Pedestrian Parsing via Deep Decompositional Network

Ping Luo[1,3] Xiaogang Wang[2] Xiaoou Tang[1,3*]
[1]Department of Information Engineering, The Chinese University of Hong Kong
[2]Department of Electronic Engineering, The Chinese University of Hong Kong
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
pluo.lhi@gmail.com xgwang@ee.cuhk.edu.hk xtang@ie.cuhk.edu.hk

Abstract

We propose a new Deep Decompositional Network (DDN) for parsing pedestrian images into semantic regions, such as hair, head, body, arms, and legs, where the pedestrians can be heavily occluded. Unlike existing methods based on template matching or Bayesian inference, our approach directly maps low-level visual features to the label maps of body parts with DDN, which is able to accurately estimate complex pose variations with good robustness to occlusions and background clutters. DDN jointly estimates occluded regions and segments body parts by stacking three types of hidden layers: occlusion estimation layers, completion layers, and decomposition layers. The occlusion estimation layers estimate a binary mask, indicating which part of a pedestrian is invisible. The completion layers synthesize low-level features of the invisible part from the original features and the occlusion mask. The decomposition layers directly transform the synthesized visual features to label maps. We devise a new strategy to pre-train these hidden layers, and then fine-tune the entire network using the stochastic gradient descent. Experimental results show that our approach achieves better segmentation accuracy than the state-of-the-art methods on pedestrian images with or without occlusions. Another important contribution of this paper is that it provides a large scale benchmark human parsing dataset[1] that includes $3,673$ annotated samples collected from 171 surveillance videos. It is 20 times larger than existing public datasets.

1. Introduction

Pedestrian analysis is an important topic in computer vision, including pedestrian detection, pose estimation, and body segmentation. It has important applications to image and video search, and video surveillance. This paper focuses on parsing a pedestrian figure into different semantic parts, such as hair, head, body, arms, and legs. This problem is challenging because of the large variations of appearance, poses and shapes of pedestrians, as well as the presence of occlusions and background clutters. Some examples are shown in the first column of Fig.1.

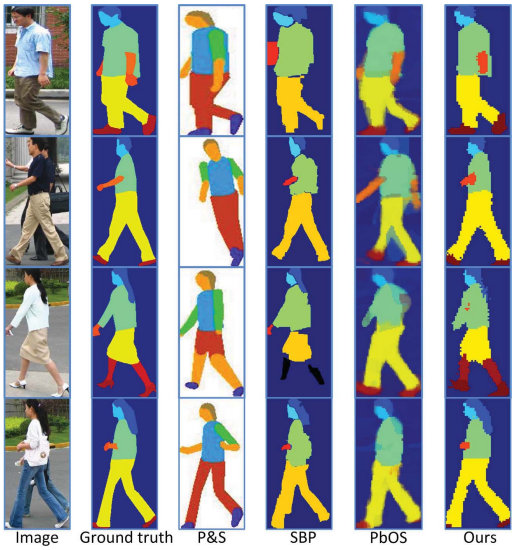

Figure 1. Pedestrian parsing is difficult due to the appearance and pose variations, occlusion, and background clutters. We illustrate the results of DDN compared to P&S [20], SBP [1], and PbOS [6].

Existing studies of pedestrian parsing [2, 1, 6, 20] generally fall into two categories: template matching and Bayesian inference. The pixel-level segmentation of body parts was first proposed in [2], which searches for templates of body parts (poselets) in the training set by incorporating the 3D skeletons of humans. The identified templates are directly used as segmentation results and cannot accurately fit body boundaries of pedestrians in tests. No quatitative experimental evaluation was provided in [2]. Bo et al. [1] (SBP) provided the ground truth annotations of the Penn-Fudan pedestrian database [27] and used it to evaluate the

*This work is supported by the General Research Fund sponsored by the Research Grants Council of Hong Kong (Project No. CUHK 417110, CUHK 417011, CUHK 429412) and National Natural Science Foundation of China (Project No. 61005057).

[1]http://mmlab.ie.cuhk.edu.hk/datasets.html.

segmentation accuracy of their algorithm. Their method segments an image into superpixels and then merges the superpixels into candidate body parts by comparing their shapes and positions with templates in the training set. These approaches rely heavily on training templates. Some examples are shown in the fourth column of Fig.1.

Rauschert et al. [20] (P&S) and Eslami et al. [6] (PbOS) treated human parsing as a Bayesian inference problem. Priors are learned to factorize shape and appearance of pedestrians. They model the appearance prior as Gaussian mixture of pixel colors, and the body shape prior is modeled by the pose skeleton in [20] and the multinomial shape boltzmann machine in [6]. Body shapes and appearance are first generated from the prior models and then verified by being matched with the observed images. The drawback of these approaches is that their appearance models are relatively weak. Their results are sensitive to background clutters, complex poses, and many possible cloth styles. Also, the inference through MCMC is slow. Some examples are shown in the third and the fifth columns of Fig.1.

No existing works consider the factor of occlusions, which occur frequently in video surveillance and can seriously deteriorate the performance of human parsing.

