IEICE TRANS. INF. & SYST., VOL.E99–D, NO.1 JANUARY 2016

PAPER

# Part-Segment Features with Optimized Shape Priors for Articulated Pose Estimation

**Norimichi UKITA**[†a)], ***Senior Member***

**SUMMARY** We propose part-segment (PS) features for estimating an articulated pose in still images. The PS feature evaluates the image likelihood of each body part (e.g. head, torso, and arms) robustly to background clutter and nuisance textures on the body. While general gradient features (e.g. HOG) might include many nuisance responses, the PS feature represents only the region of the body part by iterative segmentation while updating the shape prior of each part. In contrast to similar segmentation features, part segmentation is improved by part-specific shape priors that are optimized by training images with fully-automatically obtained seeds. The shape priors are modeled efficiently based on clustering for fast extraction of PS features. The PS feature is fused complementarily with gradient features using discriminative training and adaptive weighting for robust and accurate evaluation of part similarity. Comparative experiments with public datasets demonstrate improvement in pose estimation by the PS features.

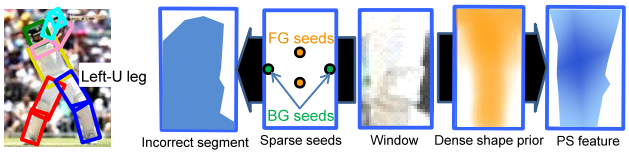
***key words:*** *human pose, part segmentation, pictorial structure models*

## 1. Introduction

Deformable part models [1] and feature classifiers (e.g. discriminative classifier [2]) have improved articulated pose estimation. As well as models and classifiers, image features are crucial. While most articulated pose estimation methods employ gradient features such as HOG [3], they include not only useful responses along a body boundary but also nuisance responses caused by background clutter and textures on a target body.

This work focuses on how to extract only the boundary of each part based on a shape prior. The boundary is represented by a *part-segment* (PS) feature (Fig. 1 (b)). For the PS feature, initial segments are obtained by binary seeds. Their distribution is automatically determined by training images depending on the part. While the seed distribution gives a weak shape prior, a more reliable dense shape prior is acquired from initial segments in training images.

Our contribution is threefold: 1) initial *shape priors* are extracted by segmentation using *part-specific* foreground (FG) and background (BG) seeds trained *automatically* (Sect. 4.1), 2) the shape priors are refined and clustered for correctly and efficiently computing PS features (Sect. 4.2), and 3) *adaptive weighting* of the PS features with domain adaptation improves their discriminativity (Sect. 4.3). Compared with our earlier work [4], efficiency in feature computation is improved and the proposed feature is evaluated with more data in detail.




(a) Human pose (b) Segments extracted by seeds (left) and ours (right)

**Fig. 1** (a) All parts, shown by rectangle windows, are configured in their proper locations. (b) Segmentation-based feature of a part, the left-upper leg in this example shown in "Window". Pixels that are more probable to be included in a part of interest are depicted by darker colors in "Dense shape prior" and "PS feature". While the proposed PS feature is extracted by a part-specific dense shape prior, correct segmentation is difficult with a weak shape prior (i.e. "Sparse seeds").

## 2. Related Work

To suppress nuisance responses, multimodal cues including segments are useful (e.g. color distributions, superpixels, and their scales in [13], edges and color-segmented regions in [14], and smooth connectivity between parts [15]). Segments can be obtained by image segmentation, such as normalized cuts [16], globalPb [17], and superpixelization [18]. However, it is not easy to extract each part as one segment because the part might be over-segmented due to textures and shades on a human body. Such over-segmentation can be suppressed by the shape prior of each part. Table 1 summarizes several properties of methods for/using segmentation.

As the prior, the configuration of roughly detected parts is useful (e.g. upper-body [9] and whole-body [19], [20] detection). In ObjCut [5] and [6], [8], [9], one or more parts are detected initially using features with no segmentation. Depending on the configuration of the detected parts, seeds for segmenting all parts in a human body are distributed. It is, however, difficult to distribute the seeds sufficiently to all the parts by using only the limited detected parts.

While several methods [11], [12] achieve part segmentation and detection independently ("DI" in Table 1), each of these methods has functional defects. Segmentation cues [12] require manually-predefined sparse seeds, while our method acquires the probabilistic pixelwise shape prior; this difference is shown in "SP" and "PS-SP" in Table 1. CHOG [11] also needs a set of manually-given seeds so that those suitable for segmentation are selected by using training images. In addition, CHOG provides only sparse seeds

Manuscript received June 12, 2015.
Manuscript revised September 4, 2015.
Manuscript publicized October 14, 2015.
[†]The author is with Nara Institute of Science and Technology, Ikoma-shi, 630–0192 Japan.
a) E-mail: ukita@is.naist.jp
DOI: 10.1587/transinf.2015EDP7228

**Table 1** Comparison of body/part segmentation methods. Each column shows whether or not the methods exhibit each property. No init: no pose initialization is required. DT: feature is discriminatively trained. Weight: each pixel/cell in a segment extracted from a test image has its weight (i.e. probability to be FG). DI: detection and segmentation are achieved independently. SP: segmentation is achieved using a shape prior. PS-SP: part-specific shape prior is given. AS: seeds are given fully automatically. NR: the method has a mechanism for coping with noise in segmentation.

| | No init | DT | Weight | DI | SP | PS-SP | AS | NR |
|---|---|---|---|---|---|---|---|---|
| (a) ObjCut [5] | | | | | Y | | | |
| (b) Parse [6], Better appearance [7] | | | Y | | | | | |
| (c) Segmentation [8] | | Y | | | Y | | | |
| (d) Progressive search [9] | | | | | | | | |
| (e) PoseCut [10] | Y | | | | Y | | | |
| (f) CHOG [11] | Y | Y | Y | Y | | | | Y |
| (g) Segmentation cues [12] | Y | Y | | Y | | | | |
| (h) PS feature (Proposed) | Y | Y | Y | Y | Y | Y | Y | Y |

(i.e. a pair of FG and BG pixels). The proposed PS feature is extracted by part-specific dense shape priors optimized by automatically-given seeds and training images; these advantages are shown in "AS" and "PS-SP" in Table 1.

