Partial Occlusion Handling in Pedestrian Detection with a Deep Model

Wanli Ouyang, *Member, IEEE,* and Xingyu Zeng, *Student Member, IEEE,* and Xiaogang Wang, *Member, IEEE,*

***Abstract*—Part-based models have demonstrated their merit in object detection. However, there is a key issue to be solved on how to integrate the inaccurate scores of part detectors when there are occlusions, abnormal deformations, appearances or illuminations. To handle the imperfection of part detectors, this paper presents a probabilistic pedestrian detection framework. In this framework, a deformable part-based model is used to obtain the scores of part detectors and the visibilities of parts are modeled as hidden variables. Once the occluded parts are identified, their effects are properly removed from the final detection score. Unlike previous occlusion handling approaches that assumed independence among the visibility probabilities of parts or manually defined rules for the visibility relationship, a deep model is proposed in this paper for learning the visibility relationship among overlapping parts at multiple layers. The proposed approach can be viewed as a general post-processing of part-detection results and can take detection scores of existing part-based models as input. Experimental results on three public datasets (Caltech, ETH and Daimler) and a new CUHK occlusion dataset[1], which is specially designed for the evaluation of occlusion handling approaches, show the effectiveness of the proposed approach.**

***Index Terms*—Deep model, pedestrian detection, object detection, human detection, occlusion handling**

I. Introduction

Object detection is a fundamental problem in computer vision and has wide applications to video surveillance, image retrieval, robotics and intelligent vehicles. Within the area of object detection, pedestrian detection is one of the most important topics because of its practical applications to automotive safety and intelligent video surveillance. It is the key for driver assistance systems to avoid collisions and to reduce the injury level. For intelligent video surveillance, it provides fundamental information for object counting, event recognition and semantic understanding of videos.

Many classification approaches, features and deformation models have been used for achieving the progress on object detection. The widely used classification approaches include various boosting classifiers [18], [95], [103], probabilistic models [4], [61], linear SVM [12], [24], histogram intersection kernel SVM [58], latent SVM [29], multiple kernel SVM [94] and structural SVM [115]. Features under investigation include Haar-like features [96], edgelets [103], shapelet [79], histogram of gradients (HOG) [12], dense SIFT [94], bag-of-words [45], [49], [61], [94], integral histogram [77], color histogram [97], gradient histogram [116], covariance descriptor [93], co-occurrence features [80], local binary pattern [98], color-self-similarity [97], depth [27], segmentation [22], [26], motion [13], features learned from training data [3], [64] and their combinations [18], [22], [23], [26], [45], [49], [64], [80], [85], [94], [97], [98]. In recent years, deformable part-based models achieved great success on object detection. They mainly model the translational deformation of parts [2], [29], [60], [79], [115]. Other approaches, such as pictorial structures [1], [31], poselet [8], [9] and mixture of parts [15], [88], [109], were also proposed to handle more complex articulations. For object detection, the PASCAL Visual Object Classes (VOC) Challenge [28] and the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) attract much attention [14], [44].

Surveys and performance evaluations on recent pedestrian detection approaches are provided in [20], [25], [35], [62], [101]. Generic detectors [12], [29], [64], [98], [115] assume that pedestrians are fully visible, and their performance degrades when pedestrians are partially occluded. For example, many deformable part-based models [29], [31], [115] summed the scores of part detectors. A pedestrian-existing input window is considered as having a high summed score. If one part is occluded, the score of its part detector could be very low and consequently the summed score will also be low. However, occlusions occur frequently, especially in crowded scenes. Some examples are shown in Fig. 7. As pointed out in [22], the key to successful detection of partially occluded pedestrians is to utilize additional information about which body parts are occluded. For example, the additional informations used in [22] were motion, depth and segmentation results. In our paper, it is only inferred from the appearance of single images through exploring correlations among the visibilities of different parts having different sizes. Once the occluded parts are identified, their effects should be properly removed from the final combined score.

Many previous approaches [11], [98], [103] estimated the visibility of a part by its detection score. However, part detectors are imperfect and such estimation is not accurate. Take the pedestrian in Fig. 1 as an example. The example in Fig. 1 shows 4 meta body parts of the pedestrian: left-head-shoulder, head-shoulder, left-leg and two-legs, which form part hierarchy and will be more precisely defined in Fig. 4 later. Although the left-head-shoulder is visible, its part detector gives a relatively low detection score because the visual cue in the image does not fit this part detector well. Although the left-leg is invisible, its part detector finds a meaningless false-positive window on the baby carriage with a relatively high

W. Ouyang, X. Zeng, and X. Wang are with the Department of Electronic Engineering at the Chinese University of Hong Kong, Hong Kong.
E-mail: wlouyang, xgwang@ee.cuhk.edu.hk.

[1] http://www.ee.cuhk.edu.hk/x̃gwang/CUHK_pedestrian.html

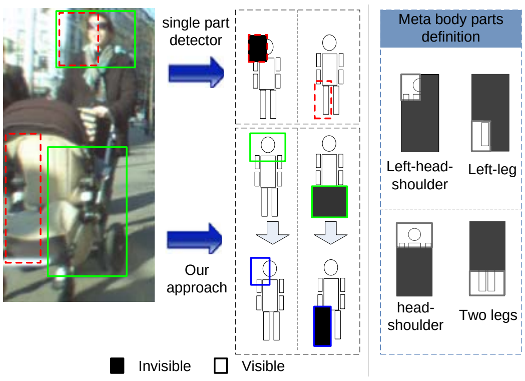


Fig. 1. Estimating the visibility of a part from its detection score or from its correlated parts. Parts estimated as invisible are represented by black rectangles. Part detection scores alone give incorrect visibility estimation. With the help of visibility correlation among parts, our approach can find the correct visibility estimation of the left-head-shoulder and the left-leg successfully.

detection score. If the detection scores of parts are directly used for estimating visibility, the pedestrian will be incorrectly estimated as having the left-head-shoulder invisible and the left-leg visible.

This paper is motivated by the fact that it is more reliable to design overlapping parts at multiple layers and verify the visibility of a part for multiple times at different layers. The detection score of one part provides valuable contextual information for the visibility estimation of its overlapping parts. Take the pedestrian in Fig. 1 as an example. The left-head-shoulder and the head-shoulder are overlapping parts at different layers. Similarly for the left-leg and the two-legs. The head-shoulder has a high detection score because its visual cue in the image fits the corresponding part detector well. If the correlation among parts is modeled in a correct way, the detection score of the head-shoulder can be used to recommend the left-head-shoulder as visible, which rectifies the incorrect estimation from the low detection score of the left-head-shoulder. Similarly, the two-legs has a low detection score because there is no visual cue that fits the corresponding part detector well. The detection score of the two-legs can be used to recommend the left-leg as invisible. Therefore, the major challenges are how to model the relationship of the visibilities of different parts and how to properly combine the results of part detectors according to the estimation of part visibility.

There are two contributions of this paper.

1. A probabilistic framework for pedestrian detection, which models the visibilities of parts as hidden variables. It is shown that various heuristic occlusion handling approaches (such as linear combination and hard-thresholding) are considered as its special cases but did not fully explore its power on modeling the correlations of different parts.

2. A deep model to learn the visibility correlations of different parts, which is inspired by the great success of deep models [5], [40], [47] in various applications such as dimension reduction [41] and recognition [40], [43], [47], [78]. The new model has some attractive features. First, the hierarchical structure of our deep model matches with the multi-layers of the parts model well. Different from the Deep Belief Networks (DBN) in [40], [41], whose hidden variables had no semantic meaning, our model considers each hidden variable as representing the visibility of a part. By including multiple layers, our deep model achieves a better variational lower bound on the likelihood of hidden variables, and in the meanwhile, achieves more reliable visibility estimation. The extended deep model learns to model the constraints among parts and learns to how to combine multiple sources of information, such as the visual dissimilarity between parts, for visibility estimation. Second, it models the complex probabilistic connections across layers with good efficiency on both learning and inference. Third, our deep model only requires the bounding boxes of positive training samples as input without requiring any occlusion information for supervision at the training stage.

Finally, although the above discussions focus on occlusions, the proposed framework is also effective on handling abnormal deformations to some extent. If some parts are abnormally deformed and cannot be detected by part detectors, they can be treated as occlusions and removed from the integration of parts. A primitive version of this paper is published in [67]. This paper provides new extension of the deep model for constraints among parts, more details on parameter learning, more experimental results on the robustness of the model, and more results on other datasets.

II. Related Work

Deformation and occlusion are two major problems to be solved in object detection. To handle the deformation problem, part-based models have been widely used [4], [8], [10], [29], [31], [33], [37], [52], [60], [75], [88], [104], [105], [109], [115]. In these models, the appearance of each part and the deformation among parts were considered. For example, the state-of-the-art approach in [29] combined both the appearance score and the translational deformation score. To model the deformation, various star models [29], [69], [72], tree models [31], [60], [75], [88], [115], loopy graph models [100], complete graph models [7], and Hough transforms [4], [33], [48] were employed. To describe the shape of deformable object parts, tree model with active masks was used in [10]. Detectors using boosting to select features from a large pool of local candidate features also considered objects as being composed of parts [17], [18], [95], [116].

Holistic object detection approaches assume fully visible objects [12], [29] and normally do not work well when objects are occluded. Since visibility estimation plays a key role for detectors to handle occlusions, various approaches [11], [21], [22], [49], [51], [98], [103], [106], [107] were proposed to estimate the visibilities of parts. The SVM responses of the block-wise HOG features were used to determine occlusion maps in [34], [98]. Based on the occlusion maps, Wang *et al.* combined the full-body classifier and part-based classifiers by heuristics [98]. Gao *et al.* [34] summed up the HOG+SVM score for visible HOG cells, and a smoothness prior was used to model the relationship among the binary block-wise labels in the occlusion maps. Leibe *et al.* [49] combined local cues from an implicit shape model [48] and global shape cues via a probabilistic top-down segmentation. Enzweiler *et al.* [22]

segmented each test sample with depth and motion cues to determine occlusion-dependent component weights. In their approach, the detection confidence scores for different parts were used to estimate their visibilities and were computed as weighted means of multiple cues for different parts. Dai *et al.* [11] constructed a set of substructures. Each substructure was composed of a set of part detectors. And the detection confidence score of an object was determined by the existence of these substructures. Following this line, a hierarchical set of substructures were constructed in [21]. The And-Or graph was used in [107] to accumulate hard-thresholded part detection scores. To deal with inter-human occlusions, the joint part-combination of multiple humans was adopted in [49], [53], [83], [103], [106]. These approaches obtain the occlusion map by occlusion reasoning using 2-D visibility scores in [53], [83], [103] or using segmentation results in [49], [106].

Most existing approaches [4], [11], [22], [33], [49], [98], [103], [106], [107] assumed that the visibility of a part was independent of other parts of the same person and estimated the visibility by hard-thresholding the detection scores of parts. Girshick *et al.* [37] designed a specific occlusion pattern and uses grammar model for handling the occlusion in human detection. The approach in [59] uses Integral channel features and learns a set of occlusion-specific classifiers for handling the partial occlusion of pedestrians. However, approach in [59] simply takes summation and maximum scores from these occlusion-specific detectors. These approaches do not learn the visibility relationship among parts, which is the main contribution of our approach. Recently, Duan et al. [21] used manually defined rules to describe the relationship between the visibility of a part and its overlapping larger parts and smaller parts, e.g. if the head or the torso was invisible, its larger part of upper-body should also be invisible. It worked as follows:

1) the binary visibility states of a part was obtained by hard-thresholding its detection score;
2) rules were used to determine whether the combination of the binary visibility states of different parts was correct. If yes, the current window was detected as positive; otherwise, negative.

