Solving Occlusion Problem in Pedestrian Detection by Constructing Discriminative Part Layers

Cong Cao¹, Yu Wang¹, Jien Kato¹, Guanwen Zhang², and Kenji Mase¹

¹Graduate School of Information Science, Nagoya University, Japan
²School of Electronics and Information, Northwestern Polytechnical University, China

cao@cmc.ss.is.nagoya-u.ac.jp, ywang@mv.ss.is.nagoya-u.ac.jp, jien@is.nagoya-u.ac.jp,
guanwen.zh@nwpu.edu.cn, mase@nagoya-u.jp

Abstract

Occlusion handling is one of the most challenging issues for pedestrian detection, and no satisfactory achievement has been found in this issue yet. Using human body parts has been considered as a reasonable way to overcome such an issue. In this paper, we propose a brand new approach based on the fusion of Mid-level body part mining and Convolutional Neural Network (CNN) to solve this problem, named DP-CNN (Discriminative Parts CNN).

Two main discussions are included in this paper. First, we take an exhaustive analysis on how to mine useful body parts that contribute to pedestrian detection. Multiple ingredients (e.g. feature representation, pedestrian attributes) are analyzed through a wide range of experiments. Second, we convert the part detectors to the middle layer of CNN and re-train the model to get a better adaption of the dataset. Compare to existing approaches based on fine-tuning CNN models, our method is not only robust to occlusion handling but also has a smaller computational cost.

1. Introduction

Pedestrian detection is widely used in the applications such as monitoring system and automatic drive. Although it has been exhaustively studied over the past decade[7, 10, 9, 27, 30], the occlusion situation remains a very challenging problem. In order to deal with this problem, one convincing method is to utilize the parts based methods for the visible parts information, and furthermore to estimate the pedestrian position.

Many part-based pedestrian detection methods have been proposed in recent years, most of those works focused on the mining method of body parts. In Bourdev et al.’s work [4], a Poselet method is proposed to implement the human detection based on the poses of human body parts under different viewpoints. However, this method needs hand-craft parts joint annotations which not insist in most of the pedestrian detection datasets. Tian et al. proposed Deepparts [26] which construct the part pool simply on the relative position of the pedestrian bounding box and train strong detectors for each part. Although it does not need additional annotations for training, because the relevance of patches in the part pool is weak, they have to consume large computational cost training CNN models to achieve a good performance. Like [26], many part-based methods have to make a compromise on computational cost when training part detectors because of those poorly mined parts.
More advanced than those methods, our DP-CNN employs a complex mining method that can find discriminative parts clusters without any extra annotations. Moreover, because the high-quality part clusters which demonstrate remarkable visual similarities, we can easily construct robust part detectors with much less training time.

We also notice that most of the methods do not pay much attention to the combining of part detectors results, including the approaches mentioned above. Many approaches use a linear SVM brutally that may not exert the potential of the part detectors. Therefore, we propose to merge part detectors within a simple CNN model, which makes full use of the detectors’ potential and achieves an astonishing optimization effect.

DP-CNN has 2 main contributions: (1) We propose a high-quality body parts mining method with CNN convolutional layer features and pedestrian subclasses. The mined part clusters own incredible discriminative and representative characteristic that helps construct powerful pedestrian detectors. (2) We propose a fusion method to combine the part detectors. We reform the part detectors to the convolutional layer of the CNN and optimize the whole pipeline by fine-tuning the CNN. In experiments, it shows the astonishing effectiveness of optimization and robustness of occlusion handling.

1.1. Related work

We introduce the related work in 3 aspects.

**CNN in pedestrian detection.** In recent years, due to the dazzling performance of CNN (Convolution Neural Network), the state-of-the-art standard line of pedestrian detection has been improved continuously. For instance, Hosang et al. [15] showed that both small and large networks can reach good performance by carefully exploring the design space and the critical implementation choices. Zhang et al. [31] provided a detailed analysis of a state-of-the-art pedestrian detector, and use the insight to construct an R-CNN-like[13] pipeline to improve baselines performance. Tian et al. [25] jointly optimize pedestrian detection with semantic tasks, including pedestrian attributes and scene attributes. They designed a single CNN model to fit the task and achieve state-of-the-art performance.

