the mean relative location and the standard deviation of the three arm keypoints from the positive examples of that armlet. Given an activation of that particular armlet, these locations give a rough estimate of where the joints might be located within the bounding box of the activation. We define an area of interest for each keypoint centered at the mean location and extending twice the empirical standard deviation. For each part, we detect its activations within the area of interest and record the highest scoring activation. In other words, each armlet activation α_i is now described by its original detection score s_{α_i} as well as the three maximum part activation scores $s_{\alpha_i}^{(J)}$, $J = 1, 2, 3$ corresponding to the three arm keypoints. Let us call $\mathbf{v}_{\alpha_i}$ the part activation vector which contains those four scores of the armlet activation α_i.

For each armlet α, we can train a linear SVM $\mathbf{w}_\alpha$ with positives $P_\alpha = \{\mathbf{v}_{\alpha_i} \mid i \in TP(\alpha)\}$ where $TP(\alpha)$ is the set of true positive activations of armlet α and negatives $N_\alpha = \{\mathbf{v}_{\alpha_i} \mid i \in FP(\alpha)\}$ where $FP(\alpha)$ is the set of false positive activations of armlet α. An armlet activation α_i has subsequently an activation score $\sigma\big(\mathbf{w}_\alpha^T \mathbf{v}_{\alpha_i}\big)$, where $\sigma(\cdot)$ is a logistic function trained for each armlet α.

6.3. Using the augmented armlets for keypoint predictions

The activations of the shoulderlets, the elbowlets and the wristlets can also be used for improved predictions of the location of the corresponding joints.

Assume α is an armlet and α_i is the i-th activation of that armlet in an image I. The prior probability that joint J is located at $\mathbf{x}$ is given by Eq. 4.

The score of a part activation at location $\mathbf{x}$ of the trained model for joint J, after fitting a logistic on the SVM scores, can be interpreted as the confidence of the part model at that location. In other words, if L_J is a binary random variable. indicating whether joint J is present, then the probability of joint J being present at location $\mathbf{x}$

$$P\big(L_J \mid I, \alpha_i, \mathbf{x}\big) = \frac{1}{1 + exp(-\gamma_\alpha^J s_{\alpha_i}^{(J)} - \delta_\alpha^J)} \tag{5}$$

where $\{\gamma_\alpha^J, \delta_\alpha^J\}$ are the trained parameters of the logistic and $s_{\alpha_i}^{(J)}$ the SVM score of the part model for joint J at $\mathbf{x}$.

The predicted location of part J conditioned on the activation α_i is given by

$$\mathbf{x}^{*(J)} = \arg\max_{\mathbf{x}} P_m(\mathbf{x}| \ \alpha_i) \cdot P\big(L_J| \ I, \alpha_i, \mathbf{x}\big) \tag{6}$$

7. Results using augmented armlets

We can use the shoulderlets, elbowlet and wristlet trained for each armlet to rescore the activations on the test set as well as make keypoint predictions, as described in Section 6. Table 3 shows the performance on the test set. The first column shows the performance after picking the highest scoring armlet activation, as described in Section 5. The

PCP	LSCG	LSCG_augmented	LSCG_posterior
R_UpperArm	49.4	50.2	**50.2**
R_LowerArm	23.5	23.4	**25.0**
L_UpperArm	48.3	**49.3**	49.2
L_LowerArm	23.7	24.5	**25.4**
Average	36.2	36.9	**37.5**

Table 3. Part localization accuracy on the PASCAL VOC dataset using all the features (LSCG) using the mean relative location for joint predictions of the highest scoring armlet activations (first column), using the mean relative location for joint predictions of the highest scoring augmented armlet activations (second column) and using the posterior probability for joint predictions in Eq. 6 (third column)

