Zero-shot Learning based Pedestrian Parsing

Xiyang Dai, Ruofei Du, Hao Zhou

University of Maryland, College Park

Abstract. Pedestrian parsing is a fundamental problem for action recognition and behavior analysis. However, unlike indoor person parsing, it remains a challenging problem due to varying luminance, occlusion and clothing. In this paper, we propose a novel pedestrian parsing approach based zero-shot learning. Firstly, we learn an transferred model that extracts clothing parsing attributes from pedestrian images. Then we combine the attributes into higher level human parts, Finally we apply a seed-based segmentation approach to get the parsing results. We test the proposed approach on the Penne-Fudan and PPSS dataset, and achieve reasonablly good results.

Keywords: pedestrian parsing, zero-shot learning, segmentation

1 Introduction

Outdoor pedestrian parsing has significant applications in video surveillance, action recognition and behavior analysis. Given a low-resolution pedestrian image from surveillance camera, our target is to parse the image into separate parts, such as head, hair, upper-body, etc. It is challenging due to the pose differences, customized clothing, varying viewpoints and complicated occlusions. Previous studies mainly focused on using templates[1][2], Bayesian network[3][4] or deep neural network[5] to parse pedestrians. In this project, we propose an novel approach to transfer clothing parsing models into pedestrian parsing by applying category constrains and seed-based segmentation. We apply our method on Penn-Fudan[6] and PPSS[5] datasets and compare our results with state-of-the-art approaches[7][2][5].

The key idea of our approach is to learn an transferred model that deploys clothing parsing resources into pedestrian parsing problem. To achieve this goal, we first refine the existing clothing parsing model to fit low-resolution pedestrian images. Afterwards, we mine the hierarchy relationships between clothing attributes and pedestrian body parts. Finally, we merge clothing segments into body parts using seed-based segmentation method. Meanwhile, we also need to consider the following challenges to implement such an approach:

– The current clothing parsing algorithm is trained on fashion dataset, it is quite different from the dataset we will use for pedestrian parsing. For example, the images in the fashion dataset are usually clear and in high resolution, however, those for pedestrian parsing are generally blurred and in low resolution.

- There are more pose variants in pedestrian parsing dataset. People in the fashion dataset are usually in a "model" pose, but people in pedestrian parsing dataset usually have varying poses.

All these challenges may lead to inaccurate parsing of pedestrians. We examine how the differences affect the pedestrian parsing through experiments and make our approach robust to inaccurate clothing parsing.

The major contributions of our approach are as following:

- To our best knowledge, this is the first zero-shot learning approach to parse outdoor pedestrians.
- Further experiments show our approach is robust to weak attribute classifiers.
- Evaluation on two challenging datasets demonstrates reasonable performance compared to the state-of-the-art methods.

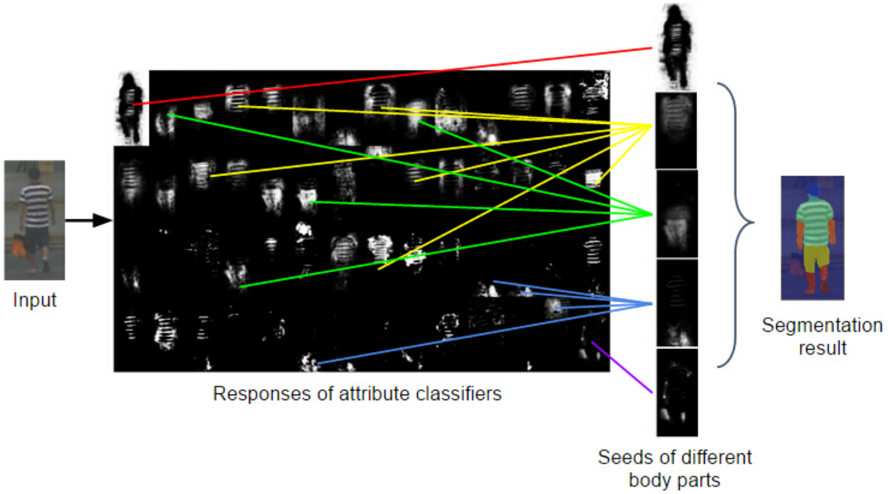
Input
Responses of attribute classifiers
Seeds of different body parts
Segmentation result

Fig. 1. The architecture of our approach.

2 Related Works

We review the recent works on pedestrian parsing and clothing parsing [4, 1, 3, 2, 5, 8–13].

Pedestrian parsing. Current studies generally focus on two kinds of approaches: template matching and probability inference. Bourdev, L. et al. [1] proposed an approach to find common body-part templates, named as poselets.