Noise in segmentation is a critical issue. To suppress this problem, the PS feature enhances its robustness by domain adaptation, while binning and histogramming are applied to a segmentation image in CHOG [11] as with HOG.

## 3. Pose Estimation Using Pictorial Structure Models

An articulated model is represented by a tree model with a set of nodes, $\boldsymbol{V}$, and a set of links each of which connects two nodes, $\boldsymbol{E}$, as presented in [1]. Each node and link respectively corresponds to a part and a connection between parts. The pose parameters of the node are optimized for pose estimation by maximizing the score function below:

$$T(\boldsymbol{P}) = \sum_{i \in \boldsymbol{V}} S_i(\boldsymbol{p}_i) + \sum_{i,j \in \boldsymbol{E}} P_{i,j}(\boldsymbol{p}_i, \boldsymbol{p}_j), \tag{1}$$

where $\boldsymbol{p}_i$ and $\boldsymbol{P}$ denote the pose parameters of $i$-th part and its set of all parts ($\boldsymbol{P} = \{\boldsymbol{p}_i | \forall i \in \boldsymbol{V}\}$).

$S_i(\boldsymbol{p}_i)$ is a similarity score of $i$-th part at $\boldsymbol{p}_i$. In this paper, $S_i(\boldsymbol{p}_i)$ is a sum of filter responses using HOG [3] and the PS feature, which are extracted from each window:

$$S_i(\boldsymbol{p}_i) = \left[F_i^T, G_i^T\right] \left[\phi(I, \boldsymbol{p}_i), \varphi(I, \boldsymbol{p}_i, i)\right]^T \tag{2}$$

where $F_i$ and $\phi(I, \boldsymbol{p}_i)$ denote the filter of $i$-th part and the HOG extracted at $\boldsymbol{p}_i$ in image $I$, respectively, and $G_i$ and $\varphi(I, \boldsymbol{p}_i, i)$ denote those of the PS feature.

$P_{i,j}(\boldsymbol{p}_i, \boldsymbol{p}_j)$ is a spring-based score between $i$-th and $j$-th parts, which has a greater value if the configuration of $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ is highly probable. In our model, $P_{i,j}(\boldsymbol{p}_i, \boldsymbol{p}_j)$ is expressed by the following form [21]:

$$P_{i,j}(\boldsymbol{p}_i, \boldsymbol{p}_j) = \boldsymbol{w}_{i,j}^T \cdot [d_{i,j}^x, d_{i,j}^{x^2}, d_{i,j}^y, d_{i,j}^{y^2}]^T \tag{3}$$

$\boldsymbol{w}_{i,j}$ is a weight parameter. $d_{i,j}^x$ and $d_{i,j}^y$ denote $x_i - x_j$ and $y_i - y_j$, respectively, where $(x_i, y_i) \in \boldsymbol{p}_i$ and $(x_j, y_j) \in \boldsymbol{p}_j$ are the locations of $i$-th and $j$-th parts.

In what follows, how to learn $G_i$ and extract $\varphi(I, \boldsymbol{p}_i, i)$ is described.

## 4. Training of Part-Segment Features

### 4.1 Initial Shape Prior Obtained by Part Segmentation Using FG and BG Seeds

The shape prior of each part is obtained from its segments in all training images. For extracting the segment in each image, the image is segmented by general image segmentation. For efficiency and accuracy, SLIC superpixelization [18] is used. Figures 2 (a), (b), and (c) show an image, its segmented image, and part windows whose locations are given in annotation data, respectively. Since the region of a part might be over-segmented, these segments must be clustered into those of the part of interest and others.

#### 4.1.1 Fully-Automatic Configuration of Seeds

Clustering over-segmented segments is achieved initially with seeds automatically given by using training data. Training data consists of images and pose annotations. The pose of each part is given as a pair of end-points of a part line (Fig. 3 (a)). In each part's window, the initial sample colors of FG are collected from segments that cross the part line. Then the distance between the color of each pixel in the window and its nearest neighbor color in the collected FG samples is computed. The color distances are binarized by [22] for dividing the pixel colors into FG (i.e. colors with a smaller distance) and BG, as shown in Fig. 3 (b). If a segment has 50% or more FG color pixels, it is clustered into FG. This clustering is executed in all parts' windows in all training images. After the window sizes are normalized, the rate of FG in all training images is computed in each pixel with respect to each part. Pixels with the top/bottom $\gamma$% FG rate are extracted as FG/BG seeds.

#### 4.1.2 Segment Clustering with Seeds

The seeds provide a weak shape prior. With the advantage of spatially-distributed binary seeds having a variety of color samples, segment clustering is re-executed as follows:

1. Segments each of which has *only* FG/BG seeds are

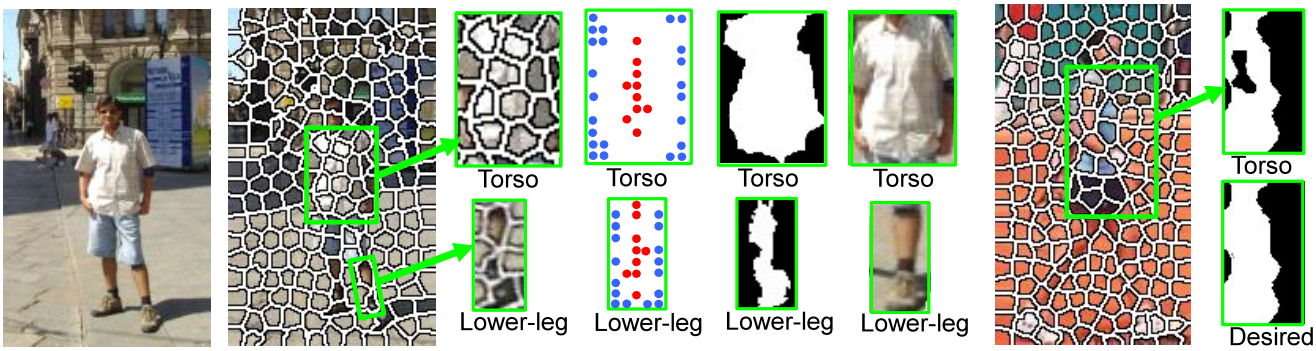
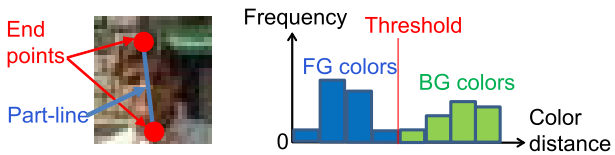



**Fig. 2** Initial part segmentation in training. (a) Input image. (b) Correct windows of two parts (i.e. torso and left lower-leg) superimposed on a segmented image. (c) Cropped windows of the torso and the left lower-leg. (d) FG and BG seeds, indicated by red and blue circles, respectively. (e) Binary segmentation using the seeds. (f) Parts' windows cropped from (a). (g) Another input image. (h-upper) Torso segment extracted from (g). (h-lower) Desired torso segment in (g).




(a) Cropped window (b) Color clustering

**Fig. 3** Learning the configuration of seeds.

clustered into FG/BG segments. If either of FG or BG segment is not found, this window is removed from the following processes.

2. Each of remaining segments is clustered into FG or BG based on its nearest neighbor colors with FG and BG segments.
3. The FG and BG segments are regarded as an initial PS feature (denoted by $\hat{\varphi}_{i,j}$ for $i$-th part in $j$-th training image) where FG/BG pixels have 1/0 as pixel values.

Binary PS features of all training images are averaged in each part. The mean is regarded as the initial shape prior of the part (denoted by $\bar{\varphi}_i$ for $i$-th part): $\bar{\varphi}_i = \left(\sum_j^{N_p} \hat{\varphi}_{i,j}\right)/N_p$, where $N_p$ denotes the number of training images.

By comparing obtained binary PS features (Fig. 2 (e), where white and black depict FG and BG pixels, respectively) with their respective images (Fig. 2 (f)), it seems segmentation is reasonable. However, a PS feature might be sometimes extracted unsuccessfully, as shown in Fig. 2 (h-upper); its successful example is (h-lower).

## 4.2 Shape Prior Refinement

### 4.2.1 Shape Prior Refinement with Updating PS Features

For refining the shape prior, a PS feature of $i$-th part in each training image is updated with $\bar{\varphi}_i$:

1. The mean color of a segment having FG/BG seeds in each window is stored as the sample color of FG/BG.
2. By comparing $\bar{\varphi}_i$ with segments in the window, the mean of pixel values of $\bar{\varphi}_i$ in $s$-th segment is regarded as the probability that the segment is FG. This probability is denoted by $P_f(s)$.
3. The nearest neighbor color of the mean color in the $s$-th segment is found from the sample colors of FG. The color distance from the nearest neighbor color is denoted by $l_f(s)$. $l_b(s)$ for BG is also computed.
4. By deeming $\exp(-l_f(s))$ and $\exp(-l_b(s))$ to be image likelihoods, the Bayes' theorem gives the following probabilities:

$$P(f|s) \propto \exp(-l_f(s))P_f(s)$$
$$P(b|s) \propto \exp(-l_b(s))(1 - P_f(s))$$

In an updated PS feature, pixels in $s$-th segment have the pixel value below: $P(f|s)/\left(P(f|s) + P(b|s)\right)$.

Finally, the mean of the updated PS features of all training images is regarded as a refined dense shape prior, $\bar{\varphi}_i$.

The process mentioned above can be repeated based on the EM algorithm, where observed data is the mean colors of segments, latent data is their clusters (i.e. foreground or background), and unknown parameters are $\bar{\varphi}_i$. The expectation and maximization steps are segment feature extraction in each image and shape prior acquisition by averaging, respectively. To avoid overfitting, the above mentioned process is repeated two times.

### 4.2.2 Clustering Dense Shape Priors

PS features are extracted for all parts because they are part-specific. Since this is time-consuming, shape priors are clustered based on their similarity for reducing PS features extracted in a window; if $p$-th and $q$-th parts share the shape prior, they also have the same PS feature. This is reasonable because, at least, left and right symmetric parts can share their shape priors. The mean of shape priors in each cluster is regarded as a shape prior of parts in that cluster.

(a-1) (a-2) (b-1) (b-2) (c-1) (c-2)

**Fig. 4** (a-*) and (b-*): Windows of the torso. (c-*): Window placed in the legs. (*-1): RGB images. (*-2): PS features obtained with the shape prior of the torso.

In our experiments, the computational cost of the proposed method in inference can be reduced by 42% by clustering of the shape priors.

### 4.3 Discriminative Training of Adaptively-Weighted Gradient and Segment Features

Discriminative training [2], [21] optimizes the model parameters in score (1), namely $F_i$ and $G_i$ in (2) and $\boldsymbol{w}_{i,j}$ in (3). In this training, optimization of the weights between gradient and PS features is imposed in $F_i$ and $G_i$. If robustness in feature extraction is comparable between gradient and PS features, this weighting is effective. To see the robustness, examples of extracted features are shown in Fig. 4. From windows (a-1), (b-1), and (c-1), PS features with respect to the torso were extracted. From these examples, we can see the following observations: (i) if windows are located properly with respect to a part of interest, PS features capture sometimes ideal responses as shown in (a-2), but bad responses like (b-2) are possibly obtained and (ii) a PS feature can accidentally have a response that matches different parts as shown in (c-2), where a PS feature obtained from the region of the legs is similar to the shape of the torso. These observations reveal unrobustness of PS features.