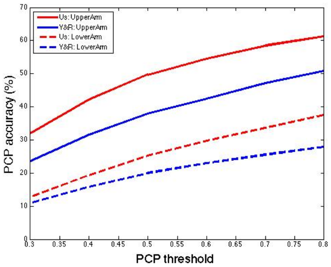

Figure 6. PCP localization accuracy for different values of the threshold in the metric. The red curves show our performance for the Upper and Lower Arm (mean across right and left). The blue curves show the performance of [26]. (**best viewed in color**)

second column shows the performance after picking the maximum scoring activation of the augmented armlets to make predictions for the joints using the mean relative locations. The third column shows the performance after using the highest scoring activation of the augmented armlets to make a prediction using the posterior probability (Eq 6). [1]

Fig. 6 shows the PCP localization accuracy of our final model compared to Yang and Ramanan [26] for different thresholds for the PCP metric evaluation.

Fig. 7, 8 show examples of correct right and left, respectively, arm keypoint predictions. The bounds of the instance in question are shown in green. The red stick corresponds to the upper arm and the blue to the lower arm of that instance.

Fig. 9 shows incorrect keypoint predictions for the right and left arm corresponding to the instance highlighted in green.

8. Discussion

We propose a straightforward yet effective framework for training arm specific poselets for the task of joint position estimation and we show experimentally that it gives superior results on a challenging dataset.

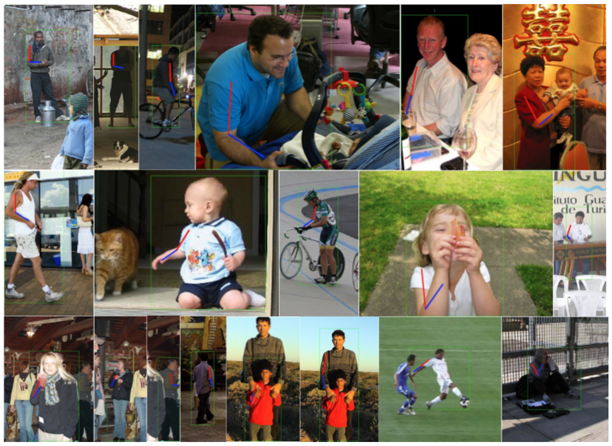

Figure 7. Examples of correct right arm keypoint predictions. Red corresponds to the upper arm and blue to the lower arm of the person highlighted in green. (**best viewed in color**)

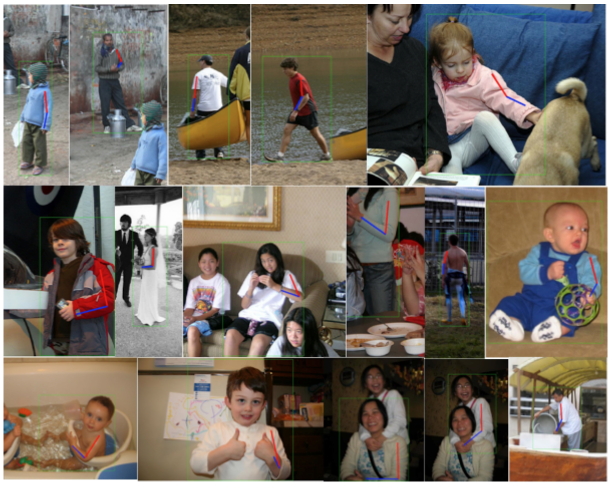

Figure 8. Examples of correct left arm keypoint predictions. Red corresponds to the upper arm and blue to the lower arm of the person highlighted in green. (**best viewed in color**)

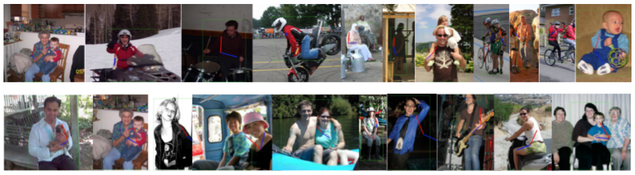

Figure 9. Examples of incorrect keypoint predictions for the right arm (**top**) and left arm (**bottom**) . Red corresponds to the upper arm and blue to the lower arm of the person highlighted in green. (**best viewed in color**)