Then the parsing problem was simplified as a problem to match different poselets. Rauschert, I. et al [2] modeled human body as a tree-structured graph and matched this graph with real images using Bayesian inference. Bo, Y. et al. [3] proposed a bottom-up approach that built a hierarchical model of pedestrian parts based on appearance and shape features. Eslami, S. et al. [4] introduced the Shape Boltzmann Machine (SBM) to pedestrian parsing problem and proposed a multinomial SBM that could capture local and global sharp statistics. Recently, Luo, P. et al. [5] applied deep learning on pedestrian parsing problem with a modified deep decompositional network (DDN) and achieved reasonable performance. The most similar work with this proposal is the very recent paper by Dong, J. et al [8]. In that paper, they tried to build the hierarchical composition of semantic parts under an And-Or graph framework. However, different with this proposal, their reasoning was only limited to body parts and didn't take the advantages of existing clothing parsing model and category cues.

Clothing parsing. Clothing parsing was first proposed by Yamaguchi, K. et al.[11] in 2012 and applied into fashion photographs. They trained a MRF-based image parsing model on clothing parsing problem using their newly collected fashion dataset. Later, in their follow-up paper [9], they further improved the performance using a retrieval-based model that combined pre-trained global parsing models of clothing items and local models of clothing items learned on the fly from retrieved examples. Kalantidis, Y. et al. [13] extended the clothing parsing problem to unconstrained settings by using a probabilistic pose estimator. Recently, Yang, W. et al. [12] proposed a co-parsing approach that jointly parsed a batch of clothing images and achieved better result.

3 Proposed Approach

3.1 Attribute Learning

Similar to [11], we create a pipeline to train our clothing attribute classifiers. We give a brief description of each step in this section.

Superpixel generation: Following recent work [14], we generate an over-segmented set of superpixels of each image. The number of superpixels we generated for each image is usually less than one thousand regions. This allows us to train our attribute classifier on these superpixels and further reduce the scale of learning problem significantly.

Pose estimation: Pose estimation plays an important role in the pipeline. It provides a necessary position prior for our attribute classifier and largely affects the accuracies of our attribute classifiers if it generates wrong poses. Hence, we adapt the widely-used implementation [15] to generate our initial pose. Given an image I, body part p_i and the types t_i of the body part, we define the following score function to evaluate the performance of the pose:

$$S(I, p, t) = \sum_i w_i(t_i)\phi(I, p_i) + \sum_{i,j} w_{i,j}(t_i, t_j)\psi(p_i, p_j) + C(t) \tag{1}$$

The first summation in equation 1 models the potential of appearance for each part, where $\phi(I, p_i)$ is the HOG feature vector[16] for a given part p_i in the image I. The second summation in equation 1 models potential of spatial location of each pair of parts. $\psi(p_i, p_j) = [dx \quad dx^2 \quad dy \quad dy^2]^T$ evaluates the position relationship between two parts where $dx = x_i - x_j$ and $dy = y_i - y_j$ are the relative positions of the two parts. We also apply a third term $C(t)$ to evaluates the configuration of parts $C(t) = \sum_i b_i(t_i) + \sum_{i,j} b_{i,j}(t_i, t_j)$ by counting the co-occurrence of different types of parts. Then, the final labeling results can be inferred by maximizing the score function in equation 1 over p and t. Meanwhile, the learning process can be implemented in a supervised framework. Assume we have labeled positive samples $z = \{I, p, t\}$ generated by manually annotation and negative samples $z' = \{I', p', t'\}$ generated by random sampling, the learning process can be modeled by minimizing a structured prediction objective function:

$$\begin{aligned} &\min_{\beta,\xi_i} \frac{1}{2}||\beta||^2 + C\sum_i \xi_i \\ \text{s.t.} \quad &\beta \cdot \varPhi(x_i, z_i) \geq 1 - \xi_i \\ &\beta \cdot \varPhi(x_i, z_i^{'}) \leq -1 + \xi_i \end{aligned} \tag{2}$$

where β is the model weight parameter, $\varPhi(x, z)$ is the structure expansion function and ξ is the slack variable.

Attribute classification: Given a superpixel of image, we extract the following features:

- Normalized color histgrams with RGB and Lab channels.
- Responses of Gabor features.
- Normalized relative position compared to the image size.
- Normalized relative position compared to the pose parts.

Then we can learn a regression model for each clothing attribute using logistic regression with L2 regularization:

$$\min_w \frac{1}{2}w^T w + C\sum_i \xi(w; x_i, y_i) \tag{3}$$

where the loss function is $log(1 + e^{-y_i w^T x_i})$.

3.2 Category Mining

The purpose of category mining is to find the corresponding relationships between clothing attributes and body parts. This is a non-trivial question because of the complex inter-relationships between clothing attributes and body parts (e.g. a short jacket may belong to upper-body but a long jacket may partially belong to lower-body either) and inner-relationships within clothing attributes (e.g. a man with jeans may be rarely possible to wear short at the same time).