To suppress bad effects due to unrobust PS features, an additional weight is given to a PS feature depending on its confidence. The confidence value $C(\varphi(I, \boldsymbol{p}_i, i), i)$ of PS feature $\varphi(I, \boldsymbol{p}_i, i)$ is determined by the subtraction between $\varphi(I, \boldsymbol{p}_i, i)$ and the shape prior of $i$-th part, $\bar{\varphi}_i$:

$$C(\varphi(I, \boldsymbol{p}_i, i), i) = \exp(-\|\varphi(I, \boldsymbol{p}_i, i) - \bar{\varphi}_i\|)$$

The next issue is how to use the confidence values of the PS features in the discriminative training framework. An easy way might be that each part is clustered with respect to the confidence values and trained independently. This independent training have the following disadvantages:

**Poor training:** The amount of training data in each cluster is decreased.

**Inefficient training:** Even if PS features are differently confident, they have similar responses in some pixels. Independent training with no correlation among these responses is inefficient.

For efficient training while partially sharing responses between ideal and other PS features, we train their appearance filters with domain adaptation by redundantly-concatenated features [23]. The feature vector, $[\phi(I, \boldsymbol{p}_i), \varphi(I, \boldsymbol{p}_i, i)]^T$, is changed to either of the followings depending on the confidence value of the PS feature:

$$[\phi(I, \boldsymbol{p}_i), \varphi_1, \varphi_2, \varphi_3]^T = [\phi(I, \boldsymbol{p}_i), \varphi(I, \boldsymbol{p}_i, i), \varphi(I, \boldsymbol{p}_i, i), \mathbf{0}]^T, \quad \text{if } C(\varphi(I, \boldsymbol{p}_i, i), i) < C' \tag{4}$$

$$[\phi(I, \boldsymbol{p}_i), \varphi_1, \varphi_2, \varphi_3]^T = [\phi(I, \boldsymbol{p}_i), \varphi(I, \boldsymbol{p}_i, i), \mathbf{0}, \varphi(I, \boldsymbol{p}_i, i)]^T, \quad \text{if } C' \leq C(\varphi(I, \boldsymbol{p}_i, i), i) \tag{5}$$

$C(\varphi(I, \boldsymbol{p}_i, i), i)$ is clustered into two classes by threshold $C'$. Given the number of the classes[†], $C'$ was determined by K-means clustering of $C(\varphi(I, \boldsymbol{p}_i, i), i)$ of PS features obtained from all training images; $C'$ coincides with the middle point between the means of two neighboring clusters. With the above feature vectors (4) and (5), appearance score (2) is rewritten:

$$S_i(\boldsymbol{p}_i) = \left[F_i^T, G_{i,1}^T, G_{i,2}^T, G_{i,3}^T\right][\phi(I, \boldsymbol{p}_i), \varphi_1, \varphi_2, \varphi_3]^T \tag{6}$$

## 5. Inference with Part-Segment Features

In pose inference, PS features are extracted from all possible windows in a test image for optimizing score (1). The PS features are extracted by steps 1–4 described in Sect. 4.2. Using the PS features, the appearance score (6) with concatenated features defined by Eqs. (4) and (5) are computed for score (1).

In a tree-based model, the globally optimized pose parameters, $\hat{\boldsymbol{P}}$, having the max score of (1) can be acquired efficiently by dynamic programming. In addition, distance transform [1] is applicable to fast message passing in dynamic programming.

## 6. Experiments

### 6.1 Human Body Model

In our implementation, a human body is represented by a tree model with a mixture of non-oriented structures proposed by Yang and Ramanan [21], [25]. In this model, each part $i$ has its $x$ and $y$ location and scale parameter $s$ as its parameters. Instead of a numerical parameter $\theta$ for orientation, the samples of $i$-th part are clustered depending on the relative location of $i$ with respect to its parent part. The ID of each cluster is called a *type*. For robustness to in-plane rotation and foreshortening of body parts, this base model [25] divides physically-rigid parts (e.g. limbs) into smaller parts. As with the base model, 26 and 18 parts were used for the full-body and upper-body models in our implementation; 2 for the head, 8 for the torso, 8 for the arms, and 8 for the legs.

For efficient representation of shape priors, left and right symmetric parts share their shape prior, as described in Sect. 4.2.2. To this end, left-side parts were flipped horizontally. In total, 14 and 10 shape priors were used.

[†]The number of classes, which was determined empirically in our experiments, can be determined in accordance with the distribution of confidence values (e.g. using Dirichlet processes [24]).

### 6.2 Datasets

We tested the proposed PS features with BUFFY stickmen [9], IP [6], and LSP [26] datasets. Each dataset consists of images for training and evaluation. The numbers of images in the BUFFY stickmen [9], IP [6], and LSP [26] datasets are 748 (472 for training and 276 for evaluation), 305 (100 for training and 205 for evaluation), and 2000 (1000 for training and 1000 for evaluation). All images were selected so that a variety of people and clothing are observed. Negative samples for discriminative training were given from 1218 background images in the INRIA Person dataset [3].

In accordance with [25], results were evaluated as follows. From annotation data of IP [6] and LSP [26], 10 line segments localizing body parts were extracted as ground-truth, while 6 line segments from BUFFY [9]. The former consists of the full body, and the latter the upper body. On the end points of the line segments, parts of the articulated model are located, while other parts are placed on the mid point of the segments. In total, 26 and 18 parts were respectively used for the full body and the upper body.