This paper addresses the aforementioned limitations by proposing a new deep model, the Deep Decompositional Network (DDN), which utilizes HOG features [3] as input and outputs the segmentation label maps. HOG features can effectively characterize the boundaries of body parts and estimate human poses. In order to explicitly handle the occlusion problem, DDN stacks three types of hidden layers, including occlusion estimation layers, completion layers, and decomposition layers (see Fig.2 (a)). Specifically, the occlusion estimation layers infer a binary mask, indicating which part of the features is occluded. The completion layers then synthesize the missing features. Finally, the decomposition layers decompose the synthesized features to the label maps by learning a mapping (transformation) from the feature space to the space of label maps (see an example in Fig.2 (a)). Unlike CNN [11], whose weights are shared and locally connected, we find fully connecting adjacent layers in DDN can capture the global structures of humans and can improve the parsing results.

At the training stage, we devise a new strategy based on least squares dictionary learning to pre-train the occlusion estimation layers and the decomposition layers, while the completion layers are pre-trained with a modified denoising autoencoder [26]. The entire network is then fine-tuned by the stochastic gradient descent. At the testing stage, our network can efficiently transform an image into label maps without template matching or MCMC sampling.

Our work has three key *contributions*. (1) This is the first time that deep learning is studied specifically for pedestrian parsing. (2) To the best of our knowledge, this is also the first study to consider the presence of occlusions in human parsing. We propose a novel deep network, where the models for occlusion estimation, data completion, and data transformation are incorporated into a unified deep architecture and jointly trained. This method has advantages over learning each module separately. (3) By carefully designing the architecture of the network and proposing training strategies, the trained DDN not only provides accurate parsing results, but is also robust to occlusions, background clutters, and complex variations of poses and cloth styles, and significantly outperforms state-of-the-art on benchmark datasets. (4) We provide a large-scale benchmark human parsing dataset (refer to footnote 1) which includes $3,673$ annotated samples collected from 171 surveillance videos, making it 20 times larger than existing public datasets.

1.1. Related Work

We review some related works on occlusion estimation [28, 4, 7, 24, 17], data completion [5, 21, 8], and cross-modality data transformation [16, 14, 10].

Occlusion estimation. In [28, 4, 7], all the proposed approaches estimated occlusions with SVM by using HOG features, depth maps and optical flows. Our DDN with deep structures are more powerful than SVM, which is a flat model [8]. Tang et al. [24] employed two restricted Boltzmann machines (RBM) [8] to model the patterns of occlusions and uncorrupted images. In their network, occlusion patterns are sampled from the models and then verified with input images. Ouyang et al. [17, 18] took the part detection scores as the input and used Deep Belief Net (DBN) to estimate the visibility of body parts. In contrast, our model directly maps the input features to occlusion masks.

Data completion. Deep networks are strong generative models for data completion. The deep belief network (DBN) [8] and the deep Boltzmann machine (DBM) [21] both consist of multiple layers of RBMs, and complete the corrupted data using probabilistic inference. Recently, the shape Boltzmann machine (SBM) [5] and multinomial shape Boltzmann machine (MSBM) [6] were proposed to complete discrete data. The denoising autoencoder (DAE) [26] has shown excellent performance at recovering corrupted data, and we have integrated it as a module in our DDN. Instead of completing the missing data, Luo et al. [13] marginalized missing data with proposed deep sum product network for facial attribute recognition.

Data transformation. Several studies have looked at transforming data from one modality, for example, an image, to another, for example, a label map. Ngiam et al. [16] proposed a multimodel deep network that concatenates data across modalities as input and reconstructs them by learning a shared representation. Luo et al. [12] learns

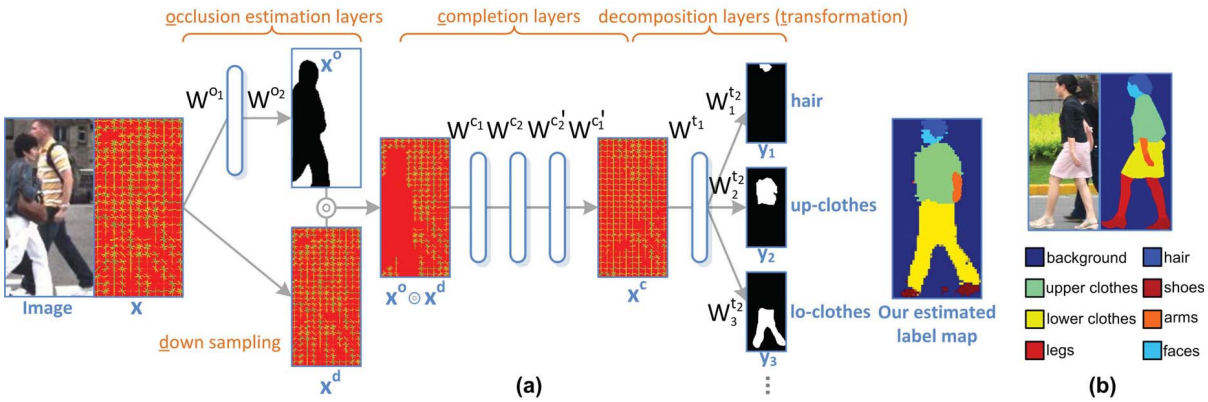

Figure 2. DDN architecture, which combines occlusion estimation, data completion, and data transformation in an unified deep network.

the joint representation of images and label maps for face parsing. Zhu et al. [29] proposed a deep network to transform a face image under arbitrary pose and lighting to a canonical view. Mnih et al. Jain et al. [10] used convolutional neural networks (CNN), which consider data of one modality as input and the corresponding data of the other modality as output. The decomposition layers in DDN are similar to CNN, but with fully-connected layers that capture the global structures of the pedestrians.