This approach has certain drawbacks. First, hard-thresholding does not distinguish partial occlusions from full occlusions. A probabilistic model would be a more reasonable way to describe occlusions. Second, a larger part that is misclassified as being occluded by hard-thresholding its detection score cannot be corrected by the rules. Third, the rules were defined manually but not learned from training data. The visibility relationship among parts systematically learned from training data may open the door to more robust methods with a wider spectrum of applications. Considering the problems faced by the approaches discussed above, we propose to use a deep model to automatically learn the probabilistic dependency of the visibilities of different parts.

Deep models have been applied for dimensionality reduction [41], hand written digit recognition [40], [47], [64], object recognition [38], [43], [47], [81], [84], [110], face parsing [55], face recognintion [89], [89], [91], object detection [36], [50], [54]–[57], [65], [66], [70], [71], [90], [111], [112], saliency detection [114], facial expression recognition and scene recognition [78]. Our model is inspired by the deep belief net [41] in learning the relationship among hidden nodes in hierarchical layers. However, our model has some difference with existing works on deep models in spirit. Existing works assume that hidden variables had no semantic meaning and learn many layers of representation from raw data or rich feature representations; our model uses the deep belief net for learning the visibility relationship from compact part detection scores. The deep belief net is used for learning the Bayesian fusion of output scores from a trained classifier. Our method and the structure of our model are highly hand-engineered for the pedestrian detection task. Stacks of convolutional RBM were used in [64] for learning features applied for hand written digit recognition and pedestrian detection. Multi-scale convolutional neural network, pre-trained by unsupervised learning, is used for pedestrian detection in [82]. The approaches in [64], [82] focused on learning features while this paper focuses on learning the visibility relationship among parts. Similar to the relationship between HOG and SVM, the features learned in [64] and our model are complementary to each other with regard to improving the detection performance. The use of a deep model for learning visibility dependency among parts is not available in previous literature. The approach in [113] uses HOG+CSS features, learns holistic detectors and contextual information for improving the performance. The main contribution of our approach is in learning the visibility relationship among multiple parts, which is not present in [113]. Therefore, our approach is orthogonal and complementary to the approach in [113]. The approach in [68] is a continuing work of this paper.

III. A Framework of Pedestrian Detection with Hidden Occlusion Variables

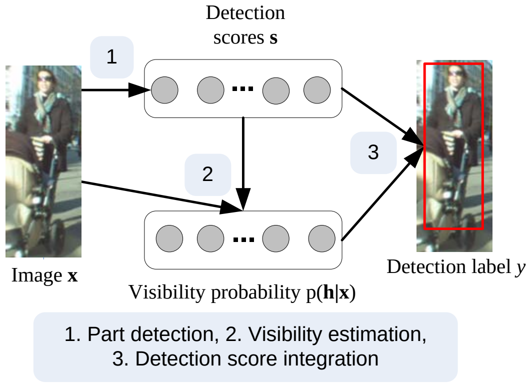
Denote the features of a detection window by $\mathbf{x}$. Denote the label of the detection window by $y \in \{0, 1\}$. Denote the detection scores of the P parts by $\mathbf{s} = [s_1, \ldots, s_P]^{\mathrm{T}} = \gamma(\mathbf{x})$, where $\gamma(\mathbf{x})$ are part detectors. In this paper, it is assumed that part-based models have integrated both the appearance scores and the deformation scores into $\mathbf{s}$. The deformation in the part-based model can be a star model, e.g. [29], or a tree model, e.g. [115]. Therefore, this paper focuses on modeling the visibilities of parts instead of modeling part locations. Denote the visibilities of the P parts by $\mathbf{h} = [h_1, \ldots, h_P]^{\mathrm{T}} \in \{0, 1\}^P$, with $h_i = 1$ meaning visible and $h_i = 0$ meaning invisible. Since $\mathbf{h}$ is not provided at the training or testing stages, it is a hidden random vector. An overview of the framework is shown in Fig. 2.

$p(y|\mathbf{x})$ can be obtained by marginalizing out the hidden variables in $\mathbf{h}$:

$$p(y|\mathbf{x}) = \sum_{\mathbf{h}} p(y, \mathbf{h}|\mathbf{x}) = \sum_{\mathbf{h}} p(y|\mathbf{h}, \mathbf{x}) p(\mathbf{h}|\mathbf{x}). \quad (1)$$

It can be implemented by setting $p(y|\mathbf{h}, \mathbf{x}) = e^{\sum_i y h_i s_i}/Z_1$:

$$p(y|\mathbf{x}) = \sum_{\mathbf{h}} \frac{e^{\sum_i y s_i h_i}}{Z_1} p(\mathbf{h}|\mathbf{x}), \quad (2)$$


Detection scores **s**
1
2
3
Image **x**
Visibility probability p(**h**|**x**)
Detection label *y*
1. Part detection, 2. Visibility estimation,
3. Detection score integration

1. obtain the detection scores $\mathbf{s}$ by part detectors;
2. use $\mathbf{s}$ and $\mathbf{x}$ to estimate visibility probability $p(\mathbf{h}|\mathbf{x})$;
3. combine the detection scores $\mathbf{s}$ with the visibility probability $p(\mathbf{h}|\mathbf{x})$ to estimate the probability of an input window being pedestrian, c.f. (2) and (3).

Fig. 2. Framework overview.

where $Z_1 = 1 + e^{\sum_i h_i s_i}$ is the partition function to make $\sum_y p(y|\mathbf{h},\mathbf{x}) = 1$. Since the weight g_i of s_i in (13) is learn, the effect of the scale of s_i on the posterior $p(y|h,x)$ is automatically compensated during the training process. s_i could be negative. If $\sum_i s_i h_i < 0, \forall i$, then $p(y=1|\mathbf{x}) < p(y=0|\mathbf{x})$ and the corresponding window does not contain pedestrian. The computational complexity of (2) is exponential to the dimension of $\mathbf{h}$. A faster approximate solution to (2) is as follows:

$$p(y|\mathbf{x}) \approx e^{\sum_i y s_i \tilde{h}_i}/Z_2 \propto e^{\sum_i y s_i \tilde{h}_i} \propto_e \sum_i y s_i \tilde{h}_i, \qquad (3)$$

where $b \propto_e a$ means that b is exponentially proportional to a, i.e. $b = k \cdot e^a$ for constant k. $Z_2 = 1 + e^{\sum_i s_i \tilde{h}_i}$ is the partition function to make $\sum_y (e^{\sum_i y s_i \tilde{h}_i}/Z_2) = 1$. $\tilde{h}_i$ is sampled from $p(h_i|\mathbf{h}\backslash h_i, \mathbf{x})$, or alternatively calculated by a mean-field approximation, in which $\mathbf{h}$ is replaced by its average configuration $\tilde{\mathbf{h}} = E[\mathbf{h}|\mathbf{x}]$, which is the expectation of $\mathbf{h}$ over the distribution $p(\mathbf{h}|\mathbf{x})$. The approximation in (3) was also similarly used in [40], [41] for computing the posterior of DBN. Details on this approximation was provided in [5]. $\tilde{h}_i$ is called the visibility term. The log-likelihood for detection rises linearly with the number of parts. This is based on the observation that if many body parts are correctly detected, it is reliable to determine the given sample to be positive. For pedestrians with few parts very reliably detected and the negative effect of their occluded parts removed, their scores should be higher than negative samples with negative part detection scores.

This framework can be used to explain some existing detection approaches, which estimate $\tilde{h}_i$ in (3) in different ways.

Many deformable part-based models [8], [9], [29], [31], [75], [115] can be considered as setting $\tilde{h}_i = 1$ for $i = 1, \ldots, P$ in (3) and have

$$p(y=1|\mathbf{s}) \approx exp(\sum_i s_i)/Z \propto_e \sum_i s_i. \qquad (4)$$

This is essentially a direct sum of part-based detection scores.

After obtaining $\mathbf{s}$ from the part-based model, many occlusion handling methods calculated the $p(y|\mathbf{s})$ as a weighted sum of detection scores. These approaches can be considered as obtaining the $\tilde{h}_i$ in (3) by thresholding detection scores [98], [107], or from other cues like segmentation, depth and motion [22], [49]. With deformation among parts and multiple cues already integrated into s_i, these approaches assumed that the $\tilde{h}_i$ in (3) depends on s_i, i.e. $\tilde{h}_i = f(s_i)$, where f is the mapping of s_i to $\tilde{h}_i$.

In summary, many approaches are special cases of the framework in (3) by setting $\tilde{h}_i = 1$ or by considering the visibility term $\tilde{h}_i$ as only depending on s_i. The full power of this framework on considering the visibility relationship among parts is not explored yet. In this paper, we explore this power and construct a deep model that learns the visibility relationship among parts. In our model, $\tilde{h}_i = p(h_i|\mathbf{h}\backslash h_i, \mathbf{x}) \neq p(h_i|s_i)$ and $p(h_i|\mathbf{h}\backslash h_i, \mathbf{x})$ is estimated from a deep model that will be introduced in the next section.

IV. The Deep Model for Part Visibility Estimation

A. Restricted Boltzmann Machine (RBM)

Since RBM [86] is a building block of our deep model introduced in the next section, an introduction to RBM is provided. Denote the binary observed variables by vector $\mathbf{v} = [v_1, \ldots, v_i, \ldots, v_I]^{\mathrm{T}}$. Denote the binary hidden variables by $\mathbf{h} = [h_1, \ldots, h_j, \ldots, h_J]$. RBM defines a probability distribution over $\mathbf{h}$ and $\mathbf{v}$ as

$$\begin{aligned} p(\mathbf{v},\mathbf{h}) &= \frac{e^{-E(\mathbf{v},\mathbf{h})}}{Z}, \\ \text{where } E(\mathbf{v},\mathbf{h}) &= -\left[\mathbf{v}^{\mathrm{T}}\mathbf{W}\mathbf{h} + \mathbf{c}^{\mathrm{T}}\mathbf{h} + \mathbf{b}^{\mathrm{T}}\mathbf{v}\right], \\ Z &= \sum_{\mathbf{v},\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}. \end{aligned} \qquad (5)$$

$\mathbf{v}$ forms the observed layer and $\mathbf{h}$ forms the hidden layer. Z is the partition function to make $\sum_{\mathbf{v},\mathbf{h}} p(\mathbf{v},\mathbf{h}) = 1$. There are symmetric connections $\mathbf{W}$ between the observed layer and the hidden layer, but no connections between variables within the same layer. The graphical model of RBM is shown in Fig. 3(a). This particular configuration of RBM makes it easy to compute the conditional probability distributions:

$$\begin{aligned} p(v_i = 1|\mathbf{h}) &= \sigma(\mathbf{w}_{i,*}\mathbf{h} + b_i), \\ p(h_j = 1|\mathbf{v}) &= \sigma(\mathbf{v}^{\mathrm{T}}\mathbf{w}_{*,j} + c_j), \end{aligned} \qquad (6)$$

where $\mathbf{w}_{i,*}$ is the ith row of $\mathbf{W}$, $\mathbf{w}_{*,j}$ is the jth column of $\mathbf{W}$, b_i is the ith element of $\mathbf{b}$, c_j is the jth element of $\mathbf{c}$ and $\sigma(t) = (1 + exp(-t))^{-1}$ is the logistic function.