### 6.3 Effects of Parameters

The PS feature has several parameters below:

- The number of superpixels for image segmentation (denoted by $\alpha$)
- The normalized window size of a shape prior (denoted by $\beta$)
- The rate of pixels of FG and BG seeds (denoted by $\gamma$)

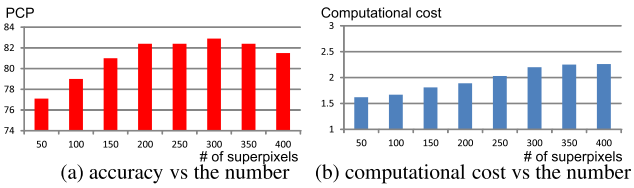
Their effects were evaluated with the IP dataset. In the following experiments showing the effect of each parameter, the other two parameters had their finally-selected values (i.e. $\alpha = 200, \beta = 11$, and $\gamma = 0.1$).

The results of pose estimation were measured quantitatively by the percentage of correctly localized parts (PCP). The PCP measure was implemented with the code in the BUFFY dataset [9] using the strictest interpretation described in [25]. In this measure, only a single pose detected with the maximum score is evaluated in each test image. Each body part in this detected pose is considered correct if both of its endpoints lie within 50% of the length of the ground-truth segment from their annotated location.

As well as PCP, the computational cost is evaluated. In what follows, the computational cost is represented by the ratio to the one of the base model [25], which is around between two and five seconds.

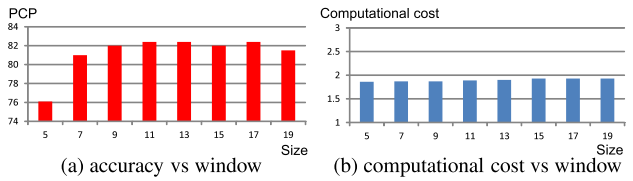
#### 6.3.1 Superpixels

Figure 5 shows the effects of the number of superpixels. From Fig. 5 (a), which shows the relationship between PCP and the number of superpixels, it can be seen that the number of superpixels has less impact on PCP and parameter selection is not sensitive, while a small peak is observed around 300 superpixels. This peak is observed because too few/many superpixels result in insufficient/over-segmentation. While PCP was saturated as the superpixels increases, the computational cost became higher as shown in Fig. 5 (b), where the mean of total computational costs for pose estimation is shown.



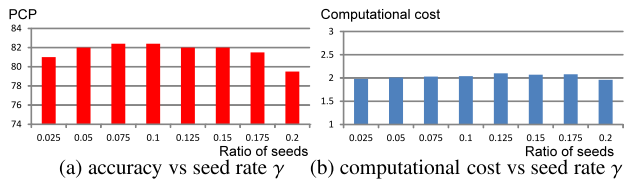

(a) accuracy vs the number (b) computational cost vs the number

**Fig. 5** Effects of the number of superpixels for computing PS features.




(a) accuracy vs window (b) computational cost vs window

**Fig. 6** Effects of the window size of a shape prior. The horizontal axis of each graph is the pixel length of each side of a squared window.




(a) accuracy vs seed rate $\gamma$ (b) computational cost vs seed rate $\gamma$

**Fig. 7** Effects of $\gamma$ for the configurations of FG and BG seeds.

Based on the observations mentioned above, the number of superpixels was determined to be 200 for trade-off between accuracy and computational cost. Note that a constant number of superpixels works fine in experiments because all images in the datasets were size-normalized so that the size of an observed person is normalized.

#### 6.3.2 Window Size of a Shape Prior

PCP was almost saturated in 11 (i.e. $11 \times 11$ window size) as shown in Fig. 6 (a), while a much smaller window cannot represent the shape of each part and a much larger window produces a noisy shape prior. Figure 6 (b) shows that the computational cost was not changed depending on the window size. This is apparent because window-size normalization is not computationally dominant in contrast to other steps (e.g. segment clustering in a window). From the results of PCP, the window size was determined to be $11 \times 11$.

#### 6.3.3 Rate of Seeds

The effects of $\gamma$ are shown in Fig. 7. With increasing $\gamma$, the seeds grow in number. As shown in Fig. 7 (a), $\gamma$ has very

**Table 2** BUFFY stickmen dataset: Comparative results of PCP. (a,b,c) related work, (d) our initial binary feature extracted only by seeds, (e) our feature without shape prior refinement, (f) our feature without domain adaptation, and (g) our proposed PS feature.

| Model | Head | Torso | Upper-arms | Lower-arms | Total |
|---|---|---|---|---|---|
| (a) Adaptive pose priors [27] | 100 | 100 | 91.1 | 65.7 | 85.9 |
| (b) Cascaded models [28] | 96.2 | 100 | 95.3 | 63.0 | 85.5 |
| (c) Mixture of parts [25] | 99.2 | 98.8 | 97.8 | 68.6 | 88.5 |
| (d) Ours by binary feature (by seeds) | 99.3 | 98.9 | 97.4 | 68.8 | 88.4 |
| (e) Ours without shape prior refinement (by non-refined shape prior) | 99.3 | 98.9 | 97.4 | 68.8 | 88.4 |
| (f) Ours without adaptation (by shape prior refinement) | 99.3 | 99.3 | 98.2 | 70.3 | 89.3 |
| (g) Ours (full model) | 99.3 | 99.3 | 98.2 | 70.3 | 89.3 |

**Table 3** IP dataset: Comparative results of PCP. (a) multiple pose models with training pose data clustering. (b) combining discriminative appearance and segmentation cues [12]. Methods (c), (d), (e), (f), and (g) are introduced in Table 2.