Naive Approach: To fast implement this part, we first carry out category mining manually. We parse the human into eight parts: hair, head, upper-body, lower-body, arm, leg, shoe and background. Our clothing parsing algorithm can give us 56 attributes, however, some of the attributes are ambiguous to categorize, such as "hat", "sunglasses", "accessories" and "belt". As these categories are tiny and not very important, we discarded those attributes. A special attribute is "dress" which contains both upper-body part and lower-body part, we cannot discarded this category since it covers a large part of the body which is important for human parsing. To deal with the ambiguity, we use a soft assignment to assign "dress" to upper-body and lower body. We treat the dress as 70% to be a upper-body and 30% to be a lower-body. Since we only have attribute "skin" in our attribute classifiers, we need to divide it into category "face", "arm" and "leg". We simply divide the skin region to those three categories based on output of the pose estimation results. Table 1 shows some examples of how we classify attributes into eight categories we use.

Table 1. Example attributes belonging to each category.

head, arm, leg	hair	upper-body	lower-body	shoe	background
skin	hair	tights, blazer, t-shirt,coat, blouse, jacket,sweater, ...	shorts,skirt, pants, leggings, ... jeans,	shoes, boots,heels, ...	null, bag, wallet, ...

However, we find that manually classifying the clothing attributes into categories doesn't work well. To make our attribute more robust, we further include some priors and remove outliers.

Prior information: To collaborate with responses from the clothing attributes, we extract spatial prior from the dataset as weights and further apply them when we merge the attributes. For each clothing attribute, the spatial prior can be calculated by computing the empirical probability of each pixel in our training dataset. Figure 2 shows the visualization of the prior information used in our merging method.

Outliers removal: To remove noisy outlier responses from attribute classifiers, we fit Gaussian distribution based on the relative location and response strength. Then we filter out outliers that are three times variance away from the mean. In this way, we can further eliminate the noises generated along with our attribute classifiers.

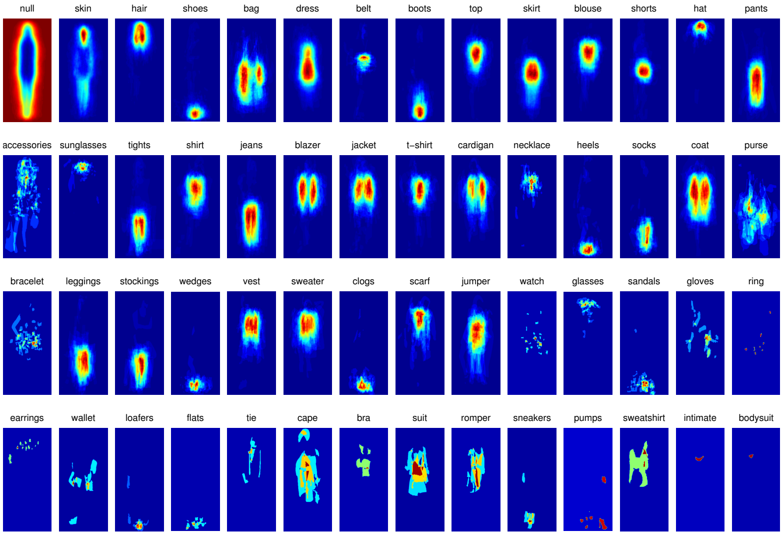

Fig. 2. The per-pixel frequency counts of the clothing attributes in the dataset (sorted in descending order).

3.3 Seed-based segmentation

In previous stage, we have generated the probability distribution according to each higher-level category including hair, head, upper-body, down-body, leg, arm and shoes. Then we generate seeds using an iterative algorithm. Based on both foreground and background seeds, we adapted a multi-level banded graph-cut to segment each category. Finally, we combine and refine multi-label segmentation by banded graph-cut. This problem is formulated as follows:

Given an image $I = \{p_i\}$, our goal is to assign each pixel p_i of I with a segmentation label $l \in L$, here $L = \{0, 1, .., N\}$ includes all the higher-level categories we discussed above. There are two steps for this multi-label problem: (1) For each category label $l \in L$, use banded multi-level graph-cut method to label the image as l and $\neg l$, thus, we have N binary images. (2) Combine these N binary images together to get the multi-labeled results.

Binary segmentation: We first consider the binary label problem, for example, given the combined priors of upper-body, we aim to label I to be upper-body and non-upper-body. To get some prior information, we need to generate seeds for this category. Since category mining can only provide the probability that each pixel belongs to an attribute, we need to figure out a way to select seeds based on these probabilities for segmentation. Manually selecting a threshold to select seeds for all the images cannot work well due to the large variance and noise from the classifiers. Inspired by [17], we proposed an adaptive method for seed-based binary segmentation. Let P_l $(l \in L)$ represent the probability distribution image from each category mining result. Each pixel of P_l represents the probability that this pixel belongs to category l. We use an iterative way to make use of the probability image to do segmentation. At the ith step, the seeds are selected as:

$$S_f = \{(x,y)|P_l(x,y) \geq f_i\} \tag{4}$$

$$S_b = \{(x,y)|P_l(x,y) \leq b_i\} \tag{5}$$

where S_f and S_b represent the seeds for foreground (labeled as l) and background (labeled as $\neg l$) respectively. f_i and b_i are two thresholds. By making use of these two seeds, we segment the image into three parts: foreground(F_l), background(B_l) and unknown(U_l). To get F_l, we use S_f as seeds for F_l and randomly sample points from rest regions as seeds for $\neg F$. By applying graph cuts, we segment image into F_l and $\neg F_l$ region. We apply similar technique to segment the image into B_l and $\neg B_l$. Then $U_l = \neg F_l \cap \neg B_l$. If $F_l \cap B_l \neq \emptyset$, we stop and report f_i and b_i, otherwise, we decrease f_i and increase b_i by a small number and repeat the process again. After we find the proper f_i and b_i, then we apply graph cuts on the entire image making use of the seeds S_f and S_b.