2. Network Architecture

Fig.2 (a) shows the architecture of DDN, the input of which is a feature vector x, and the output is a set of label maps $\{y_1, ..., y_M\}$ of body parts. Each layer is fully connected with the next upper layer, and there are one down-sampling layer, two occlusion estimation layers, two completion layers, and two decomposition layers. This architecture works well for pedestrian parsing. More layers can be added for more complex problems.

At the bottom of DDN, the input x is down-sampled to x^d. x is also mapped to a binary occlusion mask $x^o \in [0,1]^n$ through two weight matrices W^{o_1}, W^{o_2}, and biases b^{o_1}, b^{o_2}. Notice that x^o is at the same size as x^d in order to reduce the number of parameters in the network. $x^o_i = 0$ if the i-th element of the feature is occluded, and $x^o_i = 1$ otherwise. x^o is computed as

$$x^o = \tau(W^{o_2}\rho(W^{o_1}x + b^{o_1}) + b^{o_2}), \quad (1)$$

where $\tau(x) = 1/(1 + exp(-x))$ and $\rho(x) = max(0, x)$. The first occlusion estimation layer employs the rectified linear function [15] $\rho(x)$ as the activation function. The second layer models binary data with the sigmoid function.

In the middle of DDN, the completion layers are modeled as the denoising autoencoder (DAE) [26], which utilizes the element-wise product of x^o and x^d as input, and outputs the completed feature vector x^c through four weight matrices $W^{c_1}, W^{c_2}, W^{c_1'}, W^{c_2'}$, and the corresponding biases $b^{c_1}, b^{c_2}, u^{c_1}, u^{c_2}$, where W' is the transpose of W. W^{c_1} and W^{c_2} are encoders that find the compact representation of noisy data by projecting high dimensional data into a low dimensional space. $W^{c_1'}$ and $W^{c_2'}$ are decoders that reconstruct the data. x^c is reconstructed from x^o and x^d as follows,

$$z = \rho(W^{c_2}\rho(W^{c_1}(x^o \circledcirc x^d) + b^{c_1}) + b^{c_2}), \quad (2)$$

$$x^c = \rho(W^{c_1'}\rho(W^{c_2'}z + u^{c_2}) + u^{c_1}), \quad (3)$$

where z is the compact representation and $\circledcirc$ denotes the element-wise product.

On the top of DDN, the completed feature x^c is decomposed (transformed) into several label maps $\{y_1, ..., y_M\}$ with the corresponding weight matrices $W^{t_1}, W_1^{t_2}, ..., W_M^{t_2}$, and biases $b^{t_1}, b_1^{t_2}, ..., b_M^{t_2}$. Each label map $y_i \in [0,1]^n$ is estimated by

$$y_i = \tau(W_i^{t_2}\rho(W^{t_1}x^c + b^{t_1}) + b_i^{t_2}), \quad (4)$$

where $y_{ij} = 0$ indicates the pixel belongs to the background and $y_{ij} = 1$ indicates the pixel is on the corresponding body part.

3. Training Algorithms

Training DDN is done by estimating a set of weight matrices and corresponding biases. It is challenging because of the huge amount of parameters. If the dimensions of the input feature vector and the output label maps are $8,000$ and $10,000$. our network has millions of parameters. We pre-train DDN in a layer-wise manner to initialize the parameters, and then fine-tune the entire network.

3.1. Pre-training Occlusion Estimation Layer

The occlusion estimation layers infer a binary mask x^o from an input feature x. We cannot employ RBMs as in [8] to unsupervised pre-train these layers, because our input and output data are in different spaces. We devise a supervised method based on the least squares dictionary learning to pre-train these layers. We construct a training

set $X = \{x_i\}$ and $\overline{X^o} = \{\overline{x_i^o}\}$, where the column vectors x_i and $\overline{x_i^o}$ denote a feature and its ground truth mask. Initializing the weight matrices is done in order to optimize

$$\arg \min_{W^{o_1}, W^{o_2}} \| \overline{X^o} - \tau(W^{o_2} H^{o_1}) \|_F^2, \quad (5)$$

where $H^{o_1} = \rho(W^{o_1} X)$ is the output of the first layer as shown in Fig.2, and $\| \cdot \|_F$ is the Frobenius norm. Note that we drop the bias term b for simplification, since $Wx + b$ can be written as $\widetilde{W}\widetilde{x}$ with $\widetilde{W} = [W\ b]$ and $\widetilde{x} = [x'\ 1]'$. Solving Eq.5 is not trivial because of its nonlinearity. However, we can approximate W^{o_1}, W^{o_2} layer-wisely as