The parameters $\theta = \{\mathbf{W}, \mathbf{c}, \mathbf{b}\}$ in (5) can be learned by maximum likelihood estimation of $p(\mathbf{v})$.

$$p(\mathbf{v}) = \sum_{\mathbf{h}} p(\mathbf{v},\mathbf{h}) = \frac{\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}}{Z} \qquad (7)$$

Recently, many fast approaches have been proposed, e.g. contrastive divergence [39], score matching [42] and minimum probability flow [87].

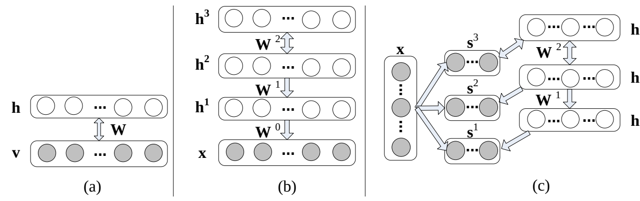


Fig. 3. **(a)** RBM **(b)** DBN, **(c)** our deep model.

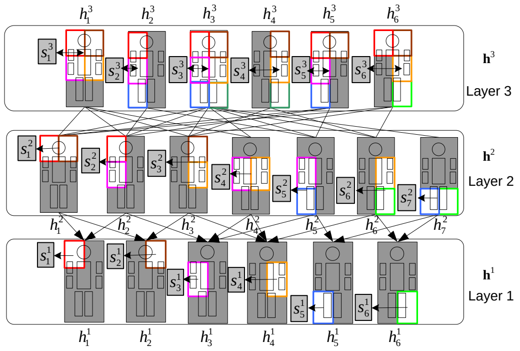


Fig. 4. The parts model used. h_i^l is the visibility state of ith part in the lth layer. For example, h_1^1 indicates the visibility of the left-head-shoulder part. $\mathbf{s}_i^l$ is part specific information of h_i^l, e.g. the detection score of this part.

B. The Deep Model for Visibility Estimation

To use the deep model for visibility estimation, we have $\mathbf{h} = \{\mathbf{h}^1 \cdots \mathbf{h}^L\}$, where $\mathbf{h}^l = [h_1^l \; \ldots h_{P_l}^l]^{\mathrm{T}}$, h_i^l is the visibility state for the ith part at layer l. For our implementation in Fig. 4, $L = 3, P_1 = 6, P_2 = 7$, $P_3 = 6$, and the 19 visibility variables in $\mathbf{h}$ are used in (3) for estimating the detection label y.

1) The Parts Model for Acquiring Detection Score: Our parts model consists of 3 layers as shown in Fig. 4. Parts are assigned to different layers according to their sizes and overlapping relationships. The parts at the bottom layer have the smallest sizes, and the parts at the top layer have the largest sizes. A part at an upper layer is composed of its children at the lower layer.

2) The Deep Model for Visibility Relationship: The graphical model of the proposed deep model is shown in Fig. 3(b). Detailed information is shown in Fig. 4.

There are connections between variables at adjacent layers but no connections between variables at the same layer. A part can have multiple parents and multiple children. In this way, the visibility of one part is correlated to the visibility of other parts at the same layer through shared parents. The probability distribution of $\mathbf{h}^1, \ldots, \mathbf{h}^L$ and $\mathbf{S} = \{\mathbf{s}^1, \ldots \mathbf{s}^l, \ldots \mathbf{s}^L\}$ is as follows:

$$\begin{aligned}
&p(\mathbf{S}, \mathbf{h}^1, \ldots, \mathbf{h}^L) = \left(\prod_{l=1}^{L} p(\mathbf{s}^l|\mathbf{h}^l)\right)\left(\prod_{l=1}^{L-2} p(\mathbf{h}^l|\mathbf{h}^{l+1})\right) p(\mathbf{h}^{L-1}, \mathbf{h}^L, \mathbf{s}^L),\\
&p(h_i^l = 1|\mathbf{h}^{l+1}) = \sigma(\mathbf{w}_{i,*}^l \mathbf{h}^{l+1} + c_i^l), \qquad (8)\\
&p(s_i^l = 1|h_i^l) = \sigma(g_i^l h_i^l + b_i^l),\\
&p(\mathbf{h}^{L-1}, \mathbf{h}^L, \mathbf{s}^L) = e^{\left[\mathbf{h}^{L-1\mathrm{T}} \mathbf{W}^{L-1} \mathbf{h}^L + (\mathbf{c}^{\mathbf{L-1}})^{\mathrm{T}} \mathbf{h}^{L-1} + (\mathbf{c}^{\mathbf{L}} + \mathbf{g}^L \circ \mathbf{s}^L)^{\mathrm{T}} \mathbf{h}^L\right]},
\end{aligned}$$

where $\circ$ denotes the Hadamard product, i.e. $(A \circ B)_{i,j} = A_{i,j} B_{i,j}$. The parameters $\mathbf{W}^l$, $\mathbf{g}^l$ and $\mathbf{c}^l$ are enumerated as follows:

- $\mathbf{W}^l$ models the correlation between $\mathbf{h}^l$ and $\mathbf{h}^{l+1}$, $\mathbf{w}_{i,*}^l$ is the ith row of $\mathbf{W}^l$.
- s_i^l, the ith element in $\mathbf{s}^l$. The $s_i^l \in [0\ 1]$ in (8).
- g_i^l, the ith element in vector $\mathbf{g}^l$, is the weight for the detection score s_i^l.
- $\mathbf{c}^l$ is the bias term.

The detection scores $\mathbf{S}$ have considered both appearance and deformation of parts. Note that, h_i^l and h_j^l are not independent, i.e. $p(h_i^l, h_j^l) \neq p(h_i^l)p(h_j^l)$. In this way, the correlation among parts at the same layer is also modeled.

3) Extension of the Deep Model for Constraints among Parts: In Section IV-B2, the deep model is used for modeling the visibility relationship among parts. This deep model is extended for including constraints among parts in this section. If two correlated parts are visible, they should follow some constraints. The deformation constraint is used in many approaches [8], [9], [29], [31], [75], [115]. Additional cues, such as depth and motion, can also be added into the constraints. In this paper, we consider another constraint, i.e. the visual similarity among parts, which is inspired by the block-wise color similarity in [97]. For example, in Fig. 1 the visual cue of head-shoulder (mainly the head region) is often similar to the visual cue of left-head-shoulder but dissimilar to the visual cue of legs, because legs usually do not have skin and hair colors. This can distinguish a pedestrian from a pole, which is often detected as a false alarm, since the “head-shoulder” of a pole is often visually similar to its “legs”.

The extended model is as follows:

$$\begin{aligned}
&p(h_i^l = 1|\mathbf{h}^{l+1}, \tilde{\mathbf{S}}^l) = \sigma(\mathbf{w}_{i,*}^l \mathbf{h}^{l+1} + c_i^l), && (9)\\
&p(\mathbf{h}^{L-1}, \mathbf{h}^L, \mathbf{s}^L|\tilde{\mathbf{S}}^{L-1}, \tilde{\mathbf{S}}^L)\\
&= e^{\left[\mathbf{h}^{L-1\mathrm{T}} \mathbf{W}^{L-1} \mathbf{h}^L + \mathbf{c}^{L-1\mathrm{T}} \mathbf{h}^{L-1} + (\mathbf{c}^L + \mathbf{g}^L \circ \mathbf{s}^L)^{\mathrm{T}} \mathbf{h}^L\right]}, && (10)\\
&p(s_{i,k}^l = 1|h_i^l) = \sigma(g_{i,k}^l h_i^l + b_{i,k}^l), k = 1, \ldots, K, && (11)\\
&\mathbf{W}^l = \mathbf{W}^{l,0} + \tilde{\mathbf{W}}^l \circ \tilde{\mathbf{S}}^l, \text{for } l = 1, \ldots L-1, && (12)\\
&\mathbf{S} = \{\mathbf{s}^1, \ldots, \mathbf{s}^L, \tilde{\mathbf{S}}^1, \ldots, \tilde{\mathbf{S}}^{L-1}\}.
\end{aligned}$$

- $\tilde{\mathbf{S}}^l$ in (12) is the color histogram dissimilarity, which is obtained from input data. Denote the ith part at layer l by t_i^l. The $\tilde{s}_{i,j}^l$ in $\tilde{\mathbf{S}}^l$ is the color histogram dissimilarity between parts t_i^l and t_j^{l+1}. $\tilde{s}_{i,j}^l (\geq 0)$ is large when t_i^l is dissimilar to t_j^{l+1} while $\tilde{s}_{i,j}^l$ is close to 0 when t_i^l is very similar to t_j^{l+1}. In the experiment, the color histogram with [18 3 3] bins in the [Hue Saturation Value] space for each part is collected. Each histogram is divided by the size of the part in order to remove the effect of size variation of parts. The dissimilarity is evaluated by sum of absolute difference.
- $\mathbf{W}^l$ in (10) and (12) models the correlation between visibility state vectors $\mathbf{h}^l$ and $\mathbf{h}^{l+1}$. $\mathbf{w}_{i,*}^l$ in (9) is the ith row of $\mathbf{W}^l$.
- $\tilde{\mathbf{W}}^l$ in (12) is the weight for pair-wise information $\tilde{\mathbf{S}}^l$. For a pedestrian existing window that has $h_i^l = 1$ and $h_j^{l+1} = 1$, if $\tilde{w}_{i,j}^l < 0$, then the dissimilarity term $\tilde{s}_{i,j}^l$ shall be close to 0 and part t_i^l, e.g. left-head-shoulder, shall be similar to part t_j^{l+1}, e.g. head-shoulder. Similarly, if $\tilde{w}_{i,j}^l > 0$, then part t_i^l, e.g. left-leg, shall be dissimilar to part t_{j+1}^l, e.g. head-shoulder. If $h_i^l = 0$ or $h_j^{l+1} = 0$, then $h_i^l w_{i,j}^l h_j^{l+1} = 0$ and the constraint between t_i^l and t_j^{l+1} is not considered.

- $s_{i,k}^{l}$ in (11) contains multiple sources of part-specific information for part i in layer l. In the experiment, $\mathbf{s}_i^l$ contains two sources of information: 1) $s_{i,1}^{l}$ is detection score of part t_i^l; 2) $s_{i,2}^{l}$ measures the visual similarity between part t_i^l at its deformed position and part t_i^l at its anchor position.
- $g_{i,k}^{l}$ is the weight for $s_{i,k}^{l}$.
- $\mathbf{c}^l$ is the bias term.

For the model in Fig. 3(b), we have $L = 3$. The special case in (8) can be obtained by removing $\tilde{\mathbf{W}}^l \circ \tilde{\mathbf{S}}^l$ and setting $K = 1$ in (11). $\mathbf{W}^{l,0}$, $\tilde{\mathbf{W}}^l$, $\mathbf{g}_i^l$ and $\mathbf{c}^l$ are the parameters to be learned. If additional cues are available, more constraints can be included by extending Eq (9) straightforwardly.

4) Parameter Learning of the Deep Model: Our model has 19 hidden variables to be inferred, i.e. the length of vector $\mathbf{h}$ is 19 in (2). The part detectors are considered as voters and the detection result can be considered as the output of the voting system. To improve the robustness of the voting system, we do not put any hard constraints such as mutual exclusiveness among the values of these hidden variables. Their soft correlations are learned from data. In this way, all part scores are softly combined for estimating the detection label. For example, $h_2^3 = 1$ in Fig. 4 indicates that the left side of a pedestrian is visible, but does not imply that the right side is invisible. It does not imply that its sub-parts h_2^2 and h_5^2 must be visible either. If a pedestrian is fully visible, any h_i^l could be 1. Therefore, there are 2^{19} possible combinations of visibility variables of different parts to enumerate during inference and the probability of each combination needs to be estimated.