After we get enough seeds for binary segmentation, we resize the image to K coarse levels for banded graph-cut algorithm using a down-sampling of 2 and $K = 3$ as [18]. Firstly, we solve the graph cut on the coarsest level (K) graph. Then we expanded the segmentation result with a narrow band (usually ±2 pixels) which bounds the candidate boundary of the foreground. We then solve graph-cut on banded graph at level $K-1$. The bands outer layer would be use as background seeds while the bands lower layer would be use as foreground seeds. In this way, the seeds are growing gradually to different coarse levels. We apply this method for each of the labels to get N binary segmented images $R_l, l \in L$ as illustrated in Algorithm 1.

Algorithm 1 Iterative and adaptive algorithm for binary segmentation.

```
for each label $l \in L$ do
  Set the initial foreground, background and two threshold $F_l = \emptyset, B_l = \emptyset$
  Set the initial threshold $f_l = 1, b_l = 0$
  while $F_l \cap B_l = \emptyset$ do
    Update the thresholds $f_i = f_i - \delta; b_i = b_i + \delta$
    Update the seeds $S_f = \{(x,y)|P_l(x,y) \geq f_i\}, S_b = \{(x,y)|P_l(x,y) \leq b_i\}$
    Compute segmentation result $F_l$ using $S_f$ by graph-cut
    Compute segmentation result $B_l$ using $S_b$ by graph-cut
  end while
  Use the final seeds $S_f, FS_b$ for banded graph-cut segmentation.
  $R_l$ = Segmentation result from the down-sampled image by a factor of $2^K$
  for each level $k \in K-1, ..., 1$ do
    Expand the result with a narrow band ±2 pixels $\Delta_k$
    $R_l = R_l \cup$ Segmentation by solving graph-cut on the banded graph $\Delta_k$
  end for
  Return the segmentation result $R_l$.
end for
```

Combining binary segmentation: After we get the n binary segmented images, we need to combine them together to get the final multi-label image. Since the n binary segmented images are independently got, it is quite likely that a pixel will have multiple labels. For example, a pixel may be labeled as "upper-body" and "face". Inspired by the method proposed in [18], we use the banded graph-cut to merge two segmented binary images R_i and R_j.

Algorithm 2 Combining multiple lables.

```
$ob = empty$, $ba = fullimage$;
for each label $i$ do
    $ba = ba \cap ba_i$
    For each label $k$ existing in $ob$, get $r = ob_i \cap ob_k$, extended it with $r_i$ and $r_k$, do binary segmentation.
    Refine $ob_k$ in $ob$ and $ob_i$
    $ob = ob \cup ob_i$
end for
```

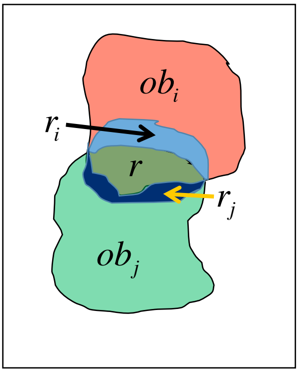

Fig. 3. Illustration of combining multiple labels.

We know that B_i is a binary segmented images, it has two regions ob_i and ba_i representing object region and background region respectively. Let C represent the combination result of two binary segmented images B_i and B_j respectively, then we know that C has three regions: region labeled as i, region labeled as j and the background region. We use c_i, c_j and c_{back} to represnet these three regions. Thus:

$$ob_i \cap ba_j \subset c_i \tag{6}$$

$$ob_j \cap ba_i \subset c_j \tag{7}$$

$$ba_i \cap ba_j \subset c_{back}, \tag{8}$$

however, region $r = ob_i \cap ob_j$ cannot be decided. We treat this region r as the initial band and extend it.

Similar to [18], we extended r to include r_i and r_j which are two narrow bands next to r from ob_i and ob_j respectively (shown in Figure 3. Then we do binary segmentation again on region $r \cup r_i \cup r_j$. Pixels from r_i and r_j will be treated as seeds for label i and label j. The process of combining all the labels is shown in Algorithm 2.