$$\arg \min_{W^{o_1}} \| \overline{X^o} - W^{o_1} X \|_F^2, \quad (6)$$

$$\arg \min_{W^{o_2}} \| \overline{X^o} - W^{o_2} H^{o_1} \|_F^2 . \quad (7)$$

We first directly use X to approximate $\overline{X^o}$ with a linear transform W^{o_1}. Once W^{o_1} has been learned, $H^{o_1} = \rho(W^{o_1} X)$ is used to approximate $\overline{X^o}$ again with another linear transform W^{o_2}. Eq.6 and Eq.7 have the closed-form solutions, $W^{o_1} = \overline{X^o} X'(XX')^{-1}$ and $W^{o_2} = \overline{X^o} H^{o_1'} (H^{o_1} H^{o_1'})^{-1}$. In the case when the data set is very large and it is hard to compute the matrix inversion, one can employ the on-line dictionary learning algorithm [23] instead of the closed form solutions to recursively update the weight matrices.

3.2. Pre-training Completion Layers

Our purpose is to synthesize the occluded portion of feature x^d. We cast it as a data reconstruction problem using a strategy similar to DAE [26], which initializes the parameters by a RBM [8] with stochastically corrupted data as input, and then fine-tunes them by minimizing the square error between the reconstructed data and the clean data. This strategy makes it possible to obtain the weight matrices robust to noise.

We pre-train each layer with two steps: parameters initialization and reconstruction error minimization. *In the first step*, for each completion layer, let $\widetilde{v^c}$ be an input, which is the corruption of a clean sample $\overline{v^c}$, and $h^c = \rho(W^c \widetilde{v^c} + b^c)$ be the output. The parameters of this layer are initialized by RBM with an energy function

$$E(\widetilde{v^c}, h^c) = \sum_i \frac{(\widetilde{v_i^c} - b_i^c)^2}{2\sigma_i^2} - \sum_j u_j^c h_j^c - \sum_{i,j} \frac{\widetilde{v_i^c}}{\sigma_i} h_j^c W_{ij}^c, \quad (8)$$

where σ is the standard deviation of the noise, and W^c, b^c, u^c are the weight matrix and biases described in Sec.2. Eq.8 can be minimized using contrastive divergence [8]. The conventional DAE corrupts each training sample with stochastic noise. However, we use the structured noises (as shown in Fig.3) to model occlusion

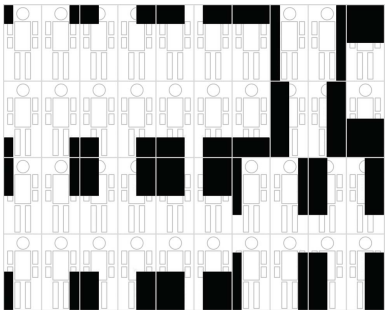

Figure 3. The 40 structured noises templates.

patterns. For each clean sample, we generate 40 corrupted samples by computing the element-wise product between the feature and the 40 templates in Fig.3. *In the second step*, the reconstructed data $v^c = \rho(W^{c'} h^c + u^c)$. We fine-tune the parameters by minimizing the square error between v^c and the clean data $\overline{v^c}$ using gradient descent.

3.3. Pre-training Decomposition Layers

The first decomposition layer transforms the output of the previous layer to a different space through the weight matrix W^{t_1}. The second layer projects the output of the first layer to several subspaces through a set of weight matrices $\{W_i^{t_2}\}$. Therefore, we have

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} = \tau\left(\begin{bmatrix} W_1^{t_2} \\ W_2^{t_2} \\ \vdots \\ W_M^{t_2} \end{bmatrix} h^{t_1} + \begin{bmatrix} b_1^{t_2} \\ b_2^{t_2} \\ \vdots \\ b_M^{t_2} \end{bmatrix} \right), \quad (9)$$

where h^{t_1} is the output of the first decomposition layer. Both decomposition layers can be pre-trained using the strategy introduced in Sec.3.1.

3.4. Fine-tuning

We fine-tune all the parameters of DDN by minimizing the following loss function after pre-training

$$E(X; \mathbf{W}, \mathbf{b}) = \|\overline{Y} - Y\|_F^2, \quad (10)$$

where $X = \{x_i\}$, $\overline{Y} = \{\overline{y_i}\}$, and $Y = \{y_i\}$ are a set of input features, a set of ground truth label maps, and a set of outputs of our network. $\mathbf{W}$ and $\mathbf{b}$ are a set of weight matrices and biases. They are optimized with the stochastic gradient descent. For example, the weight matrices can be updated as

$$\Delta_{i+1} = 0.9 \cdot \Delta_i - 0.001 \cdot \epsilon \cdot W_i^\ell - \epsilon \cdot \frac{\partial E}{\partial W_i^\ell}, \quad (11)$$

$$W_{i+1}^\ell = W_i^\ell + \Delta_{i+1}. \quad (12)$$

$\ell \in \{1, \ldots, L\}$ and i are the indices of layers and iterations. L is the total number of layers. Δ is the momentum variable