| Model | Head | Torso | U-legs | L-legs | U-arms | L-arms | Total |
|---|---|---|---|---|---|---|---|
| (a) Pose clustering [26] | 76.1 | 85.4 | 73.4 | 65.4 | 64.7 | 46.9 | 66.2 |
| (b) Segmentation cues [12] | 68.8 | 77.6 | 61.5 | 54.9 | 53.2 | 39.3 | 56.4 |
| (c) Mixture of parts [25] | 99.0 | 96.1 | 85.9 | 79.0 | 79.0 | 53.4 | 79.0 |
| (d) Ours by binary feature (by seeds) | 99.0 | 96.1 | 87.3 | 78.5 | 79.5 | 52.7 | 79.1 |
| (e) Ours without shape prior refinement (by non-refined shape prior) | 99.0 | 97.0 | 88.8 | 79.5 | 86.3 | 55.1 | 81.5 |
| (f) Ours without adaptation (with shape prior refinement) | 99.0 | 97.6 | 88.8 | 79.5 | 85.9 | 56.1 | 82.0 |
| (g) Ours (full model) | 99.0 | 97.6 | 89.8 | 80.5 | 87.3 | 56.1 | 82.4 |

**Table 4** LSP dataset: Comparative results of PCP. (a) pictorial structures with poselets [29]. (b) model with inaccurate annotations [30]. Methods (c), (d), (e), (f), and (g) are introduced in Table 2.

| Model | Head | Torso | U-legs | L-legs | U-arms | L-arms | Total |
|---|---|---|---|---|---|---|---|
| (a) Poselet PS [29] | 87.5 | 78.1 | 75.7 | 68.0 | 54.2 | 33.9 | 62.9 |
| (b) Learning from inaccurate annotation [30] | 88.1 | 74.6 | 74.5 | 66.5 | 53.7 | 37.5 | 62.7 |
| (c) Mixture of parts [25] | 84.1 | 77.1 | 69.5 | 65.6 | 52.5 | 35.9 | 60.8 |
| (d) Ours by binary feature (by seeds) | 86.5 | 78.4 | 73.7 | 64.9 | 55.2 | 39.9 | 63.2 |
| (e) Ours without shape prior refinement (by non-refined shape prior) | 87.1 | 78.7 | 74.6 | 66.5 | 58.6 | 41.3 | 64.8 |
| (f) Ours without adaptation (with shape prior refinement) | 86.9 | 78.9 | 75.0 | 67.0 | 58.3 | 41.6 | 65.0 |
| (g) Ours (full model) | 87.2 | 78.9 | 75.0 | 68.7 | 60.6 | 41.8 | 65.8 |

less impact on PCP within a reasonable range, $0 < \gamma < 0.2$. Too large $\gamma$ leads to the overlap between FG and BG seeds, which causes difficulty in extracting reasonable PS features. In addition, $\gamma$ has no impact on computational cost; see Fig. 7 (b). Those results reveal the nice property of the proposed PS feature, namely robustness to $\gamma$ selection. From the results shown above, $\gamma = 0.1$ was selected.

### 6.4 Comparative Experiments

Tables 2, 3 and 4 show the results of quantitative evaluation. For comparison, the results obtained by several other methods (i.e. (a), (b), and (c)) are shown.

The effects of the proposed schemes were evaluated with (d) initial binary PS features obtained only by seeds (i.e. shape priors were not used for part segmentation), (e) PS features obtained by dense shape priors without shape prior refinement, (f) PS features without domain adaptation (i.e. concatenated features defined by Eqs. (4) and (5) were not used), and (g) the full PS features obtained by using all schemes proposed in this paper.

In all results shown in Tables 2, 3 and 4, our proposed method with the full model, (g), outperformed all others. On the other hand, our methods had less impacts in BUFFY. This might be because 1) many images in BUFFY have low contrast that makes segmentation difficult, and 2) people in BUFFY, who were pictured larger than those in IP and LSP, were too over-segmented by SLIC [18]. While more deliberate segmentation methods (e.g. globalPb [17]) might alleviate those problems, they need much computational cost. For example, globalPb took 30 sec or more, while SLIC [18] took around 1 sec for segmentation of each image in IP.

By comparing the results of the base and our methods (c), (d), (e), (f), and (g), we can see the effects of the proposed schemes, namely binary PS feature (difference between (c) and (d)), dense shape prior (difference between (d) and (e)), shape prior refinement (difference between (e) and (f)), and domain adaptation (difference between (f) and (g)). Comparison among (d), (e), (f), and (g) reveals that shape prior refinement has less impact on PCP in contrast to other schemes. For visualizing the effects of the proposed schemes, Fig. 11 shows poses estimated by methods (c), (d), (e), (f), and (g). In this typical example, the score of PCP increases monotonically from (c) to (g), while several schemes cannot improve the PCP score. However, such schemes also get some qualitative improvements. For example, domain adaptation can localize the left leg better as shown in Fig. 11 (g), while the PCP scores of (f) and (g) are

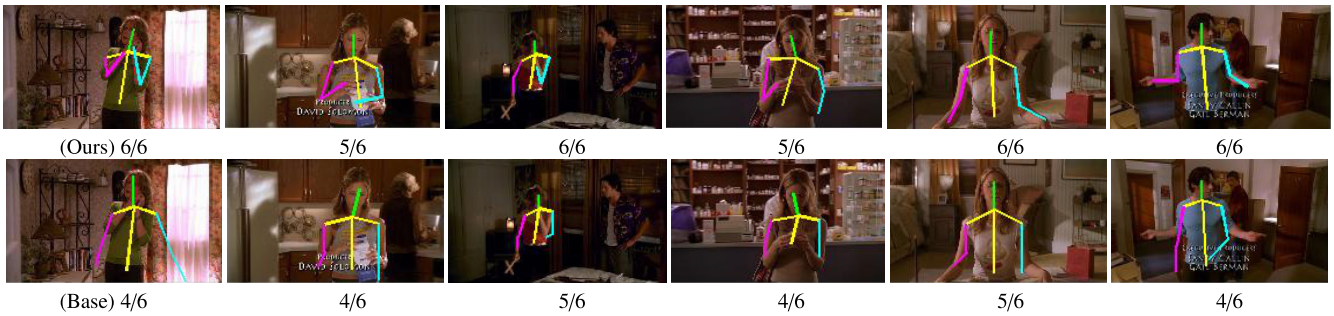



**Fig. 8** BUFFY stickmen dataset: pose estimation results. For each test image, two results are shown: (Top) our method, (Bottom) mixture model of non-oriented parts [25]. The number of correctly localized parts is shown under each result.