The shortage of data for developing efficient human pose estimation algorithms has a significant impact on the performance for both our and Yang and Ramanan's approach. Our knowledge of the ground truth configuration on the test set enables us to cluster each test instance to one of the armlets. Thus, we can compute the PCP accuracy per armlet type

[1] We also cross-checked by using bounding boxes rather than torsos to determine the ground truth person, and the numbers change only slightly to 47.5% for Upper and 23.8% for Lower Arms

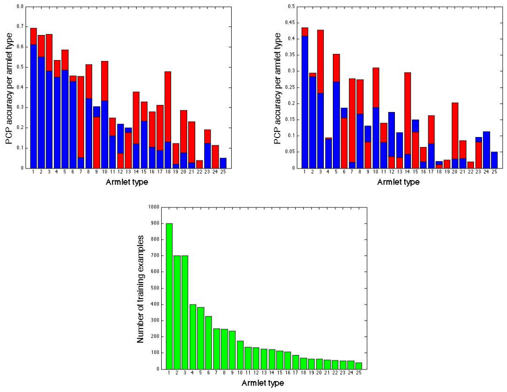

Figure 10. PCP localization accuracy per armlet type for upper arm (**top left**) and for lower arm (**top right**), where red indicates the performance of our approach while blue the performance by Yang and Ramanan [26]. The number of training examples per armlet type is shown in the **bottom**.

and associate it with the number of training examples for that armlet. Fig. 10 shows PCP accuracy per armlet type on the test set for the upper arm (top left) and for the lower arm (top right), as well as the number of training examples per armlet type (bottom). These plots show that our approach dominates Y&R's for most armlet types on the test set, and also reveal that both methods are strongly correlated with the amount of training data. In particular, the Pearson's correlation coefficient between the number of training examples and the PCP accuracy for the upper arm is 0.79 for our approach and 0.88 for Y&R while for the lower arm it is 0.75 for our approach and 0.83 for Y&R. Clearly more training data will be needed to achieve higher pose estimation accuracies. We give the last word to Sherlock Holmes:

> "Data! Data! Data!" he cried impatiently. "I can't make bricks without clay."
> *The Adventure of the Copper Beeches*

Acknowledgements

This research was supported by the Intel Visual Computing Center and by ONR SMARTS MURI N00014-09-1-1051. We also thank R1 for thoughtful comments as well as Yi Yang and Deva Ramanan for their help.

References

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

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 2011.

[3] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. *ECCV*, 2010.

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

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

[6] D. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. *ECCV*, 2012.

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

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html, 2011.

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

[10] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 2010.

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

[12] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Comput.*, 1973.

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

[14] S. Johnson and M. Everingham. Learning effective human pose estimation from inaccurate annotation. *CVPR*, 2011.

[15] T. Malisiewicz, A. Gupta, and A. Efros. Ensemble of exemplar-svms for object detection and beyond. *ICCV*, 2011.

[16] G. Mori and J. Malik. Estimating human body configurations using shape context matching. *ECCV*, 2002.

[17] G. Mori and J. Malik. Recovering 3d human body configurations using shape contexts. *PAMI*, 2006.

[18] R. Nevatia and T. Binford. Description and recognition of curved objects. *Artif. Intell.*, 1977.

[19] D. Ramanan. Learning to parse images of articulated bodies. *NIPS*, 2006.

[20] D. Ramanan and C. Sminchisescu. Training deformable models for localization. *CVPR*, 2006.

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

[22] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. *ICCV*, 2003.

[23] Y. Tian, L. C. Zitnick, and S. G. Narasimhan. Exploring the spatial hierarchy of mixture models for human pose estimation. *ECCV*, 2012.

[24] A. Torralba and A. A. Efros. Unbiased look at dataset bias. *CVPR*, 2011.

[25] Y. Wang, D. Tran, and Z. Liao. Learning hierarchical poselets for human parsing. *CVPR*, 2011.

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