[19], ϵ is the learning rate, and $\frac{\partial E}{\partial W^\ell}$ is the derivative. $\frac{\partial E}{\partial W^\ell} = h^{\ell-1}(e^\ell)'$ is computed as the outer product of the back-propagation error e^ℓ and the output of the previous layer $h^{\ell-1}$. In our network, the error e^ℓ is computed in three different ways. For the output layer of DDN,

$$e^L = diag(\overline{y} - y)diag(y)(1 - y), \tag{13}$$

where $diag(\cdot)$ is the diagonal matrix. For the ℓ-th lower layer with the sigmoid function, the backpropagation error is denoted as $e^{\ell,\tau}$,

$$e^{\ell,\tau} = diag(W^{\ell+1'}e^{\ell+1})diag(h^\ell)(1 - h^\ell), \tag{14}$$

where $W^{\ell+1}$ and $e^{\ell+1}$ are the weight matrix and the error of the next layer, and h^ℓ is the output of the ℓ-th layer. For a lower layer with the rectified linear function, the backpropagation error is computed as

$$e_i^{\ell,\rho} = \begin{cases} [W^{\ell+1'}e^{\ell+1}]_i, & \delta_i^\ell > 0 \\ 0, & \delta_i^\ell \leq 0 \end{cases}, \tag{15}$$

where $\delta_i^\ell = [W^\ell h^{\ell-1} + b^\ell]_i$. $[\cdot]_i$ denotes the i-th element of a vector.

4. Experiments

We conduct two sets of experiments. Sec.4.1 evaluates the effectiveness of pre-training. The occlusion estimation layers are pre-trained with 600 images selected from the CUHK occlusion dataset [17], where the ground truth of occlusion masks was obtained as the overlapping regions of the bounding boxes of neighboring pedestrians, *e.g*. the second row of Fig.4. Both the completion and decomposition layers are pre-trained with the HumanEva dataset [22], which contains 937 clean pedestrians with the ground truth of label maps annotated by [1]. Pre-training the decomposition layers requires clean images and their label maps. Pre-training the completion layers requires clean images, the corrupted data of which can be obtained by element-wise multiplication with the 40 occlusion templates shown in Fig.3.

Sec.4.2 shows the results of pedestrian parsing on two datasets: the Penn-Fudan dataset [27] and a new dataset constructed by us. The Penn-Fudan dataset includes 169 pedestrians taken in campus without occlusions. Our pedestrian parsing dataset contains $3,673$ images from 171 videos of different surveillance scenes (PPSS), where $2,064$ images are occluded and $1,609$ are not. The ground truth of label maps for all these images is provided. Some examples are shown in Fig.7, which shows that large pose, illumination, and occlusion variations are present. Compared with Penn-Fudan, PPSS is much larger and more diversified on scene coverage, and is therefore suitable to evaluate the performance of pedestrian parsing algorithms in practical applications.

Figure 4. We show the images and the ground truth masks from the CUHK occlusion dataset in the first two rows. Estimated masks with DDN after pre-training are shown in the last row.

SVM [7]	RoBM [24]	DDN
62.3	72.9	72.3

Table 1. The per-pixel accuracies (%) of occlusion estimation.

4.1. Effectiveness of Pre-training

I. Occlusion Estimation Layers. We compare with structured SVM [7] and RoBM [24] for occlusion estimation on the CUHK dataset [17]. 500 images are selected for training and another 100 images for testing. All of the methods use HOG/mask as input/output pairs for training. Each image and its mask have the size of $[160, 80]$ and $[80, 40]$ respectively. The cell size of HOG is 6, which means that the feature vector has $8,525$ dimensions. Both occlusion estimation layers have $3,200$ neurons. The above settings are adopted in all of the remaining experiments. We augment the original training images by randomly disturbing bounding boxes and randomly changing pixel values in the same way as [9]. Eventually, $50,000$ training samples are obtained. In our method, we run gradient descent for several iterations after the closed-form initializations. Table 1 reports the per-pixel accuracies, while Fig.4 presents some examples. Our result is better than SVM and comparable to RoBM. It is more efficient than RoBM because its pre-training has a closed-form solution.

II. Completion Layers. We compare with PCA [25], DBN [8], DBM [21], and MSBM [6] for data completion. All these approaches are trained on the HumanEva dataset [22] and tested on 100 images with random noises as shown in Fig.3. The two completion layers in DDN have neurons 10^4 and $3,000$, respectively. DBN and DBM have two hidden layers. The architecture of MSBM is the same as [6]. Table 2 reports the mean square errors of the completed feature values of invisible parts. Some exemplar results of DDN are shown in Fig.5 on real occluded images.

PCA [25]	DBN [8]	DBM [21]	MSBM [6]	DDN
121	216	391	420	89

Table 2. The mean square errors of feature completions.

III. Decomposition Layers. We compare with bimodel

bMAE [16]	CNN [11]	DDN
57.2	71.8	72.9

Table 3. The per-pixel accuracies (%) of label maps.