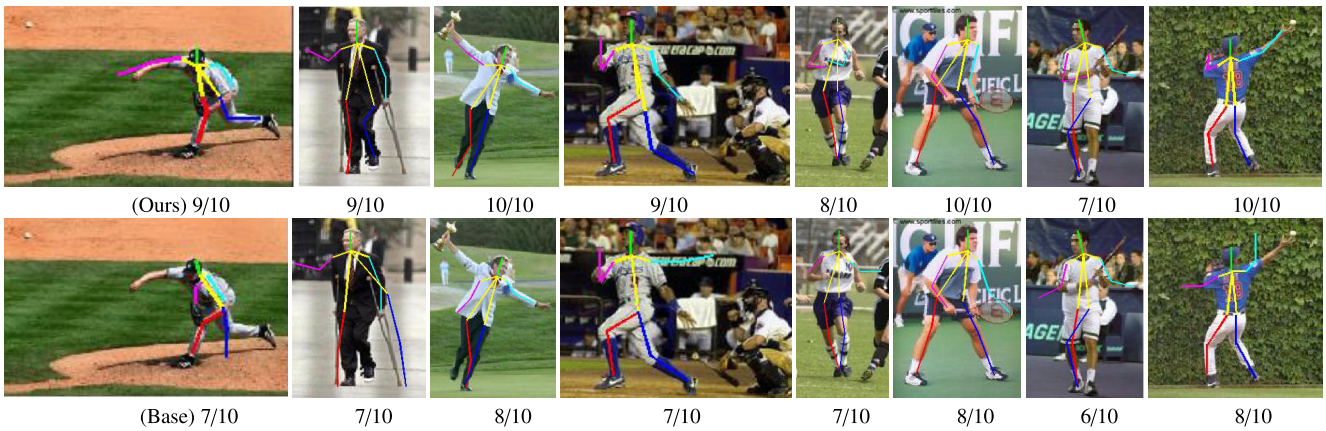



**Fig. 9** IP dataset: pose estimation results.

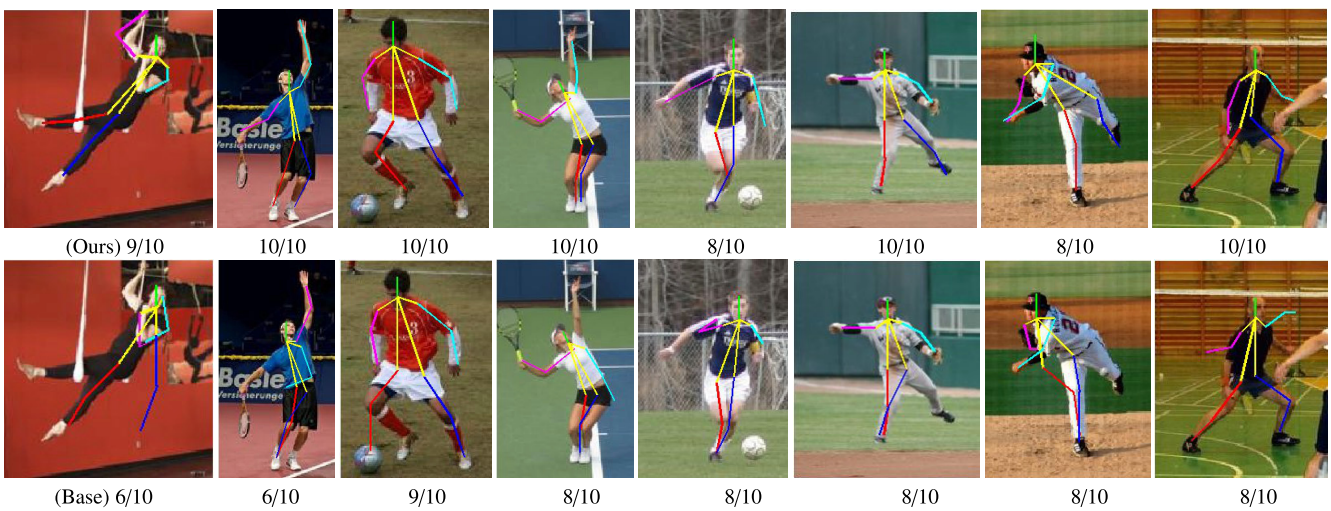



**Fig. 10** LSP dataset: pose estimation results.

equal.

Figures 8, 9 and 10 show examples of results improved by the proposed method with the full model. For visualization, 6 and 10 parts, whose joints are a subset of those of full-body 26 parts, are displayed. The rightmost example in Fig. 9 shows a typical case where the PS features could localize a limb (i.e. lower-arms) without being disturbed by a noisy background.

Figure 12 shows unsuccessful results, where most parts were mislocated, obtained by the proposed method. From the results of BUFFY, it can be confirmed that PS features did not work well in low-contrast regions. The results in IP reveal that heavy foreshortening and self-occlusion disturbed correct pose estimation. The rightmost example was

(c) 5/10 (d) 6/10 (e) 9/10 (f) 10/10 (g) 10/10

**Fig. 11** Effects of the proposed schemes.

**Fig. 12** Examples of typical unsuccessful pose estimation results obtained by the proposed method.

one of terrible results, although the body boundary is relatively observed clearly. Further investigation is needed for finding the causes of these mistakes.