Since the proposed model is a loopy graphical model, it is normally time consuming and hard to train. Hinton et al. [40], [41] proposed a fast learning algorithm for deep belief net (DBN) which has shown its success in many applications. In this work, we adopt a similar learning algorithm to train our model. The difference between our model and DBN is as follows:

1) $\tilde{\mathbf{S}}^l$ and $\mathbf{s}^l$ for $l = 1, \dots, L$ in our model are directly estimated from input data by functions $\tilde{\mathbf{S}}^l = \phi(\mathbf{x}, l)$ and $\mathbf{s}^l = \psi(\mathbf{x}, l)$. In this model, we will not model $p(\mathbf{x})$ and $\phi(\mathbf{x}, l)$ is learned by supervised training.
2) With the term $\tilde{\mathbf{w}}_{i,*}^l \circ \tilde{\mathbf{s}}_{i,*}$ added for h_i^l and hidden nodes $s_{i,*}$ connected with h_i^l, each hidden unit h_i^l now has specific meaning related to the semantic meanings of $s_{i,*}^l$ and $\tilde{\mathbf{s}}_{i,*}$ obtained from input data. Taking the term $g_i^l s_i^l$ in (8) for pedestrian detection as an example, if s_i^l is the detection score of part i at layer l, then the hidden unit h_i^l can be considered as the visibility of that part with $h_i^l = 1$ meaning a visible part and $h_i^l = 0$ meaning an occluded part. Without the terms $\mathbf{g}_i^{l^T} \mathbf{s}_i^l$ and $\tilde{\mathbf{w}}_{i,*}^l \circ \tilde{\mathbf{s}}_{i,*}$, which is the case in DBN, the meaning of each hidden unit is not clear.
3) In DBN, observed variables are arranged at the first layer and connected to hidden variables at the second layer. In our model, the observed variables $\tilde{\mathbf{S}}$ and $\mathbf{s}$ are connected to hidden variables at many different layers.

Because of these differences, the learning algorithm of DBN cannot be directly applied to our model. We modified the training and inference algorithms in [40] when applying them to our model.

The training algorithm is to learn the parameters $\theta = \{\mathbf{W}^{l,0}, \tilde{\mathbf{W}}^l, \mathbf{g}_i^l, \mathbf{c}^l\}$ for $l = 1, \dots, L$ and $k = 1, \dots, K$ in (8), with two stages.

1) Stage 1: For l=1 to 2 { Train parameters for layer l and $l+1$ using RBM. }
2) Stage 2: Fine-tune all the parameters by backpropagating error derivatives.

At Stage 1, the parameters are trained layer by layer and two adjacent layers are considered as an RBM that has the following distributions:

$$\begin{aligned}
&p(\mathbf{h}^l, \mathbf{h}^{l+1}, \mathbf{s}^{l+1} | \mathbf{s}^l, \tilde{\mathbf{S}}^l, \tilde{\mathbf{S}}^{l+1}) \\
&= e^{\left[\mathbf{h}^{l\mathrm{T}} \mathbf{W}^l \mathbf{h}^{l+1} + (\mathbf{c}^l + \tilde{\mathbf{c}}^l)^{\mathrm{T}} \mathbf{h}^l + (\mathbf{c}^{l+1} + \tilde{\mathbf{c}}^{l+1})^{\mathrm{T}} \mathbf{h}^{l+1}\right]}, \\
&p(h_i^l = 1 | \mathbf{h}^{l+1}, \mathbf{s}^l, \tilde{\mathbf{S}}^l) = \sigma(\mathbf{w}_{i,*}^l \mathbf{h}^{l+1} + c_i^l + \mathbf{g}_i^{l^T} \mathbf{s}_i^l), \\
&p(h_j^{l+1} = 1 | \mathbf{h}^l, \mathbf{s}^{l+1}, \tilde{\mathbf{S}}^{l+1}) = \sigma(\mathbf{h}^{l^{\mathrm{T}}} \mathbf{w}_{*,j}^l + c_j^{l+1} + \mathbf{g}_j^{l+1^T} \mathbf{s}_j^{l+1}), \\
&p(s_{i,k}^{l+1} | \mathbf{h}^{l+1}) = \sigma(g_{i,k}^{l+1} h_i^{l+1} + b_{i,k}^{l+1}), \\
&p(\mathbf{h}^1, \mathbf{s}^1) = e^{\sum_{i,k} g_{i,k}^1 h_i^1 + b_{i,k}^1 s_{i,k}^1 + c_i^1 h_i^1}, \\
&\mathbf{W}^l = \mathbf{W}^{l,0} + \tilde{\mathbf{W}}^l \circ \tilde{\mathbf{S}}^l, \\
&\tilde{\mathbf{c}}^l = [\tilde{c}_1^l \ \tilde{c}_2^l \dots \tilde{c}_i^l \dots]^T, \tilde{c}_i^l = \mathbf{g}_i^{l^{\mathrm{T}}} \mathbf{s}_i^l,
\end{aligned} \tag{13}$$

where $\mathbf{w}_{i,*}^l$ is the ith row of $\mathbf{W}^l$ and $\mathbf{w}_{*,j}^l$ is the jth column of $\mathbf{W}^l$, $s_{i,k}^l$ for $k = 1, \dots, K$ is the kth element in vector $\mathbf{s}_i^l$, $g_{i,k}^l$ is the kth element in vector $\mathbf{g}_i^l$ of length K. $K = 2$ in our experiment. In the layer-wise pretraining, $\mathbf{s}^1$ is considered as the observed variable and $p(\mathbf{h}^1, \mathbf{s}^1)$ is considered as the RBM for learning the $g_{i,k}^1$,and c_i^1 in (13). Then $\mathbf{h}^1$ is fixed, h^1 and s^2 are considered as the visible vector for training $p(\mathbf{h}^1, \mathbf{h}^2, \mathbf{s}^2 | \mathbf{s}^1, \mathbf{x})$, similarly for $p(\mathbf{h}^2, \mathbf{h}^3, \mathbf{s}^3 | \mathbf{s}^2, \mathbf{x})$. The gradient of the log-likelihood for this RBM is computed as follows:

$$\begin{aligned}
&\frac{\partial L(\mathbf{h}^l)}{\partial w_{i,j}^{l,0}} \propto (< h_i^l h_j^{l+1} >_{data} - < h_i^l h_j^{l+1} >_{model}), \\
&\frac{\partial L(\mathbf{h}^l)}{\partial \tilde{w}_{i,j}^{l}} \propto (< \tilde{s}_{i,j}^l h_i^l h_j^{l+1} >_{data} - < \tilde{s}_{i,j}^l h_i^l h_j^{l+1} >_{model}), \\
&\frac{\partial L(\mathbf{h}^l)}{\partial c_i^l} \propto (< h_i^l >_{data} - < h_i^l >_{model}), \\
&\frac{\partial L(\mathbf{h}^l)}{\partial g_{i,k}^l} \propto (< h_i^l s_{i,k}^l >_{data} - < h_i^l s_{i,k}^l >_{model}), k = 1, 2,
\end{aligned} \tag{14}$$

where $w_{i,j}^{l,0}$ and $\tilde{w}_{i,j}^l$ are the (i,j)th element in matrices $\mathbf{W}^{l,0}$ and $\tilde{\mathbf{W}}^l$ respectively. The contrastive divergence in [39] is used as the fast algorithm for learning the parameters in (13). In the appendix, we prove that this layer-wise training algorithm is optimizing likelihood function $p(\mathbf{h}^l)$ by a lower bound $\sum_{\mathbf{h}^{l+1}} Q(\mathbf{h}^{l+1} | \mathbf{h}^l) \log \frac{p(\mathbf{h}^{l+1}, \mathbf{h}^l)}{Q(\mathbf{h}^{l+1} | \mathbf{h}^l)}$, where $Q(\mathbf{h}^{l+1} | \mathbf{h}^l)$ is the probability learned for layer l and $l+1$ using RBM. At Stage 2, the variables are arranged as a backpropagation (BP) network as shown in Fig. 5 for fine tuning all parameters.

The inference stage is to infer the label y from detection window features $\mathbf{x}$. At the inference stage, we use the framework in (3) for obtaining $p(y|\mathbf{x})$. And the 19 part visibility variables $\tilde{h}_j^{l+1}$ for (3) are obtained using the BP network in

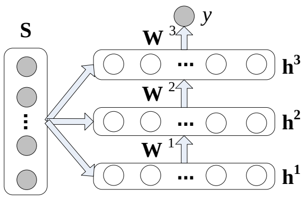


Fig. 5. The BP network for fine tuning and estimating visibility.

Fig. 5, i.e.

$$\begin{aligned}\tilde{h}_j^{l+1} &= p(h_j^{l+1}=1|\mathbf{h}\backslash h_j^{l+1},\mathbf{x}) = p(h_j^{l+1}=1|\mathbf{h}^l,\mathbf{x})\\ &= \sigma(\mathbf{h}^{l^{\mathrm{T}}}\mathbf{w}^l_{*,j} + c_j^{l+1} + \mathbf{g}_j^{l+1^T}\mathbf{s}_j^{l+1}),\\ \tilde{h}_j^1 &= \sigma(c_j^1 + \mathbf{g}_j^{1^T}\mathbf{s}_j^1).\end{aligned} \tag{15}$$

In order to reduce the bias of training data and regularize the training process, we enforce the visibility correlation parameter $\mathbf{W}^{l,0}$ in (9) to be non-negative. Therefore, our training process has used the prior knowledge that negative correlation among the visibility of parts is unreasonable, e.g. the invisible left-leg shall not indicate the visible two-legs. Furthermore, the element $w^l_{i,j}$ of $\mathbf{W}^l$ in (8) is set to zero if there is no connection between units h^l_i and h^{l+1}_j in Fig. 4. Taking the parts in Fig. 1 as an example, the visibility of the part left-leg is considered as not correlated with the visibility of the part head-shoulder. For the extended model in Section IV-B3, on the other hand, the visual dissimilarity among different parts is considered as an important visual cue and there is not constraint for the elements in $\tilde{\mathbf{W}}^l$ in (9)(*i.e.*, we consider the visual dissimilarity between any two parts). In this way, we keep the most important correlation parameters based on prior knowledge.

V. Experimental Results

The proposed framework is evaluated on four datasets: Caltech [20], ETHZ [27] and Daimler [22] datasets are publicly available; the CUHK occlusion dataset is constructed by us[2]. The INRIA training dataset in [12] is directly used to train our approach if not specified. Occlusion information is not required during training. Once the model is learned from this training set, it is fixed and tested on the four datasets mentioned above. Our deep model is to learn the visibility correlations among different parts, which is feasible even though the INRIA training set does not have many occluded pedestrian samples. It shares similar spirit with some data reconstruction problems solved with deep models [76], [92]. The data model is learned from positive samples without being corrupted. If any test sample is corrupted, its missing values can be reconstructed with the learned deep models in [76], [92]. In pedestrian detection, the performance might get improved if the training set includes occluded positive samples. However it will also take the risk of introducing bias, since the distribution of occlusion configurations in the training set could be different than the test set. Our current experimental results show that only using INRIA training set without many occlusions leads to good performance on various test datasets.