Including the methods mentioned above, most of the CNN methods are based on the R-CNN pipeline, which utilizes the preexisting detector to provide a large number of pedestrian proposals, then focus on constructing the CNN model to implement classification due to its impressive capability of feature representation. In this paper, we utilize the similar pipeline that absorbed in classification phase with CNN models.

**Mid-level visual element mining.** Mid-level visual element mining aims to discover clusters of the representative and discriminative image patches. It can be used for image classification[24, 8, 16] according to former works. Recently, Li et al.[18] utilize the CNN full-connected layer feature as the input of Apriori mining algorithm[2], then used mined clusters to construct detectors and encoding target images. Their method achieved state-of-the-art performances both on scene classification and object classification tasks.

Their research gave us a lot of inspiration about how to mine discriminative body parts from the pedestrian training data. Considering the obvious differences between pedestrian detection and object classification task, which our target is much small-sized and indistinguishable. We employ more local-adapted CNN conv layer features and design attribute-based pedestrian subclasses to mine high-quality body part clusters.

**Pedestrian detection based on body-parts.** Recent researches have showed that using the body-part detectors[4, 5, 6, 26] helps improve the pedestrian detection performance. The definition of body-part, which related with its configuration method, is the key to combining a more powerful pedestrian detector. Bourdev et al. proposed Poselets[4], body parts in different viewpoint and poses, which depended on the 3D joint keypoint annotation in the dataset. However, its dependency of annotation also leads to its inapplicability in other datasets. Recently, Tian et al. propose Deepparts[26], which construct part pool based on relative locations and train CNN models for each part pool. Their method achieved state-of-the-art pedestrian detection performance as well as the occlusion handling. However, because their easy mining method leads to the poor quality(sometimes irrelevant parts in the same part cluster), the computational cost of learning part detectors becomes very large.

Moreover, we found that most of the part-based method[4, 5, 6, 26] just focus on the construction of part detectors but ignore the combination method of part detection scores. For example, [26] and [5] both use a linear SVM to combine part detection scores to pedestrian score. For those methods, pedestrian detection performance may be improved with a better combination of part detectors.

Compare with the methods above, firstly, DP-CNN can mine high-quality body parts without any additional annotations, which makes the process of training robust part detectors easier and faster. Secondly, we propose a fusion method to transform learned part detectors to CNN middle layers, which proved to be an astonishing optimization method on combining part detection to pedestrian detection in our experiments.

2. Basic idea

In this paper, we focus on 2 key processes of constructing the part-based pedestrian detection approach: body-parts mining and part-detector combining in Sec.4. First, we pro-
pose a body parts mining method which can discover discriminative part clusters that can train the part detectors easily. Second, we transform the learned part detectors to the CNN middle layer and train the CNN model entirely, and it shows an astonishing optimization to improve the pedestrian detection performance.

2.1. Constructing part detectors

Overall, we mine body parts from pedestrian and background images using CNN-based association rule mining. The mining process is similar with MDPM[18], which mined mid-level visual elements for scene classification. Because our task is much tougher than theirs, which the body parts of pedestrians is small-sized and indistinguishable. We take the following 2 methods to ensure the mining quality:

1) We extract the convolution layer of the CNN model as the feature of the image patches, showed in Fig.1. Compare to the traditionally fully-connected layer representation method, it is more efficient and suitable to small patches.

2) We separate the pedestrian training images to several subclasses according to their pre-known attributes. This method improves the mining quality (visual consistency of the cluster) and furthermore helps to build the robust part detectors.

The details are discussed in Sec.3.

2.2. Combining part detectors

Many part-based pedestrian detection methods just focus the construct of part detector but coarsely combine the part detection results. We propose a brand new method which transforms the part detector to CNN middle layers and train the CNN model entirely to achieve the best combination consequent.

Thanks to the fact that we use CNN conv layer as image’s feature in the mining phase, we can stuff part detectors into the CNN model seamlessly and add fc layers behind as the classifier. The re-training not only construct a classifier to combine the part detection results, but also optimize the feature extraction and part detection phase.

The details are discussed in Sec.4.