## 7. Concluding Remarks

This paper proposed the part-segment features for evaluating the shape of each part. In training, the PS features are extracted with automatically trained initial seeds and then refined for improving the shape prior on each part. The shape priors are shared by symmetric parts for efficiency. The extracted features are discriminatively trained, and their adaptive weights with respect to gradient features are also learnt. In three publicly-available datasets, the proposed method achieved improvements of 0.8% (from 88.5% to 89.3%), 3.4% (from 79.0% to 82.4%), and 2.9% (from 62.9% to 65.8%) in the rate of successful pose estimation. Future work includes more efficient extraction and discriminative representation of the PS feature.

### References

[1] P.F. Felzenszwalb and D.P. Huttenlocher, "Pictorial structures for object recognition. International Journal of Computer Vision," Int. J. Comput. Vision., vol.61, no.1, pp.55–79, 2005.
[2] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.9, pp.1627–1645, 2010.
[3] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp.886–893, 2005.
[4] N. Ukita, "Part-segment features for articulated pose estimation," 2015 14th IAPR International Conference on Machine Vision Applications (MVA), pp.114–117, 2015.
[5] M.P. Kumar, P.H.S. Torr, and A. Zisserman, "OBJCUT: Efficient Segmentation Using Top-Down and Bottom-Up Cues," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.3, pp.530–545, 2010.
[6] D. Ramanan, "Learning to parse images of articulated bodies," In NIPS, pp.1129–1136, 2006.
[7] M. Eichner and V. Ferrari, "Better appearance models for pictorial structures," Proc. British Machine Vision Conference 2009, pp.3.1–3.11, 2009.
[8] D. Ramanan, "Using Segmentation to Verify Object Hypotheses," 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp.1–8, 2007.
[9] V. Ferrari, M. Marín-Jiménez, and A. Zisserman, "Progressive search space reduction for human pose estimation," 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp.1–8, 2008.
[10] M. Bray, P. Kohli, and P.H.S. Torr, "PoseCut: Simultaneous Segmentation and 3D Pose Estimation of Humans Using Dynamic Graph-Cuts," Computer Vision – ECCV 2006, Lecture Notes in Computer Science, vol.3952, pp.642–655, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
[11] P. Ott and M. Everingham, "Implicit color segmentation features for pedestrian and object detection," 2009 IEEE 12th International Conference on Computer Vision, pp.723–730, 2009.
[12] S. Johnson and M. Everingham, "Combining discriminative appearance and segmentation cues for articulated human pose estimation," 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, pp.405–412, 2009.
[13] G. Mori, X. Ren, A.A. Efros, and J. Malik, "Recovering human body con .gurations:combining segmentation and recognition," Proc. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.326–333, 2004.
[14] V.K. Singh, R. Nevatia, and C. Huang, "Efficient Inference with Multiple Heterogeneous Part Detectors for Human Pose Estimation," Computer Vision – ECCV 2010, Lecture Notes in Computer Science, vol.6313, pp.314–327, Springer Berlin Heidelberg, 2010.
[15] N. Ukita, "Articulated pose estimation with parts connectivity using discriminative local oriented contours," 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp.3154–3161, 2012.
[16] J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol.22, no.8, pp.888–905, 2000.
[17] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol.33, no.5, pp.898–916, 2011.
[18] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," IEEE Trans. Pattern Anal. Mach. Intell., vol.34, no.11, pp.2274–2282, 2012.
[19] W.R. Schwartz, A. Kembhavi, D. Harwood, and L.S. Davis, "Human detection using partial least squares analysis," 2009 IEEE 12th International Conference on Computer Vision, pp.24–31, 2009.
[20] D. Parikh and C.L. Zitnick, "Finding the weakest link in person detectors," CVPR 2011, pp.1425–1432, 2011.
[21] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," CVPR 2011, pp.1385–1392, 2011.
[22] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," IEEE Trans. Syst., Man, Cybern., vol.9, no.1, pp.62–66, 1979.
[23] H. Daumé III, "Frustratingly easy domain adaptation," ACL, pp.256–263, 2007.
[24] Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei, "Hierarchical Dirichlet Processes," J. Am. Stat. Assoc., vol.101, no.476, pp.1566–1581, 2006.
[25] Y. Yang and D. Ramanan, "Articulated Human Detection with Flexible Mixtures of Parts," IEEE Trans. Pattern Anal. Mach. Intell., vol.35, no.12, pp.2878–2890, 2013.
[26] S. Johnson and M. Everingham, "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation," Proc. British Machine Vision Conference 2010, pp.12.1–12.11, 2010.
[27] B. Sapp, C. Jordan, and B. Taskar, "Adaptive pose priors for pictorial structures," 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.422–429, 2010.
[28] B. Sapp, A. Toshev, and B. Taskar, "Cascaded Models for Artic-

ulated Pose Estimation," Computer Vision – ECCV 2010, Lecture Notes in Computer Science, vol.6312, pp.406–420, Springer Berlin Heidelberg, 2010.

[29] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Poselet Conditioned Pictorial Structures," 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp.588–595, 2013.

[30] S. Johnson and M. Everingham, "Learning effective human pose estimation from inaccurate annotation," CVPR 2011, pp.1465–1472, 2011.

**Norimichi Ukita** received the Ph.D. degree in Informatics from Kyoto University, Japan, in 2001. After working as an assistant professor at Nara Institute of Science and Technology (NAIST), he became an associate professor in 2007. He was a research scientist of PRESTO, JST from 2002 to 2006, and a visiting research scientist at Carnegie Mellon University from 2007 to 2009. His main research interests are human behavior analysis such as pose estimation and action recognition.