In the experiment, we use the modified HOG in [29] as the feature for detection. HOG feature was proposed in [12] and modified in [29]. The parts at the bottom layer and the head-shoulder part at the middle layer compute HOG features at twice the spatial resolution relative to the features computed by the other parts. In our implementation, the deformable part-based model in [29] is used for learning part detectors and modeling the deformation among the 19 parts in Fig. 4. The parts are arranged in the star model with the full body part being the root. Since the detection scores obtained from our parts model are considered as the input of our deep model, the deep model keeps unchanged if other deformable part-based models and features are used.

The part detector follows the approach in [29] on using 10 intermediate scales for increasing the image size by 2 in both width and height. The non-maximum-suppression follows the approach in [20]. The occlusion handling is only applied to a reduced set of high scoring hypotheses obtained from our part model in order to save the computation required for the deep model.

The approaches HOG+SVM [12] and LatSVM-V2 [29] to be compared and our approach use the same features for part-based detection. They are also trained from the INRIA dataset. The evaluation criteria proposed in [20] is used. The labels and evaluation code provided by Dollár et al. online[3] is used for evaluating the Caltech dataset and the ETHZ dataset. As in [20], *log-average miss rate* is used to summarize the detector performance, computed by averaging miss rate at nine FPPI rates evenly spaced in log-space from 10^{-2} to 10^0.

A. Experimental Results on the CUHK Occlusion Dataset

Most existing pedestrian detection datasets are not specifically designed for evaluating occlusion handling. For example, although the Caltech training dataset contains 192k pedestrians and 128k images, it is from 30 frames per second video sequences, where many frames are very similar to each other. In order to save computation and avoid evaluating nearly-the-same images, existing literatures [3], [16], [17], [17], [18], [20], [73], [82], [97] report the results on the Caltech dataset using every 30th frame (starting with the 30th frame), i.e. 4250 images in the Caltech training dataset are used for evaluation. In these 4250 images, only 105 images contain occluded pedestrians. If such datasets are used for evaluation, it is not clear how much improvement comes from occlusion handling or other factors. In order to specifically compare pedestrian detection algorithms under occlusions, we construct the CUHK occlusion dataset that mainly include images with occluded pedestrians. All the 105 images containing occluded pedestrians in the 4250 Caltech training images and occluded images from ETHZ, TUD-Brussels, INRA and Caviar datasets have been included in the CUHK dataset. We also record 212 images from surveillance cameras. The composition of the dataset is shown in Table I. The dataset contains 3476 non-occluded pedestrians and 2373 occluded pedestrians. Images are strictly selected according to the following criteria.

1. Each image contains at least one occluded pedestrian.

[2] Available on www.ee.cuhk.edu.hk/~xgwang/CUHK_pedestrian.html

[3] Available on www.vision.caltech.edu/ImageDatasets/CaltechPedestrians/

TABLE I
THE COMPOSITION OF THE CUHK DATASET.

Dataset	Number of images selected
Caltech train [20]	105
INRIA test [12]	70
TUD-Brussels [102]	110
ETHZ [27]	211
Caviar [32]	355
Our	212

2. Datasets Caviar and ETHZ are video sequences with high frame rate, e.g. 25 frames per second for Caviar. In these datasets, the current frame may be very similar to the next frame. In our dataset, the frame rate is reduced to ensure variation among selected images.

3. The image shall not contain sitting humans, since it is potentially controversial whether they should be detected as pedestrian or not.

Each pedestrian is labeled with a bounding box and a tag indicating whether the pedestrian is occluded or not. Since a lot of occluded pedestrians in datasets like INRIA, ETHZ and TUD-Brussels are not considered as positive testing samples, the occluded pedestrians are relabeled in our dataset. Occluded pedestrians have been labeled in the Caltech dataset, their labels are unchanged in our dataset. Selected detection results of our approach on this dataset are shown in Fig. 7.

We evaluate the performance of our approach on occluded pedestrians and unoccluded pedestrians separately and compare with the part-based models [115], LatSVM-V2 [29], LatSVM-V5-VOC [37], and LatSVM-V5-Inria [37] in Fig. 6. Zhu et al. define part and their sub parts while LatSVM-V2 only defines a root and its parts. Our approach has similar performance with [115] and [29] on unoccluded pedestrians and achieved 9% improvement on occluded pedestrians compared with [115] and [29] (the smaller the miss rate in the y-axis the better). LatSVM-V5-VOC and LatSVM-V5-Inria are trained on VOC2007 and INRIA dataset separately. The model and code provided by the authors for LatSVM-V5-VOC and LatSVM-V5-Inria are directly used for evaluation. As shown in Fig. 6, the grammar model in [37] does not perform well on our dataset. To investigate the effectiveness of using the deep model to estimate the visibility of parts, we also test our part-based model that directly sums up detection score using (4) and exclude the deep model. It has comparable performance as [115] and [29] on occluded pedestrians. By including more information of the pairwise visual dissimilarity among parts, the extended model introduced in Section IV-B3, i.e. *Ours-D2*, is better than the model in Section IV-B2, i.e. *Ours-D1*.

In order to investigate various schemes for integrating the part detection scores, we conduct another set of experiments in Fig. 6(c)-(f). They all use our parts model and therefore have the same detection scores as input. *Our-P* in Fig. 6 is the weighted mean of part scores and the weights are trained by linear SVM. Fig. 6(c) and (d) show the results of estimating the visibility by thresholding the detection scores, i.e. part score s_i is ignored if $s_i < T_i$. Using the same T_i for all the parts is not optimal. Therefore, we assume that different parts have different threshold T_i and obtain T_i from training data. For each part, T_i is chosen such that certain percentage $\epsilon (= 0.1\%, 1\%, 5\%, 10\%, 20\%, 40\%, 50\%)$ of parts in the positive training samples are considered as not correctly detected by part detectors due to occlusion, abnormal deformation, appearance or illumination. We also learn one bias for each part using SVM, which is denoted by Part-SVM-Bias. Experimental result shows that Part-SVM-Bias does not have improvement compared with the thresholding having $\epsilon = 5\%$. The approach in [21] defines rule for estimating visibility of parts and integrating detection scores. We use the same rules proposed in [21] to integrate our part scores. As shown in Fig. 6 (c) and (d), the rule based integration does not work well on our parts model although it has reported satisfactory results on the parts model in [21]. This may be due to the fact that we use different features and different parts model from [21]. We cannot exactly obtain the results in [21] on our dataset because its implemenation is not available. The *DBN* in Fig. 6 arranges all part detection scores as the bottom observed layer and 3 layers of hidden units on top of the observed layer as shown in Fig. 3(c). The approach in [40] is then used for training parameters and classifying whether an input window is a pedestrian or not. It is observed that directly applying DBN to parts detection scores does not solve the occlusion problem effectively. Fig. 6(e) and (f) show the results of taking $k = 1, 2, 4, 8, 10, 15, 18$ maximum part scores for computing the weighted mean. The experimental results show that all the schemes discussed above perform worse than our deep model (represented by $Ours - D2$).

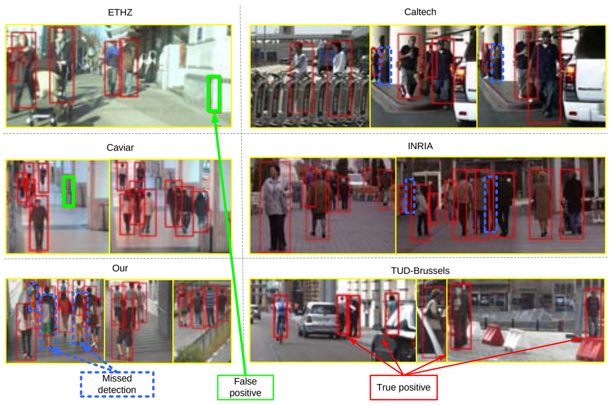
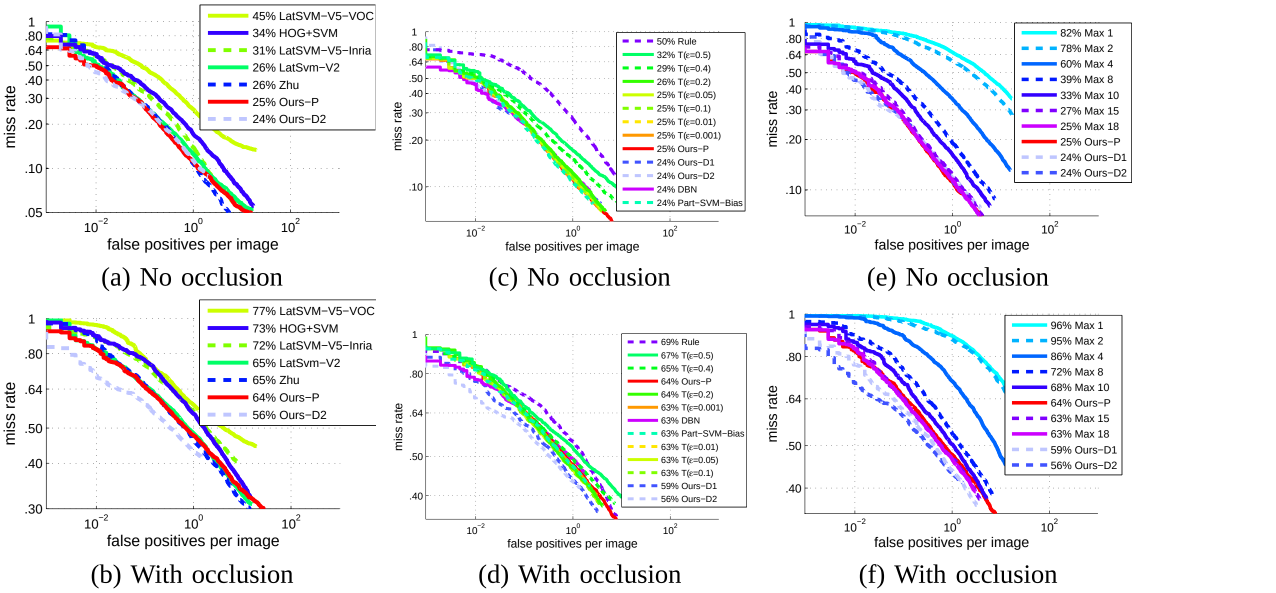


Fig. 7. Selected detection results using our framework on the CUHK occlusion dataset. The sources of images are given. All results are obtained by using the same threshold. Blue rectangles in dashed lines are missed detections, green rectangles in solid lines are false positive windows, and red rectangles in solid lines are true positive detections.