4 Experiments

4.1 Attribute Classifiers

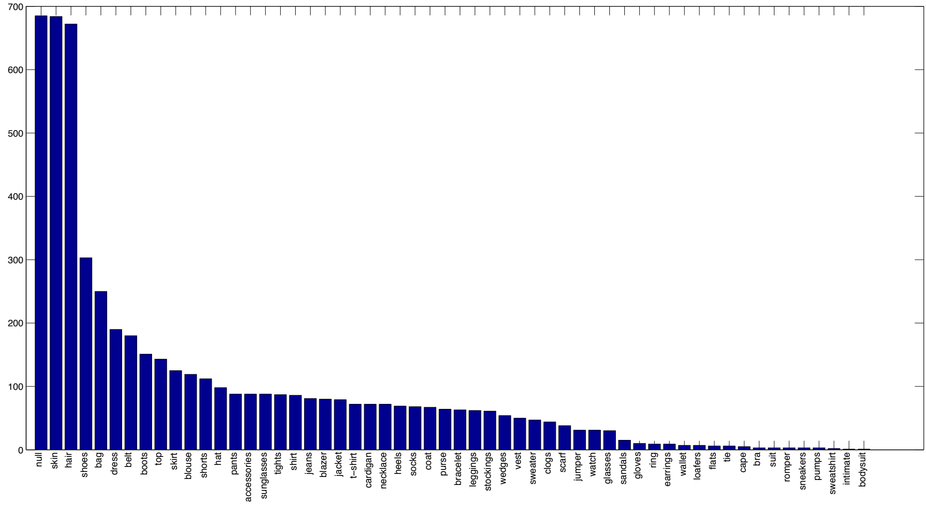

Fig. 4. Illustration of combining multiple labels.

We train our attribute classifiers on Fashionista dataset introduced by recent work[11]. This dataset contains 685 pixel-wise annotated samples with 53 different clothing items and 3 additional labels (hair, skin and null/background). The average number of samples for each class is around 50. However, there are 20 classes with less than 10 samples, which make these classes unreliable. The statistic of this dataset is shown in Figure 4. We randomly select two thirds (456 samples) of the samples as training data and use rest of them as testing data. The confusion matrix that shows the overall accuracies of our attribute classifiers is shown in Figure 5. Table 2 shows the attributes with top 30 accuracies. We notice that the performance of our attribute classifiers is not promosing. But our following experiments show that our approach is robust to these week attribute classifiers.

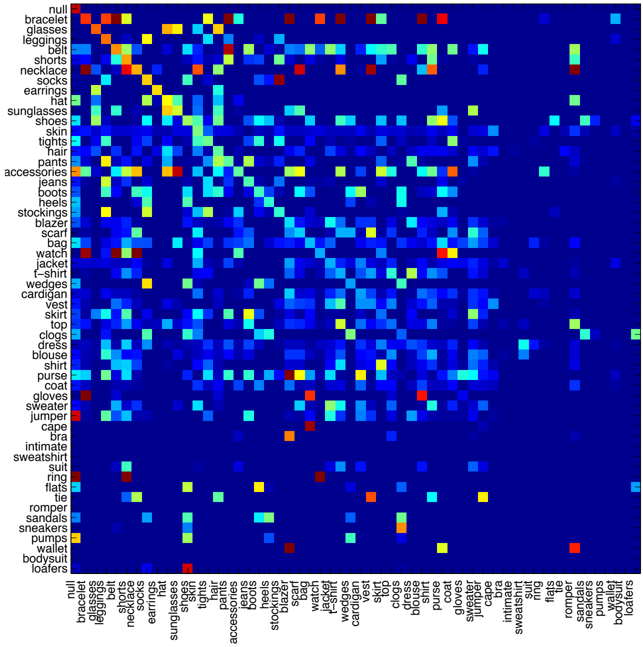

Fig. 5. Illustration of combining multiple labels.

4.2 Testing Datasets

We evaluate our approach on two challenging datasets:

Penn-Fudan pedestrians[6]: This dataset was originally used in pedestrian detection but recently deployed to pedestrian parsing problem. It contains 169 clear images of different pedestrians without occlusion. All images are annotated with 7 different body parts (hair, face, upper-cloth, lower-cloth, shoes, legs and arms);

PPSS[5]: This dataset contains 3673 images from different multiple surveillance scenes and all images are provided with groundtruth annotations, which are similar to Penn-Fudan dataset. Most of the images are blurred and over half of the images (2064) are occluded, which makes this dataset even more challenging.

4.3 Comparison with state-of-the-art

Penn-Fudan dataset: we compare our experimental resutls with [7] [2] and [5]. We show the quantitative results in Table 4. Here "ub" means upper-clothes, "lb" means lower-clothes, "oa1" is the overall precision which is computed directly as the mean of precision of all the parts, "oa2" is the overall precision computed as the weighted mean of the precision of all the other parts, the weight is the number of pixels for each part. We find that our proposed method is comparable

Class	Accuracy	Class	Accuracy	Class	Accuracy
’bg’	0.929	’sunglasses’	0.567	’stockings’	0.4189
’bracelet’	0.8182	’shoes’	0.5234	’blazer’	0.3310
’glasses’	0.7803	’skin’	0.4855	’scarf’	0.3267
’leggings’	0.7618	’tights’	0.4718	’bag’	0.3092
’belt’	0.734	’hair’	0.4628	’watch’	0.2941
’shorts’	0.7104	’pants’	0.4567	’jacket’	0.2857
’necklace’	0.6918	’accessories’	0.456	’t-shirt’	0.2846
’socks’	0.6657	’jeans’	0.4445	’wedges’	0.2805
’earrings’	0.6544	’boots’	0.4251	’cardigan’	0.2744
’hat’	0.6117	’heels	0.4214	’vest’	0.2548

Table 2. Attribute classifiers with top 30 accuracies.