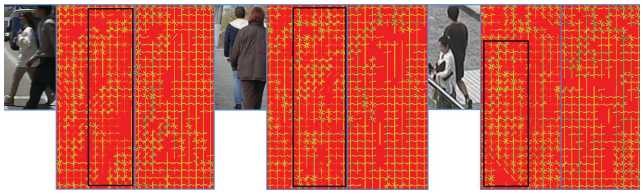

Figure 5. Examples of feature completion of DDN. The image is shown on the left, the original HOG features with the occluded region (black box) in the center, and the completed features on the right. For instance, the features of the right leg of the woman in the first image is well completed.

autoencoder (bMAE) [16] and CNN [11] for data transformation on the HumanEva dataset. We randomly choose 100 images for test and the remainder for training. Table 3 reports the per-pixel accuracies for the above algorithms. bMAE has two hidden layers. CNN has three convolutional layers, and each layer has 32 filters. DDN outperforms bMAE and is slightly better than CNN.

This section shows that the pre-training results of DDN has achieved performance that is at least comparable to the state-of-the-art. However, the major advantage of DDN is that all the three modules can be well integrated into an unified deep architecture and jointly fine-tuned for the ultimate goal of human parsing. Sec.4.2 shows that the performance of human parsing is significantly improved after fine tuning. The performance also greatly outperforms the baseline of simply cascading the best performer from the state-of-the-art on each of the three tasks.

4.2. Pedestrian Parsing

Results on Penn-Fudan Dataset. This experiment is conducted on images without occlusions. DDN is fine-tuned with the HumanEva dataset [22]. It takes two hours to train our network on one NVIDIA GTX 670 GPU, and takes less than 0.1 second to parse an image. We also show the results by using only the decomposition layers (DL), which are trained with the HumanEva dataset.

Table 4 (a) reports the human parsing accuracy of DDN compared with SBP [1], P&S [20], and PbOS [6]. The table includes segmentation results on six fine-scale regions: "hair", "face", "up-cloth" (upper clothes), "arms", "lo-cloth" (lower clothes), and "legs"; and also on five coarse-scale regions: "head", "up-body" (upper-body), "lo-body" (lower body), "FB" (foreground), and "BG" (background). The definitions of these regions are illustrated in Fig.2 (b). PbOS [6] did not report its result on the fine-scale regions.

For fine-scale regions, DDN outperforms both P&S and SBP on the averaged accuracies. It achieves the best results on four regions except "face" and "arms". SBP has the best accuracy on "face" because its template matching works well in this case, as "face" has similar shape and appearance. For the coarse-scale regions, DDN performs best on all the regions. DDN adopts HOG features and the fully-connected network architecture. This design enables it to effectively capture global pose variations. On the other hand, it may lose certain fine-grained descriptive power because HOG is not sensitive to small local changes. This partially explains DDN does not outperform SBP on "face" and "arms". Some segmentation examples of DDN are shown in Fig.6. For our methods, using DL alone achieves better results than DDN, since this dataset has no occlusion, which means that the occlusion estimation and completion layers may slightly induce noise to the DL in the DDN.

Results on PPSS Dataset. We evaluate the robustness of DDN to occlusions with the PPSS data set. Images from the first 100 surveillance scenes are used for training, and those from the remaining 71 scenes for testing. We pre-train DDN as described in Sec.4.1, and fine-tune the network with the training data of PPSS. We also report results of DDN without fine-tining and using DL alone, which are trained on PPSS.

Baselines. Since the implementations of SBP, P&S, and PbOS are unavailable, we cannot evaluate their performance under occlusions. Instead, we cascade RoBM [24], PCA [25], and CNN [11] described in Sec.4.1 as our baseline. These three methods have the best performance among state-of-the-art methods on occlusion estimation, data completion, and data transformation. The RoBM and CNN are tuned on PPSS for fair comparison.

Table 4 (b) reports the parsing accuracy. First, the performance of DL drops significantly when occlusion is present. DL essentially transform the occluded HOG features to the label maps, which is difficult since the feature space of occlusion is extremely large. Fig.8 presents some segmentation examples of DDN compared to DL, and shows that DDN can effectively capture the pose of the human when large occlusion is present, because our network is carefully designed to handle occlusion.

Second, DDN outperforms the baseline and improves the pre-training because it jointly optimizes three types of hidden layers. Fig.7 presents some segmentation results of DDN, and shows that DDN can recover the "upper body" and "lower body" even with the presence of large occlusions, because DDN can capture the global structures and poses of the pedestrians. Note that some small body parts can still be estimated, such as "shoes" and "arms", since the correlations between pixels are implicitly maintained by our network structure.

Fig.9 presents some incorrect results, which show that DDN fails to distinguish some subtle pose variations sometimes. This is partially due to the limitation of HOG as discussed above.