In another experiment, we investigate the robustness of the model when the training dataset is under different levels of disturbances. The goal is systematically study if the occlusion states in the training set has bias (e.g. left leg is more frequently occluded than other parts), whether the performance of the trained deep model will be deteriorated on test samples occluded in different ways. Results show that it is worse than a properly trained deep model but still slightly better than directly summing up the part detection scores. Fig. 9 shows the experimental results. In this experiment, we distort the INRIA training dataset and obtain 4 distorted training datasets $Dstrt1 - Dstrt4$. The distorted images are only used for training the parameters of the occlusion model but



(a) No occlusion

(c) No occlusion

(e) No occlusion

(b) With occlusion

(d) With occlusion

(f) With occlusion

Fig. 6. Experimental comparisons of different part-based models ((a)-(b)) and different schemes of integrating part detection scores ((c) - (f)) on the CUHK dataset for pedestrians *without occlusions* (upper row) and *with occlusions* (bottom row). *Zhu* denotes results of using the parts model proposed by Zhu et al. in [115]. LatSVM-V5-VOC and LatSVM-V5-Inria denote the approach in [37] trained on VOC2007 and INRIA dataset separately. *Ours-P* denotes results of using our parts model in Fig. 4 and directly summing up detection score however without the deep model. In this case, it is equivalent to computing the weighted mean of part scores. *Ours-D1* and *Ours-D2* denote the results of using our parts model and the deep model introduced in SectionIV-B. Ours-D1 denotes the deep model in Section IV-B2 and Ours-D2 denotes the extended model in Section IV-B3. *DBN* denotes the results of replacing our deep model by DBN. *Rule* denotes the results of using the rule in [21] for integrating our part scores. *T(ϵ=ϵ_0)* denotes the results of estimating visibility by hard-thresholding. T_i is learned from the training data such that ϵ percentage of parts in the positive training samples are considered as not correctly detected by part detectors due to occlusion, abnormal deformation, appearance or illumination. *Max* k denotes taking the k maximum part scores for computing the weighted mean.

not the part model. The negative training samples are kept unchanged. For a dataset, say $Dstrt1$, all positive training samples have the same region replaced by randomly selected negative patches. In this way, the detection scores $\mathbf{s}$ related to this region are distorted for all positive samples. Fig. 8 shows the examples of distorted positive examples. Dataset $Dstrt1$ has the left leg and left torso replaced by negative patches, dataset $Dstrt2$ has the two legs replaced and dataset $Dstrt3$ has the torso and legs replaced. All positive pedestrians in $Dstrt3$ have about $3/4$ region distorted. $Dstrt4$ has equal distribution of occlusion for the six regions (left/right head-shoulder, left/right torso, left/right leg). As shown by the experimental results in Fig. 8, the distortion does influence the detection performance of the deep model. All compared approaches have similar performance as that when pedestrians are not occluded. The performance on testing data degrades for occluded pedestrians when distortion exists in the positive training samples, compared with $Ours-D2$, which is properly trained. When the distortion is the largest, i.e. dataset $Dstrt3$, the detection performance is the worst. Even if about $3/4$ region of the pedestrian is distorted for all positive samples in $Dstrt3$, the model still learns reasonable parameters and outperforms the case when the part detection scores are directly summed up without the deep model, i.e. Ours-P. The deep model aims at learning the visibility relationship among parts. The worst bias caused by the disturbed region, e.g. left-leg for the dataset $Dstrt1$, is to have negative relationship learned among parts, e.g. between left-leg and two-legs for the dataset $Dstrt1$. With the non-negative enforcement on the elements in $\mathbf{W}^{l,0}$, negative relationship is impossible. Therefore, the relationship learned for the disturbed region is zero at the worst case, in which the deep model degenerates into using no relationship and directly using part score for detection. Since the relationship among undistorted parts, e.g. the relationship between left-head-shoulder and head-shoulder, is still effectively learned, the deep model outperforms the case where no relationship is used.

Fig. 10 shows the experimental results on different implementations of the deep model. Compared with the implementation that restricts the weights among hidden nodes to be non-negative (Ours-D2), the implementation without this restriction increases the miss rate by 4% for pedestrians without occlusion and 3% for pedestrians with occlusion (Ours-D2-NW). Compared with the implementation (Ours-D2) that uses 19 hidden nodes for (3), the implementation that uses the top 6 hidden nodes in Fig. 4 (Ours-D2-6h) for (3) increases the miss rate by 3% for pedestrians without occlusion and 4% for pedestrians with occlusion. Compared with the implementation that restricts certain connections among hidden nodes to be zero (Ours-D2), the implementation without this restriction increases the miss rate by 2% for pedestrians without occlusion and 2% for pedestrians without zero restriction on connections (Ours-AllCon). The implementation that uses only BP for training (Ours-D2-BP) the model (without unsupervised RMB pre-training) increases the miss rate by 6% for pedestrians with occlusion. The number of rounds used in Ours-D2-BP for BP is equal to the number of rounds for RBM used in Ours-D2 plus the number of rounds for BP used in Ours-D2. The implementation that uses only BP for training (Ours-D2-BP) the model increases the miss rate by 6% for pedestrians with occlusion. The number of rounds used in Ours-D2-BP for BP is equal to the number of rounds for RBM plus the number of

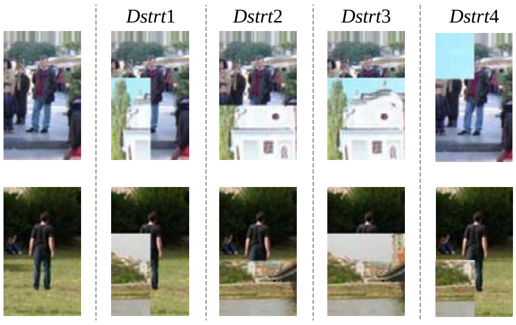
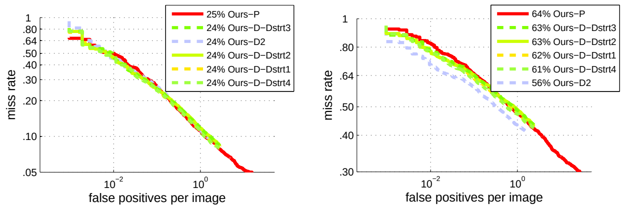


Fig. 8. Selected original positive samples (1st column) and distorted positive examples with disturbance in dataset $Dstrt1$, $Dstrt2$, $Dstrt4$, and $Dstrt3$ (2nd, 3rd, 4th column, and 5th column). The same region is distorted in the same dataset. The same positive sample is distorted by the same negative sample. 2416 negative samples are randomly selected for replacing the corresponding regions of the 2416 positive samples.

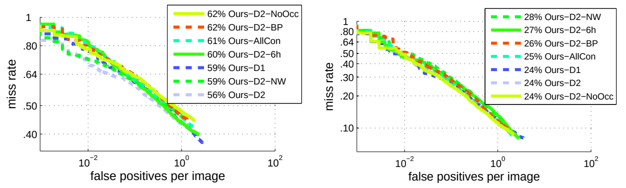


(a) No occlusion (b) With occlusion

Fig. 9. Experimental results on the CUHK occlusion dataset for the deep model when the positive training data is distorted. Ours-D2 denotes the case when there is no distortion. *Ours-D-Dstrt1* denotes results for the deep model trained on dataset $Dstrt1$. Similarly for *Ours-D-Dstrt2*, *Ours-D-Dstrt3* and *Ours-D-Dstrt4*.

rounds for BP used by Ours-D2. The implementation Ours-D2-NoOcc that only linearly combines the color histogram dissimilarity terms without the deep model for occlusion handling increases the miss rate by 6% for pedestrians with occlusion. Experimental results for Ours-D2-BP and Ours-D2-NoOcc on the Caltech Testing dataset are shown in Fig. 14. Theoretically, the issue of whether unsupervised pre-training helps later supervised learning in the same network is controversial and far from being decided. The empirical results on the two datasets show that indeed the unsupervised stage contributes to performance.

Although this paper focuses on using the deep model for pedestrian detection, the proposed deep model is also applicable for estimating the visibility of parts. Fig. 12 shows the visibility estimation results obtained from the deep model. Fig. 12 shows the ROC curve in estimating the occlusion status, in which 1000 positive samples from the CUHK occlusion dataset are used for evaluation. The average precision for occlusion estimation is 81%.



(a) No occlusion (b) With occlusion

Fig. 10. Experimental results on the CUHK occlusion dataset for different implementations of the deep model. *Ours-D2* denotes the case when 19 hidden nodes are used for estimating detection label and the weights among hidden nodes to be non-negative. *Ours-D2-6h* denotes the case when 6 hidden nodes are used for estimating detection label. *Ours-D2-NW* denotes the case when the weights among hidden nodes are allowed to be negative. *Ours-AllCon* denotes the case when the weights among hidden nodes are not restricted to be zero.

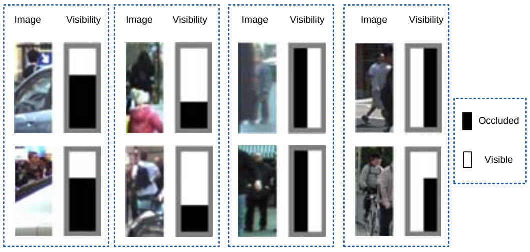


Fig. 11. Visibility estimated from the deep model. Black rectangle corresponds to invisible parts.

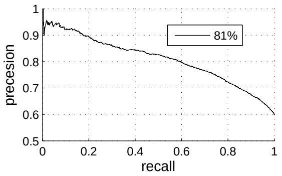


Fig. 12. ROC curve in estimating the occlusion status.

To investigate the execution time required by our model, we run the LatSVM-V2 and our parts model for 8 images with resolution 1280×960. The experiment is run for 3 times and the difference in total execution time is less than 1%. The average detection time required by our parts model is about 1.3 times of that required by LatSVM-V2 on a 3.3GHz CPU with multi-threading turned off. The most time consuming tasks, i.e. feature and SVM computation, for our parts model are implemented by the same c code as the LatSVM-V2 provided by Felzenszwalb etc. online [30]. Our parts model contains 25730 features and LatSVM-V2 contains 12741 features. The number of features mainly influence the time required for computing SVM. According to our experiment, although our parts model contains about 2 times the number of features of LatSVM-V2, the execution time required by our parts model for computing SVM is less than 1.4 times the time required by LatSVM-V2. This might be caused by the fact that both models compute SVM on the same feature window and take the same execution time caused by cache miss, which is a main factor that influences the time required for computing SVM on sliding windows. The time required by our deep model for estimating visibility using the deep model is less than 10% of the time required by our part-based detector. Since our deep model has only 20 hidden variables in all for 3 layers, training time for the deep model is also much less than that for the parts model.

B. Experimental Results on Caltech

The evaluated pedestrian detection approaches on the Caltech dataset are VJ [96], Shapelet [79], PoseInv [52], LatSVM-V1 [29], HikSVM [58], HOG+SVM [12], MultiFtr [101], HogLbp [98], Pls [80], FeatSynth [3], FtrMine [19], MultiFtr+CCS, MultiFtr+Motion [97], FPDW [17], ChnFtrs [18], CrossTalk [16], and MultiResC [73], MultiSDP [113], Franken [59], ACF+SDt [74], JointDeep [68], LDCF [63], LatSVM-V5-VOC [37], and LatSVM-V5-Inria [37].