Table 3. Our full results on Penn-Fudan dataset

	bg	ub	lb	shoe	face	arm	leg	hair	oa1	oa2
Our	87.1%	64.9%	56.2%	36.9%	71.2%	43.1%	13.2%	39.8%	59.9%	73.1%

to some of the state-of-the-art methods. We find that generally, we get better results for “face” and “arm”, this is because the clothing parsing method can give us relatively good priors on face and arms since a lot of images in the training data has skins and the appearances of these parts do not change much. Moreover, as we train our clothing parsing model on a high resolution image, the training results may be accurate compared with other method who trained their method on Penn-Fudan dataset. However, due to large variations of clothes and hair style, the parsing results for those parts are not as good as the state-of-the-art method. Since for these regions, our testing dataset is about pedestrian images instead of fashion images, the style of clothing and hair style may be quite different. In Table 3, we show the full parsing results of our method. Our method is the only one that can parse shoes and our accuracy for shoe label is 36.0% which is quite good.

We show some parsing results in Figure 6 and Figure 7. We find that visually, our results look very promising. An interesting observation is that our method can even do multi-pedestrian parsing as shown in the third row of Figure 7.

Table 4. Compare with state-of-the-art for Penn-Fudan

	ub	lb	face	arm	leg	hair	oa1	oa2
Our	64.9%	56.2%	71.2%	43.1%	32.0%	39.8%	51.2%	56.4%
SBP[7]	74.8%	71.2%	60.8%	26.1%	42.0%	44.9%	53.3%	-
P & S[2]	75.2%	73%	42%	24.7%	46.6%	40.0%	50.4%	-
DDN [5]	78.1%	75.0%	54.2%	25.3%	49.8%	44.7%	54.7%	-
DL [5]	77.5%	75.3%	57.1%	27.4%	52.3%	43.2%	56.2%	-

Table 5. Our full results on ppss dataset

	bg	ub	lb	shoe	face	arm	leg	hair	oa1	oa2
Our	92.89%	52.3%	56.1%	2.4%	50.2%	15.9%	0.0%	25.0%	36.8%	77.0%

Table 6. Compare with state-of-the-art on ppss

	ub	lb	face	arm	leg	hair	oa1	oa2
Our	52.3%	56.1%	50.1%	16.0%	02.5%	25.0%	35.4%	49.5%
DL [5]	68.4%	46.1%	29.1%	10.6%	12.9%	22.0%	30.0%	-
DDN [5]	68.4%	61.7%	44.1%	17.0%	23.8%	35.5%	41.8%	-

PPSS dataset: we compare our results with [5]. We show the quantitative results in Table 6. Similar to the performance on Penn-Fudan dataset, our method performs relatively good on face and arms and performs better than only using decomposition layer method mentioned in [5]. However, the performance of our method is worse than the full model proposed in [5]. One reason is that PPSS dataset contains a lot of occlusions, the Deep Decomposition Network in [5] is proposed to deal with occlusions, however we did not spend much effort on occlusion in our method. On the other hand, our method is based on zero-shot learning, the training data we use is Fashionista dataset proposed in [11], in which all the images are in high resolution, images in PPSS dataset, however, are in very low resolution. We also show our full parsing results in Table 5, we find that we have 0% on leg label, this is because the people in PPSS dataset all wear long pants, there is actually no leg label in the dataset. In general, although this dataset is difficult, our method can give a reasonable result.

Some parsing results in PPSS date set are shown in Figure 8 and Figure 9. As shown in the fourth row of Figure 9, when pedestrian have some wired poses, the results we get are quite bad. The reason is that our pose estimator cannot give an accurate pose esitmation if the pose does not appear in the training dataset.

5 Conclusion

In this project, we propose a zero-shot learning approach to parse pedestrian in outdoor scene. We train multiple attribute classifiers by taking the advantage of current clothing parsing resources. Then we merge clothing attribute responses into body-part category seeds using prior information and apply multi-label seed-based segmentation to get the final parsing result. Further experiments show that our approach is robust to the weak attribute classifiers and demonstrate comparable performance on two state-of-the-art testing datasets.