3. Constructing body parts detectors

In this section, we explain how we constructing body parts detectors in the Caltech-usa[12, 11] dataset. We get much inspiration from MDPM[18], which first imported the CNN feature to mining mid-level visual elements. To adapt the tough pedestrian detection task, we use the CNN conv layer as feature representation and add the multi-attributes mining to take advantage of the pedestrian dataset. The whole pipeline can be summarized to 3 steps. (1) Extract CNN features of each image patch. (2) Mining visual element clusters in accordance with extracted features. (3) Construct detectors for body parts clusters.

3.1. Extract CNN features of image patches

In recent years, many authors realize that pre-trained CNN fc layer feature can represent the image much precisely than the traditional hand-craft features. For example, in [18], authors utilized the fc6 layer of per-trained AlexNet[17] to represents the feature of patches sampled from the original image. As for our task, which most of the pedestrian bounding boxes are under the average height of 80 pixels, the fc layer seems to be unreasonable. Because, in this situation, resize a small part of the pedestrian bounding box(e.g. 32×32) to feed the input of the CNN model(e.g. 227×227 of AlexNet) is either overqualified nor non-efficiency. Therefore, we propose to extract the last convolution layer of the whole image and use every N-dimension pixel of it to represent the corresponding image patches.

Sec.5.1 shows the experiments that confirm the conv layer feature can represent the object in the image as well as the fc layer.

3.2. Body parts mining

In this subsection, we give details of how to mine body parts on the base of the association rule mining[1], which is originally motivated by market basket analysis. This algorithm aims to discover a collection of if-then rules from the target data named transactions. In this case, the rule means “If the specified dimensions of CNN feature are active, then it is a pattern of pedestrian”.

3.2.1 Association rule mining

First we slightly introduce the fundamental of association rule mining[1]. Let $I = \{i_1, i_2, ..., i_n\}$ a set of $n$ items. Let $D = \{t_1, t_2, ..., t_m\}$ a set of $m$ transactions called database. Each transaction $t \in D$ contains a subsets of the items in $I$. The rule is defined as $\{X \rightarrow Y\}$, where $X, Y \subseteq I$. The support value of $X$ reflects this quantity:

$$supp(X) = \frac{|\{t|t \in D, X \subseteq t\}|}{m},$$

where $|\cdot|$ measures cardinality. $supp(X)$ shows how frequently the itemset $X$ appears in the database $D$.

The confidence value of the rule $\{X \rightarrow Y\}$ reflects this quantity:

$$conf(X \rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

This value shows how often the rule $\{X \rightarrow Y\}$ has been found to be true.
3.2.2 Pattern mining on pedestrians

In this paper, we consider every single image patch as a transaction \( t \). The 512-dimension Conv feature is 512 independent items. For each transaction, we just keep the top \( n \) items which have the largest activation in Conv feature and add the label of the class as another item. For example, if a patch that taken from the pedestrian image with the 1st, 10th, 512th dimension as its largest activation, the transaction will be \( \{1, 10, 512, \text{pedestrian}\} \). We use the Apriori algorithm [2] to find a set of patterns \( P \) that satisfy the following 2 inequalities:

\[
\text{supp}(P) > \text{supp}_{\text{min}}, \\
\text{conf}(P \rightarrow \text{pedestrian}) > \text{conf}_{\text{min}},
\]

\( \text{supp}_{\text{min}} \) and \( \text{conf}_{\text{min}} \) are handcraft thresholds of support value and confidence, which measure the representativeness and discriminativeness of rules we want to mine.

3.2.3 Mining Parts from attribute subclasses

By observing the visual elements of the clusters from pedestrian. We notice that different body parts (e.g. arms and legs) with similar visual elements sometimes are mined in the same cluster. It reminded us if we separate the data according to their per-known attributes, we can get more precise parts with human cognitive information. We assume that those additional part clusters may be the supplement of original clusters. Therefore, we designed following 2 simple subsets for mining extra part clusters:

- 1. \( \text{UpMidDown} \): Construct subset based on relative position. We cut the pedestrian bounding box to Up, Mid, Down 3 parts to reconstruct the training data. The classes became \( \{\text{Up, Mid, Down, Background}\} \).