In the first experiment, the Caltech training dataset is used as our testing set and the INRIA training dataset is used as our

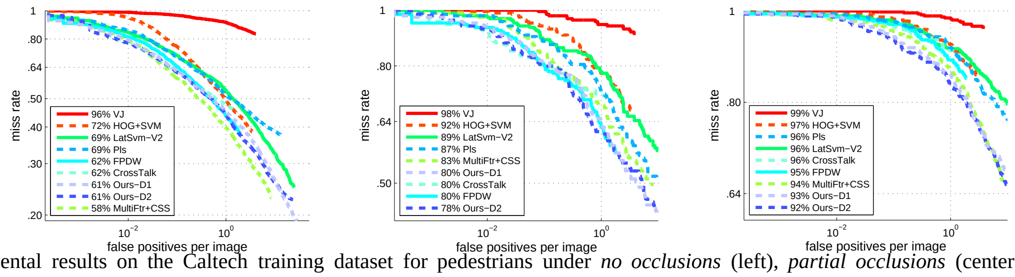


Fig. 13. Experimental results on the Caltech training dataset for pedestrians under *no occlusions* (left), *partial occlusions* (center) and *heavy occlusions* (right). The ratio of visible area is larger than 0.65 for *partial occlusions* and [0.2 0.65] for *heavy occlusions*. The log-average miss rate of our model is 60% for no occlusions, 79% for partial occlusions and 92% for heavy occlusions.

training set to be consistent with most compared approaches, e.g. [3], [19], [97]. In Fig. 13, we compare with 16 approaches under varying levels of occlusion. Compared with LatSVM-V2, our approach has 8%, 11% and 4% improvement on the log-average miss rate for pedestrians with no occlusions, partial occlusions and heavy occlusions respectively. Compared with the state-of-the-art approaches evaluated in [20] (excluding those using motions), our approach ranks as the third, the second and the first for pedestrians with no occlusions, partial occlusions and heavy occlusions respectively. The two approaches MultiFtr+CCS [97] and ChnFtrs [18], which performed better than ours in the cases of no occlusions and partial occlusions, both used a large number of extra features such as color self-similarity, local sums, histograms, Haar features and their various generalizations beside HOG. Only HOG+SVM, LatSVM-V2 and our approach used the HOG features to compute the detection score. With more features being included, the performance of our approach can be further improved.

In the second experiment, the Caltech training dataset is used as our training set and the Caltech testing dataset, is used as our testing set to be consistent with the approach MultiResC [73]. In this experiment, we evaluate the performance on the *reasonable* subset, which is the most popular portion of the datasets. It consists of pedestrians with more than 49 pixels in height, who are fully visible or partially occluded. The approach in [73] used the value $[bb_h - (a \cdot bb_y + b)]^2$ as the geometric constraint, where bb_h is the bounding box height, bb_y is the y-location of the lower edge of the bounding box, a and b are linear regression parameters learned from the ground truth bounding box of Caltech training dataset in [73]. This geometric constraint is also used by our approach to make a fair comparison with the approach in [73]. However, we obtain the linear regression parameters a and b from detection bounding boxes on the Caltech testing dataset in an unsupervised way, i.e. we need not the ground truth bounding box for learning a and b. As shown by Fig. 14, our approach has 4% average miss rate improvement compared with the MultiResC [73]. This geometric constraint is only used on the Caltech testing dataset but not used on other datasets, since [73] was not reported on other datasets. Our approach performs better than the grammar model in [37] on the ETHZ and Caltech Test dataset. Our approach outperforms MultiSDP and Franken on the Caltech Test dataset. Both MultiSDP and Franken have used more effective features than our approach. The results for our approach are only based on HOG feature. With HOG+CSS features, our approach has average miss rate 40%, which is 5% lower than MultiSDP and 9% lower than Franken.

Compared with the deformable model LatSVM-V2, our deep model reduces the miss rate from 63% to 44% on the Caltech testing dataset and from 51% to 46% on the ETHZ dataset. By including more information of the pair-wise visual dissimilarity among parts, the extended model introduced in Section IV-B3, i.e. Ours-D2, performs better than the model in Section IV-B2, i.e. Ours-D1.

C. *Experimental Results on ETHZ*

The experimental results on the ETHZ testing sequences are shown in Fig. 14. It is reported in [20] that LatSvm-V2 has the best performance among the 14 state-of-the-art approaches evaluated on the ETHZ dataset. It can be seen that our approach has 5% improvement over LatSVM-V2. The ETHZ dataset consists of 3 testing video sequences. Table II shows the miss rates at 1 FPPI for the 3 sequences. The results of ISF are obtained from [21]. The results of HOG+SVM and LatSvm-V2 are obtained from [20] using the results and evaluation code provided online. Our model performs better than the traditional deep learning approach [82] on both ETHZ and Caltech testing dataset. With better features, the recent approaches LDCF, Franken, and JointDeep performs better than our approach.

D. *Experimental Results on Daimler*

The experimental results on the Daimler benchmark testing data in [22] are shown in Fig. 15. Since the dataset is used for occluded pedestrian classification instead of detection, false positive versus detection rate is used for evaluation (the larger the detection rate in the y-axis the better). Since our focus is on detection for single images, we only use the image intensity for all evaluated algorithms. Compared with LatSVM-V2, our approach has similar performance on unoccluded pedestrian, and our approach achieves about 20% detection rate improvement for occluded pedestrian. LatSVM-V2, HOG+SVM and our approach in Fig. 15 are trained on INRIA for consistency with previous experimental results. Since all the results in [22] are trained on the Daimler training data and have different implementation of HOG feature from ours, we did not show the results in [22]. For example, the HOG+SVM trained on INRIA using the code in [29] have quite different result from the HOG+SVM trained on Daimler training data reported in [22].

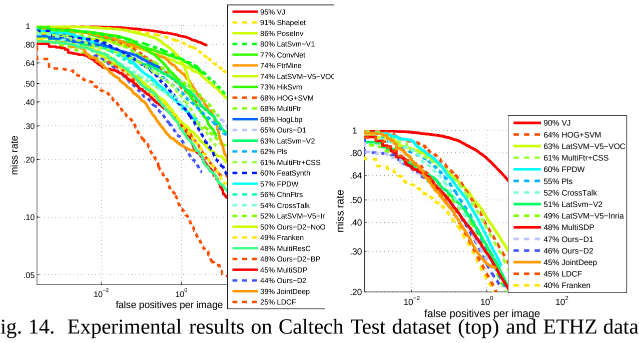


Fig. 14. Experimental results on Caltech Test dataset (top) and ETHZ dataset (bottom).

TABLE II
MISS RATE AT 1 FPPI FOR DIFFERENT APPROACHES. SEQ 1 (BAHNHOF) HAS 999 FRAMES, SEQ 2 (JELMOLI) HAS 450 FRAMES AND SEQ 3 (SUNNY DAY) HAS 354 FRAMES.

	Seq 1	Seq 2	Seq 3
ISF [21]	47%	38%	52%
HOG+SVM [12]	34%	44%	44%
LatSvm-V2 [29]	30%	34%	32%
Ours	23%	33%	26%

VI. DISCUSSION

In this paper, the star model is used to model the deformation correlation among parts, because the star model is widely used in pedestrian detection and many recent papers [29], [73] based on the star model achieved the state-of-the-art results on both the ETH and the Caltech dataset. If it is replaced with other part models like the tree model [31], [88], [115], loopy graph models [100] and complete graph models [7], our approach cannot be directly used in a straightforward way. However, it is still feasible after certain modification. Take the tree model as an example. The appearance score s_i^l and the deformation score d_i^l need to be treated separately in (9)-(12). The terms $a_i^l d_i^l h_i^{l+1} + b_i^l(1 - h_i^{l+1})$ related to d_i^l depend on the visibility h_i^{l+1} of the parent part. If the parent part is visible, i.e. $h_i^{l+1} = 1$, the penalty term is $a_i^l d_i^l$, which depends on the deformation score; otherwise, it is a constant b_i^l as the deformation score has become meaningless. Both a_i^l and b_i^l are parameters to be learned. This still leads to RBM, $\mathbf{W}^l$ in (12) becomes $\mathbf{W}^l = \mathbf{W}^{l,0} + \tilde{\mathbf{W}}^{l,1} \circ \tilde{\mathbf{S}}^l + \tilde{\mathbf{W}}^{l,2} \circ \tilde{\mathbf{D}}^l$. This is a future work. Our design of parts model is based on the knowledge about the constituents of human beings. Design and learning of new parts models that is optimized for human detection is a possible way of improving the detection result. Since our deep model takes the detection scores of parts as input, it is very flexible to incorporate with new features [64], [82] and new deformable part-based models.

The part model and deep model in this paper does not use annotations of occlusion for training. If occlusion labels are available, they can be used as supervision so that the estimated label should be close to the annotation labels, which can potentially improve the results. Further implementation on the use of occlusion labels is an interesting future work.

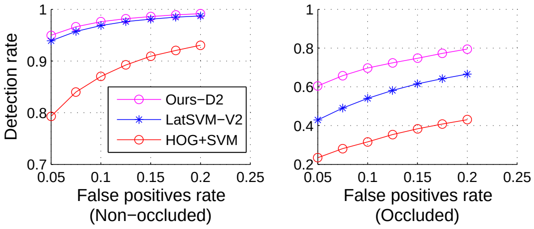


Fig. 15. Experimental results on Daimler occlusion dataset.

The detection score is assumed to be provided in order to be independent of detectors and features. However, interaction between the deep model and specific detector is a future work for improvement. For example, since features can be learned by a deep model, e.g. the one in [64], it is possible to incorporate the DBN into the learning of the part-based detector and estimating the visibility. It is also an interesting and open question of how to integrate the estimation of part locations into the deep model.

Although we only use single image pixel values for detection. The extended deep model in (9) has considered multiple sources of information and is naturally applicable for multiple cues like depth, motion and segmentation.

This paper estimates the detection label using the mean field approximation in (3) for faster speed. Investigation on the use of other methods for obtaining detection label from the visibility states of parts is a potential way of improving detection accuracy.

This paper aims at modeling occlusion at part level. However, modeling occlusion at pixel level is a promising direction for handling occlusion. For example, the masked RBM in [46] can be used for explicitly modeling occlusion boundaries in image patches by factoring the appearance of a patch region from its shape.

The main contribution of this paper is to learn the visibility relationship among parts using a hierarchical probabilistic model. Both directed model and undirected models can be used for learning this relationship. DBN is a combination of undirected graphical model at the top layer and directed graphical model at the other layers. Directed graphical models often lead to the explaining away problem, in which recovering the posterior $p(\mathbf{h}|\mathbf{x})$ is often computationally challenging and even intractable, especially when $\mathbf{h}$ is discrete [6]. The DBN style model is chosen because it is easy for inference and has fast training algorithm [40] that can find a fairly good set of parameters quickly.

Currently, evaluation on Caltech-train and ETHZ, including ours, are mostly based training on INRIA. This results in the problem of domain-shift, in which training the model on a specific dataset and testing on others. For instance, the INRIA pedestrian dataset contains relatively high resolution pedestrians, while Caltech and ETHZ can contains pedestrian instances at much lower resolution. There are two groups of approaches handling this problem. The first group learns different detectors for different resolutions [73], [108]. The second group takes this domain-shift into account and learns scene specific detectors [99], [111]. Since our approach does not take this domain shift into account, the combination of our approach and domain adaptive approach is applicable for further improving the performance on datasets like Caltech-train and ETHZ.

VII. Conclusion

This paper describes a probabilistic framework for pedestrian detection with occlusion handling. It effectively estimates the visibility of parts at multiple layers and learns their relationship with the proposed deep model. Since it takes the detection scores of parts as input, it is very flexible to incorporate with new features and other deformable part-based models. Through extensive experimental comparison on multiple datasets, various schemes of integrating part detectors are investigated. Our approach outperforms the state-of-the-arts especially on pedestrian data with occlusions.

Acknowledgment: This work is supported by the General Research Fund sponsored by the Research Grants Council of Hong Kong (project No. CUHK417110 and CUHK417011) and National Natural Science Foundation of China (project no. 61005057).