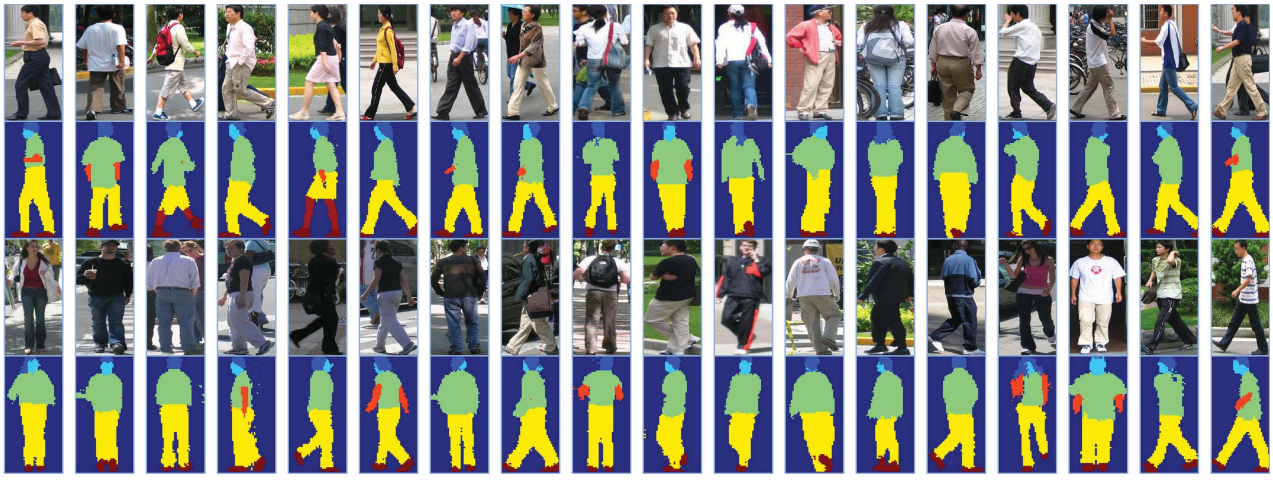

Figure 6. More results of DDN on the Penn-Fudan data set [27].

(a) Segmentation accuracies on Penn-Fudan.

	hair	face	up-cloth	arms	lo-cloth	legs	Avg
SBP [1]	44.9	60.8	74.8	26.2	71.2	42.0	53.3
P&S [20]	40.0	42.8	75.2	24.7	73.0	46.6	50.4
DDN	44.7	54.2	78.1	25.3	75.0	49.8	54.7
DL	43.2	57.1	77.5	27.4	75.3	52.3	56.2
	head	up-body	lo-body	FG	BG		Avg
SBP [1]	51.8	73.6	71.6	73.3	81.0		70.3
P&S [20]	58.2	72.5	72.9	76.2	83.0		72.6
PbOS [6]	54.1	69.9	68.5	71.6	73.8		66.6
DDN	60.2	75.7	73.1	78.4	85.0		74.5
DL	60.0	76.3	75.6	78.7	86.3		75.4

(b) Segmentation accuracies on PPSS.

	hair	face	up-cloth	arms	lo-cloth	legs	Avg
Baseline	29.1	38.7	60.2	17.2	53.0	21.5	36.7
DL	22.0	29.1	57.3	10.6	46.1	12.9	30.0
DDN (pre-train)	29.5	39.0	61.7	16.2	54.6	21.9	37.1
DDN	35.5	44.1	68.4	17.0	61.7	23.8	41.8
	head	up-body	lo-body	FG	BG		Avg
Baseline	38.3	62.6	60.0	67.1	75.0		60.6
DL	30.2	51.5	52.8	59.1	68.6		52.4
DDN (pre-train)	39.1	60.5	57.9	68.4	74.3		59.6
DDN	41.2	69.3	65.5	71.4	80.0		65.5

Table 4. Per-pixel segmentation accuracies (%) on the Penn-Fudan [27] (a) and PPSS (b) datasets.

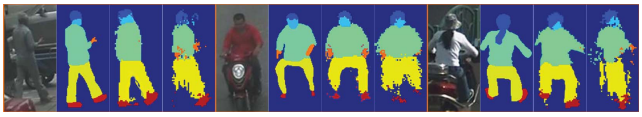

Figure 8. The image, ground truth, the result of DDN, and DL are shown.

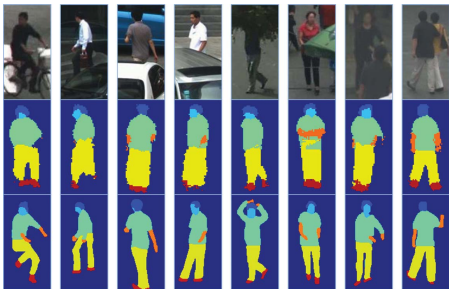

Figure 9. Some incorrect results on the PPSS dataset.

5. Conclusions

We present a new Deep Decompositional Network (DDN) for pedestrian parsing. DDN combines the occlusion estimation layers, completion layers, and the decomposition layers in an unified network, which can handle large occlusions. We construct a large benchmark parsing dataset that is larger and more difficult than the existing dataset. Our method outperforms the state-of-the-art on pedestrian parsing, both with and without occlusions.

References

[1] Y. Bo and C. C. Fowlkes. Shape-based pedestrian parsing. *CVPR*, 2011.

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

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

[4] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. *CVPR*, 2010.

[5] S. Eslami, N. Heess, and J. Winn. The shape boltzmann machine: a strong model of object shape. *CVPR*, 2012.

[6] S. Eslami and C. Williams. A generative model for parts-based object segmentation. *NIPS*, 2012.

[7] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. *CVPR*, 2011.