- 2. \( \text{UpMidDown+Scales} \): Construct subset based on both the relative position and scales. We separate each Up-MidDown subclass to small scales \((\text{Height} < 80)\) and large scales \((\text{Height} > 80)\). The classes became \( \{\text{Up, small, Up−large, Mid−small, Mid−large, Down−small, Down−large, Background}\} \).

By the way, cause we think the discriminativeness between subclasses, which all belong to pedestrian, is important for the task. We just execute the mining algorithm between the specific class and background to find the rule \( \{X \rightarrow \text{attribute}\} \), attribute can be replaced with the subclasses we defined above. The support value here becomes:

\[
\text{supp}(X) = \frac{|\{t \in D', X \subseteq t\}|}{|D'|},
\]

where database \( D' = \{t | t \in D, \text{attribute} \subseteq t\} \cup \{t | t \in D, \text{background} \subseteq t\} \). Moreover, we do not mine rules from the background cause the large data quantity and diversity make the computational cost in calculable.

3.3. Train part detectors

According to the rules we fund in Sec 3.2, first, we retrieve the corresponding images patches (512-dimension Conv layer features) to construct visual elements clusters. Then use the patches in the clusters to train related part detectors. In this step, we use Linear Discriminant Analysis (LDA) [14] to train our detectors. We also use the merging algorithm which is proposed in [18] to remove the redundancy of visual concept overlap between clusters.

After merging, we get 100-1000 detectors from each subclass. The numbers differ from each other greatly cause the quantity of data and the discriminativeness of the subclass. The responding visual elements of the ‘UpMid-Down+Scales’ subclasses are showed in Fig. 2.

4. From Part detection to Pedestrian detection

In this section, we give details of 2 methods we used to combine the part detectors to pedestrian scores. The shallow method (the traditional method): encoding images with part detectors and train a linear SVM classifier to output the pedestrian’s score. The deep method (our proposal): transform the detectors to a convolutional layer of the CNN, and trained the model to get the pedestrian score.

4.1. The shallow method

Similar to the method used in [18]. We encode the image with trained part detectors in Sec 3.3, which correspond to the pattern of pedestrian or pedestrian subclasses. Consider the computational cost, mined detectors are not all used. We first rank the rules in a subclass base on the number of the training images that they cover, then select the detectors corresponding to the rules which cover the maximum number of training images. The same number of detectors are selected from each subclass and stacked together to N detectors.

We run our N detectors at each \( W \times H \times 512 \) Conv layer feature map locations to get a \( W \times H \times N \) new feature map. Then apply max pooling 5 times (1 covers the whole image and 4 cover every 1/4H) to get a \( 1 \times 1 \times 5N \) feature vector. Finally, we train a linear SVM to combine a pedestrian score. The pipeline of the shallow method is showed in Fig. 3(a).

4.2. The deep method

By observing the pipeline of the shallow method, we found that the process of encoding and classification is surprisingly similar with the CNN forward processing. Owing to the fact that we use the Conv layer feature to construct detector. The seamless connection of feature extraction, part detection, encoding and classification can be merged into one simple CNN model. Showed in Fig. 3(b) and 3(c). Then, we train the entire CNN model on the Caltech-usa...
Figure 2. Examples of image patches clusters that correspond with the mined rules. These clusters of patches are used to train part detectors.

Figure 3. Two methods to combine part detectors for getting the pedestrian score. (a). The shallow method: Encoding the image by 5 times' max pooling of part detection results. Then use a linear SVM to combine the encoded feature vector. (b). The deep conv method: Transform part detectors to Conv layer and trained the whole CNN model to get the pedestrian score. (c). The deep fc method: Transform part detectors to Fc layer and trained the whole CNN model to get the pedestrian score.
dataset to get the pedestrian score. The details of model construction are given as follows:

**Turn part detectors to Conv layer**

1) Normalization size: we fix the input size to 256 × 128 which both adapt the average ratio of the Caltech-usa dataset and the input size of our base model VGG19[23].