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: people detection and articulated pose estimation. In *CVPR*, 2009.
[2] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *ECCV*, 2012.
[3] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg. Part-based feature synthesis for human detection. In *ECCV*, 2010.
[4] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using hough transforms. In *CVPR*, 2010.
[5] Y. Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009.
[6] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Trans. PAMI*, 35(8):1798–1828, 2013.
[7] M. Bergtholdt, J. H. Kappes, S. Schmidt, and C. Schnorr. A study of parts-based object class detection using complete graphs. *IJCV*, 87(1-2):93–117, 2010.
[8] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *ECCV*, 2010.
[9] L. Bourdev and J. Malik. Poselets: body part detectors trained using 3D human pose annotations. In *ICCV*, 2009.
[10] Y. Chen, L. L. Zhu, and A. Yuille. Active mask hierarchies for object detection. In *ECCV*, pages 43–56. 2010.
[11] S. Dai, M. Yang, Y. Wu, and A. Katsaggelos. Detector ensemble. In *CVPR*, 2007.
[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.
[13] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006.
[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: a large-scale hierarchical image database. In *CVPR*, 2009.
[15] C. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. In *ECCV*, 2012.
[16] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *ECCV*, 2012.
[17] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *BMVC*, 2010.
[18] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009.
[19] P. Dollár, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *CVPR*, 2007.
[20] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: an evaluation of the state of the art. *IEEE Trans. PAMI*, 34(4):743 – 761, 2012.
[21] G. Duan, H. Ai, and S. Lao. A structural filter approach to human detection. In *ECCV*, 2010.
[22] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. In *CVPR*, 2010.
[23] M. Enzweiler and D. Gavrila. A mixed generative-discriminative framework for pedestrian classification. In *CVPR*, 2008.
[24] M. Enzweiler and D. Gavrila. Integrated pedestrian classification and orientation estimation. In *CVPR*, 2010.
[25] M. Enzweiler and D. M. Gavrila. Monocular pedestrian detection: survey and experiments. *IEEE Trans. PAMI*, 31(12):2179–2195, 2009.
[26] M. Enzweiler and D. M. Gavrila. A multilevel mixture-of-experts framework for pedestrian classification. *IEEE Trans. Image Process.*, 20(10):2967–2979, 2011.
[27] A. Ess, B. Leibe, and L. V. Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007.
[28] M. Everingham, L. V. Gool, C. K. I.Williams, J.Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.
[29] P. Felzenszwalb, R. B. Grishick, D.McAllister, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Trans. PAMI*, 32:1627–1645, 2010.
[30] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. www.cs.brown.edu/~pff/latent/, accessed in 2012.
[31] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61:55–79, 2005.
[32] R. Fisher. homepages.inf.ed.ac.uk/rbf/caviar/, accessed in 2012.
[33] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *CVPR*, 2009.
[34] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR*, 2011.
[35] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey on pedestrian detection for advanced driver assistance systems. *IEEE Trans. PAMI*, 32(7):1239–1258, 2010.
[36] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.
[37] R. Girshick, P. Felzenszwalb, and D. McAllester. Object detection with grammar models. In *NIPS*, 2011.
[38] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*. 2014.
[39] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:1771–1800, 2002.
[40] G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.
[41] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504 – 507, July 2006.
[42] A. Hyvärinen. Estimation of non-normalized statistical models using score matching. *J. of Machine Learning Research*, 6:695–709, 2005.
[43] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *CVPR*, 2009.
[44] S. V. Lab. http://www.image-net.org/challenges/lsvrc/2011/, accessed in 2012.
[45] C. Lampert, M. Blaschko, and T. Hofmann. Beyond sliding windows: object localization by efficient subwindow search. In *CVPR*, 2008.
[46] N. Le Roux, N. Heess, J. Shotton, and J. Winn. Learning a generative model of images by factoring appearance and shape. *Neural Computation*, 23(3):593–650, 2011.
[47] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *ICML*, 2009.
[48] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV Workshop on Stat. Learn. in Comp. Vis*, 2004.
[49] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR*, 2005.
[50] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.
[51] Y. Lin, T. Liu, and C. Fuh. Fast object detection with occlusion. In *ECCV*, 2004.
[52] Z. Lin and L. Davis. A pose-invariant descriptor for human detection and segmentation. In *ECCV*, 2008.
[53] Z. Lin, L. S. Davis, D. Doermann, and D. DeMenthon. Hierarchical part-template matching for human detection and segmentation. In *ICCV*, 2007.
[54] P. Luo, Y. Tian, X. Wang, and X. Tang. Switchable deep network for pedestrian detection. In *CVPR*, 2014.
[55] P. Luo, X. Wang, and X. Tang. Hierarchical face parsing via deep learning. In *CVPR*, 2012.
[56] P. Luo, X. Wang, and X. Tang. A deep sum-product architecture for robust facial attributes analysis. In *ICCV*, 2013.
[57] P. Luo, X. Wang, and X. Tang. Pedestrian parsing via deep decompositional neural network. In *ICCV*, 2013.
[58] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, 2008.
[59] M. Mathias, R. Benenson, R. Timofte, and L. Van Gool. Handling occlusions with franken-classifiers. In *CVPR*, 2013.
[60] C. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*, 2004.
[61] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple object class detection with a generative model. In *CVPR*, 2006.
[62] S. Munder and D. M. Gavrila. An experimental study on pedestrian

classification. *IEEE Trans. PAMI*, 28(11):1863–1868, 2006.

[63] W. Nam, P. Dollár, and J. H. Han. Local decorrelation for improved pedestrian detection. In *Advances in Neural Information Processing Systems*, pages 424–432, 2014.

[64] M. Norouzi, M. Ranjbar, and G. Mori. Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning. In *CVPR*, 2009.

[65] W. Ouyang, X. Chu, and X. Wang. Multi-source deep learning for human pose estimation. In *CVPR*, pages 2337–2344. IEEE, 2014.

[66] W. Ouyang, P. Luo, X. Zeng, S. Qiu, Y. Tian, H. Li, S. Yang, Z. Wang, Y. Xiong, C. Qian, et al. Deepid-net: multi-stage and deformable deep convolutional neural networks for object detection. *arXiv preprint arXiv:1409.3505*, 2014.

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

[68] W. Ouyang and X. Wang. Joint deep learning for pedestrian detection. In *ICCV*, 2013.

[69] W. Ouyang and X. Wang. Single-pedestrian detection aided by multi-pedestrian detection. In *CVPR*, 2013.

[70] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang, Z. Wang, C.-C. Loy, and X. Tang. Deepid-net: Deformable deep convolutional neural networks for object detection. In *CVPR*, 2015.

[71] W. Ouyang, X. Zeng, and X. Wang. Modeling mutual visibility relationship in pedestrian detection. In *CVPR*, 2013.

[72] W. Ouyang, X. Zeng, and X. Wang. Single-pedestrian detection aided by 2-pedestrian detection. *IEEE Trans. PAMI*, 2015.

[73] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. In *ECCV*, 2010.

[74] D. Park, C. L. Zitnick, D. Ramanan, and P. Dollár. Exploring weak stabilization for motion feature extraction. In *CVPR*, 2013.

[75] M. Pedersoli, J. Gonzalez, A. D. Bagdanov, and J. J. Villanueva. Recursive coarse-to-fine localization for fast object detection. In *ECCV*, 2010.

[76] H. Poon and P. Domingos. Sum-product networks: A new deep architecture. In *UAI*, 2011.

[77] F. Porikli. Integral histogram: a fast way to extract histograms in cartesian spaces. In *CVPR*, 2005.

[78] M. Ranzato, J. Susskind, V. Mnih, and G. Hinton. On deep generative models with applications to recognition. In *CVPR*, 2011.

[79] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *CVPR*, 2007.

[80] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis. Human detection using partial least squares analysis. In *ICCV*, 2009.

[81] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[82] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. Lecun. Pedestrian detection with unsupervised and multi-stage feature learning. In *CVPR*, 2013.

[83] V. D. Shet, J. Neumann, V. Ramesh, and L. S. Davis. Bilattice-based logical reasoning for human detection. In *CVPR*, 2007.

[84] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR*, 2014.

[85] A. Smeulders, T. Gevers, N. Sebe, and C. Snoek. Segmentation as selective search for object recognition. In *ICCV*, 2011.

[86] P. Smolensky. *Information processing in dynamical systems: foundations of harmony theory*. MIT Press, Cambridge, MA, USA, 1986.

[87] J. Sohl-Dickstein, P. Battaglino, and M. R. DeWeese. Minimum probability flow learning. In *ICML*, 2011.

[88] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, 2011.

[89] Y. Sun, X. Wang, and X. Tang. Deep convolutional network cascade for facial point detection. In *CVPR*, 2013.

[90] Y. Sun, X. Wang, and X. Tang. Hybrid deep learning for computing face similarities. In *ICCV*, 2013.

[91] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, 2014.

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

[93] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *IEEE Trans. PAMI*, 30(10):1713–1727, Oct. 2008.

[94] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *ICCV*, 2009.

[95] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[96] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *IJCV*, 63(2):153–161, 2005.

[97] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *CVPR*, 2010.

[98] X. Wang, X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *CVPR*, 2009.

[99] X. Wang, M. Wang, and W. Li. Scene-specific pedestrian detection for static video surveillance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(2):361–374, 2014.

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

[101] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM*, 2008.

[102] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, 2009.

[103] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, 2005.

[104] B. Wu and R. Nevatia. Cluster boosted tree classifier for multiview, multi-pose object detection. In *ICCV*, 2007.

[105] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *IJCV*, 75(2):247–266, 2007.

[106] B. Wu and R. Nevatia. Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. *IJCV*, 82(2):185–204, 2009.

[107] T. Wu and S. Zhu. A numeric study of the bottom-up and top-down inference processes in and-or graphs. *IJCV*, 93(2):226–252, Jun. 2011.

[108] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *CVPR*, 2013.

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

[110] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. *arXiv preprint arXiv:1311.2901*, 2013.

[111] X. Zeng, W. Ouyang, M. Wang, and X. Wang. Deep learning of scene-specific classifier for pedestrian detection. In *ECCV*, pages 472–487. 2014.

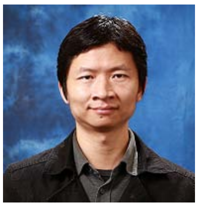
[112] X. Zeng, W. Ouyang, and X. Wang. Multi-stage contextual deep learning for pedestrian detection. In *ICCV*, 2013.

[113] X. Zeng, W. Ouyang, and X. Wang. Multi-stage contextual deep learning for pedestrian detection. In *ICCV*, 2013.

[114] R. Zhao, W. Ouyang, H. Li, and X. Wang. Saliency detection by multi-context deep learning. In *CVPR*, 2015.

[115] L. Zhu, Y. Chen, A. Yuille, and W. Freeman. Latent hierarchical structural learning for object detection. In *CVPR*, 2010.

[116] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, 2006.



Wanli Ouyang received the B.S. degree in computer science from Xiangtan University, Hunan, China, in 2003. He received the M.S. degree in computer science from the College of Computer Science and Technology, Beijing University of Technology, Beijing, China. He received the PhD degree in the Department of Electronic Engineering, The Chinese University of Hong Kong, where he is now a Research Assistant Professor. His research interests include image processing, computer vision and pattern recognition.

Xingyu Zeng received the B.S. degree in Electronic Engineering and Information Science from University of Science and Technology of China in 2011. He is currently a PHD student in the Department of Electronic Engineering at the Chinese University of Hong Kong. His research interests include computer vision and deep learning.

Xiaogang Wang received the BS degree from the University of Science and Technology of China in 2001, the MS degree from the Chinese University of Hong Kong in 2003, and the PhD degree from the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology in 2009. He is currently an Assistant Professor in the Department of Electronic Engineering at The Chinese University of Hong Kong. His research interests include computer vision and machine learning.