6 Future Work

Our current attribute classifier heavily depends on the initial pose generated by separated DPM based pose estimator. However, such pose estimator does not perform well in outdoor unrestricted configuration due to the large varieties of poses, scales and lights. The interesting followup is how to design attribute classifiers independent with poses. Hence, we plan to design an interactive loop to train and refine pose estimator along with attribute classifier:

- Initially train attribute classifiers without pose.
- Estimate pose from initial segmentation result.
- Retrain attribute classifiers with pose.
- Refine pose from new segmentation result

Meanwhile, we will also explore more advanced category modeling methods that may better generate the body-part seeds. We expect to further improve the performance of our approach and make it competitive with state-of-the-art approaches.

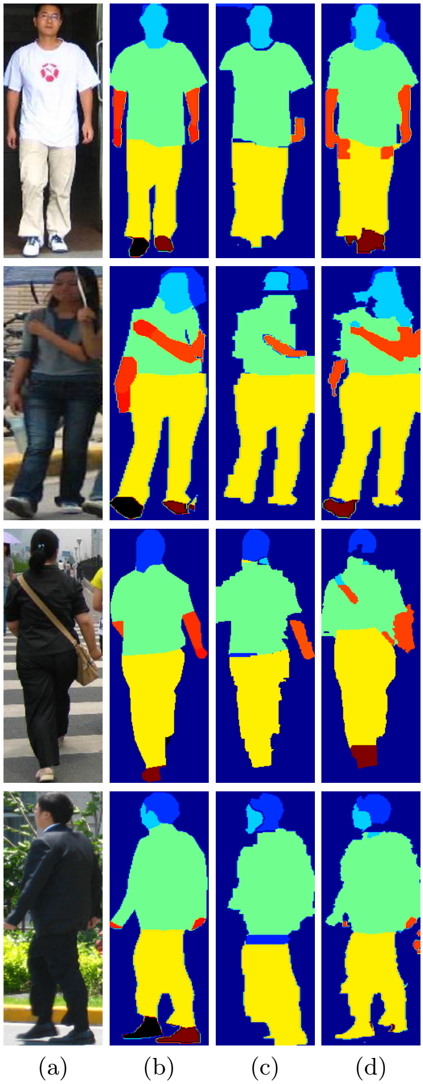

Fig. 6. Some good results. (a) shows the source image. (b) shows the ground truth label. (c) shows the results from [7]. (d) shows the proposed method.

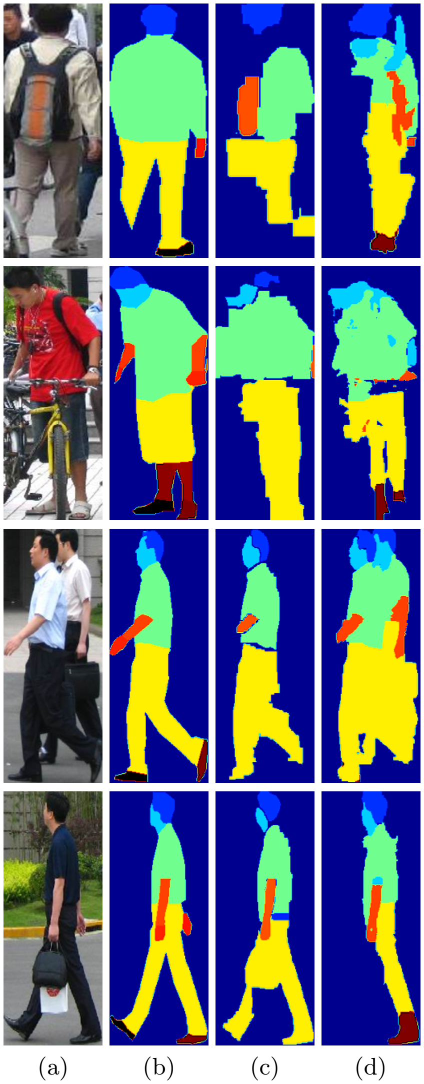

Fig. 7. Some bad results.(a) shows the source image. (b) shows the ground truth label. (c) shows the results from [7]. (d) shows the proposed method.

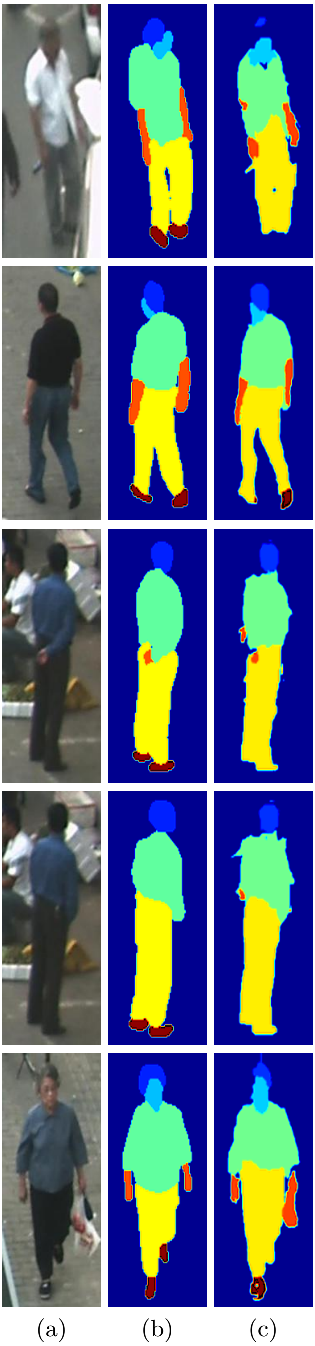

Fig. 8. Some good results. (a) shows the source image. (b) shows the ground truth label. (c) shows the proposed method.