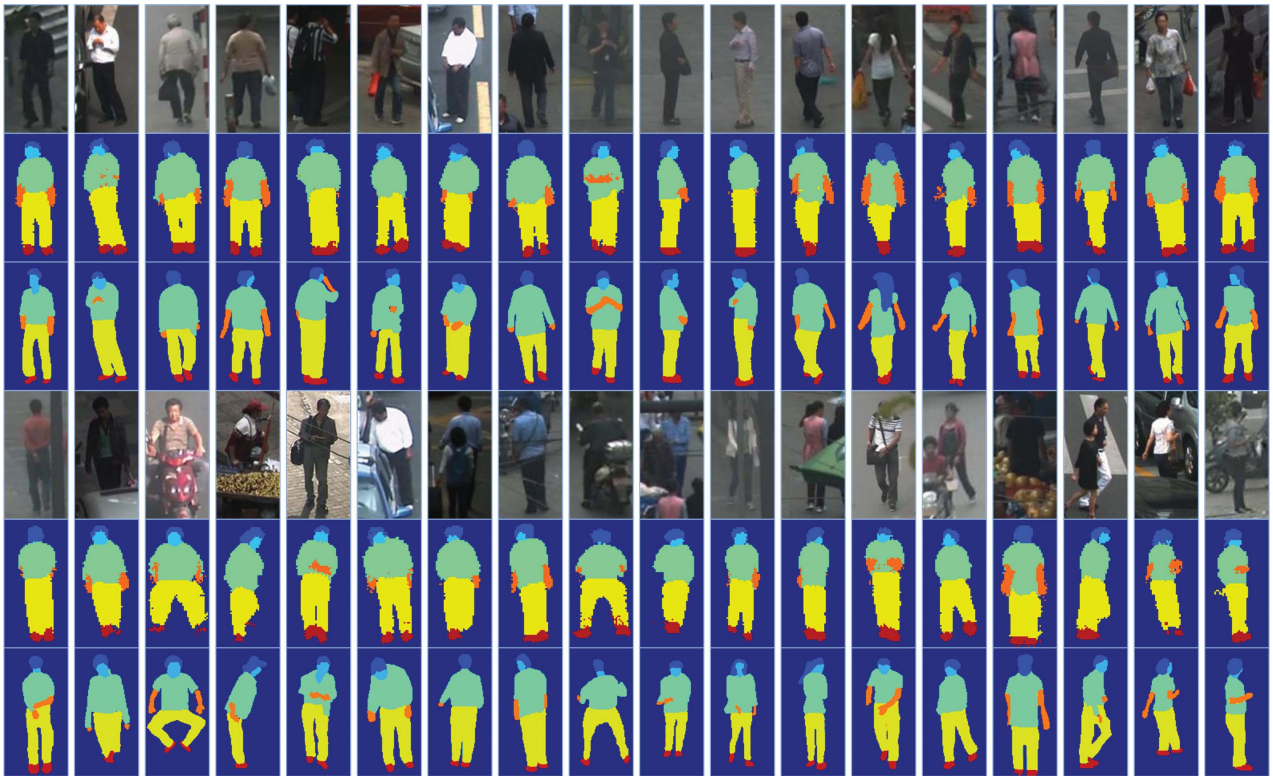

Figure 7. Results of DDN on PPSS dataset. The above three rows are for the unoccluded pedestrians, and the below for the occluded pedestrians, where the images, predicted label maps and ground truthes are shown respectively.

[8] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[9] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arxiv.org, 1207.0580*, 2012.

[10] V. Jain and H. S. Seung. Natural image denoising with convolutional networks. *NIPS*, 2008.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[12] P. Luo, X. Wang, and X. Tang. Hierachical face parsing via deep learning. *CVPR*, 2012.

[13] P. Luo, X. Wang, and X. Tang. A deep sum-product architecture for robust facial attributes analysis. *ICCV*, 2013.

[14] V. Mnih and G. E. Hinton. Learning to label aerial images from noisy data. *ICML*, 2012.

[15] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. *ICML*, 2010.

[16] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. *ICML*, 2011.

[17] W. Ouyang and X. Wang. A discriminative deep model for pedestrian detection with occlusion handling. *CVPR*, 2012.

[18] W. Ouyang and X. Wang. Modeling mutual visibility relationship with a deep model in pedestrian detection. *CVPR*, 2013.

[19] N. Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 1999.

[20] I. Rauschert and R. T. Collins. A generative model for simultaneous estimation of human body shape and pixel-level segmentation. *ECCV*, 2012.

[21] R. Salakhutdinov and G. Hinton. Deep boltzmann machines. *AISTATS*, 2009.

[22] L. Sigal and M. J. Black. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. *Technical Report CS-06-08, Brown University*, 2006.

[23] K. Skretting and K. Engan. Recursive least squares dictionary learning algorithm. *IEEE Trans. Signal Process*, 2010.

[24] Y. Tang, R. Salakhutdinov, and G. Hinton. Robust boltzmann machines for recognition and denoising. *CVPR*, 2012.

[25] M. Turk and A. Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 1991.

[26] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *JMLR*, 2010.

[27] L. Wang, J. Shi, G. Song, and I. fan Shen. Object detection combining recognition and segmentation. *ACCV*, 2007.

[28] X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. *ICCV*, 2009.

[29] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity preserving face space. *ICCV*, 2013.