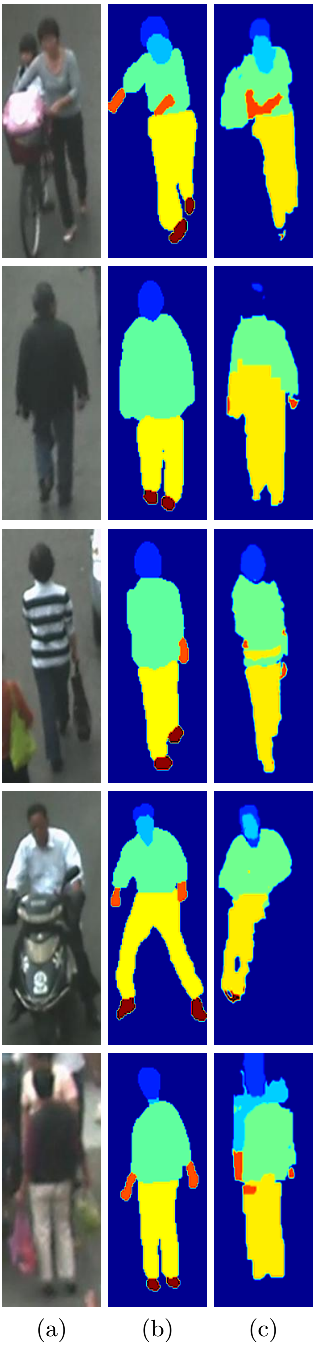

Fig. 9. Some bad results.(a) shows the source image. (b) shows the ground truth label. (c) shows the proposed method.

References

1. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: IEEE 12th International Conference on Computer Vision, 2009. (Sept 2009) 1365–1372
2. Rauschert, I., Collins, R.T.: A generative model for simultaneous estimation of human body shape and pixel-level segmentation. In: Proceedings of the 12th European Conference on Computer Vision - Volume Part V. ECCV'12, Berlin, Heidelberg, Springer-Verlag (2012) 704–717
3. Bo, Y., Fowlkes, C.C.: Shape-based pedestrian parsing. In: The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011. (2011) 2265–2272
4. Eslami, S.M.A., Williams, C.K.I.: A generative model for parts-based object segmentation. In: Proceedings of 26th Annual Conference on Neural Information Processing Systems 2012. (2012) 100–107
5. Luo, P., Wang, X., Tang, X.: Pedestrian parsing via deep decompositional network. In: Computer Vision (ICCV), 2013 IEEE International Conference on. (Dec 2013) 2648–2655
6. Wang, L., Shi, J., Song, G., Shen, I.f.: Object detection combining recognition and segmentation. In Yagi, Y., Kang, S., Kweon, I., Zha, H., eds.: Computer Vision ACCV 2007. Volume 4843 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2007) 189–199
7. Bo, Y., Fowlkes, C.: Shape-based pedestrian parsing. (2011)
8. Dong, J., Chen, Q., Shen, X., Yang, J., Yan, S.: Towards unified human parsing and pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 201. (June 2014) 843–850
9. Yamaguchi, K., Kiapour, M.H., Berg, T.L.: Paper doll parsing: Retrieving similar styles to parse clothing items. In: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE (2013) 3519–3526
10. Nataraj Jammalamadaka (IIIT Hyderabad), Ayush Minocha (IIIT Hyderabad), D.S.I.H.C.J.: Parsing clothes in unrestricted images. In: Proceedings of the British Machine Vision Conference (BMVC), BMVA Press (2013)
11. Yamaguchi, K., Kiapour, M., Ortiz, L., Berg, T.: Parsing clothing in fashion photographs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012. (June 2012) 3570–3577
12. Yang, W., Luo, P., Lin, L.: Clothing co-parsing by joint image segmentation and labeling. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2014) 3182–3189
13. Kalantidis, Y., Kennedy, L., Li, L.J.: Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In: Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval. ICMR '13, New York, NY, USA, ACM (2013) 105–112
14. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5) (May 2011) 898–916
15. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition. CVPR '11, Washington, DC, USA, IEEE Computer Society (2011) 1385–1392
16. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Volume 1. (June 2005) 886–893 vol. 1

17. Liu, J., Sun, J., Shum, H.Y.: Paint selection. In: ACM Transactions on Graphics (ToG). Volume 28., ACM (2009) 69
18. Lombaert, H., Sun, Y., Grady, L., Xu, C.: A multilevel banded graph cuts method for fast image segmentation. In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. Volume 1., IEEE (2005) 259–265