2) Conv layer: we keep all of the conv layers of VGG19 and put another conv layer, named DP layer(Discriminative Part layer). DP layer transformed from part detectors with the weight of 1 × 1 × 512 × N comes from N part detectors’ LDA classifier weights. The transform formula is showed as follows:

\[ \hat{convW}(1, 1, i) = L_i, \]  

(5)

\( \hat{convW}(1, 1, i) \) means \( \text{ith} \) channel of the conv weight and \( L_i \) means the LDA’s 512-dimension weight of \( \text{ith} \) part detector.

3) fully-connected layers: add two 4096-dimensions fc, one 2-dimensions fc and one softmax.

**Turn part detectors to Fc layer**

1) Normalization size: we fix the input size to 256 × 128 as same as the above conv layer method.

2) Conv layers: we keep all of the conv layers and add DP fc layer and normal fc layers behind.

3) Fully-connected layers: we transform part detectors as the first fc layer. Cause detectors mined from the subclasses have per-given location information(Up,Mid or Down), the part detectors are only use to substitute the fc weights from the corresponding location. Finally we get a N-dimensions fc layer. The formula is showed as follows:

\[ \hat{fcW}(h_i, w, i) = L_i, \]  

(6)

where \( \hat{fcW}(h, w) \) means the weight of the fc layer which connects to \( (h, w) \) of the front conv layer. \( h_i = \{h_{up}, h_{mid}, h_{down}\} \) which represents the detectors original location of the height. Finally, add one N-dimensions fc, one 2-dimensions fc and one softmax.

### 5. Experiment

We process all the experiment on Caltech-usa dataset[12, 11]. Strictly follow the evaluation protocol of [12, 11], we use subsets set00~set05 for training and subsets set06~set10 for testing. We use Caltech1x and Caltech10x subsets(every 30th and 3rd frame of the video) to adapt different tasks in the experiments. The log-average miss rate over nine points ranging from \(10^{-2}\) to \(10^{0}\) False Positive Per Image are employed to evaluate the pedestrian detection performance. We use LDCF[29] to provide detection proposals. By renewing the scores of each proposal we evaluate our method’s performance on pedestrian detection. We use following 3 subsets of Caltech-test set to evaluate the synthesized pedestrian detection performance and occlusion handling:

*Reasonable* subset. Pedestrians are larger than 49 pixels in height and have at least 65 percent visible body parts.

*Partial occlusion* subset. Pedestrians are larger than 49 pixels in height and have at 1-35 percent occluded body parts.

*Heavy occlusion* subset. Pedestrians are larger than 49 pixels in height and have at 36-80 percent occluded body parts.

The *Reasonable* subset is used in all the experiments. *Partial occlusion* subset and *Heavy occlusion* subset are used in Sec.5.3 and 5.4 to evaluate the occlusion handling ability of our best proposal.

#### 5.1. Evaluation on CNN feature representation

As we mentioned in Sec3.1, this experiment is used to confirm that the Conv layer feature has the ability to represent images as well as the fc layer. In this experiment, the CNN model we use is VGG19[23] per-trained in ImageNet. We compare 3 representation using the CNN features as follows:

- **Fc.** Use the 4096-dimension fully connected layer as the feature of the image.
- **Conv.** Normalize the image to 256 × 128, and feed it to the CNN model without all the fully-connected layers. Simply connected all the pixels in the last conv feature map\((16\times8\times512)\) to construct a 66536-dimension feature.
- **Conv max pooling.** Feed a normalized image to the CNN model like method 2, and use the max pooling of the last conv layer as the feature.

Then train a linear SVM to classify the pedestrians from backgrounds from Caltech1x. The log-average miss rate is used in this case to evaluate the performance of the feature, which is achieved by renewing the stores of proposals bounding box. The miss rate is summarized in Table 1. Although the max pooling of conv layer is slightly worse than fc layer, the performance of connected conv layer is better than fc layer which means there is more information in the conv layer to represent the pedestrian. Therefore, we decided to use conv layer as our representation of image patches.

### 5.2. Evaluation on attribute subclasses

In this subsection, we evaluate the effectiveness of the part detectors mined from attribute subclasses based on the shallow method. We select 100-300 detectors from each subclass, which cover the most of the training images. The results are shown in Table 2. It shows that the parts detector mined from subclasses perform better
### Table 1

<table>
<thead>
<tr>
<th>Method</th>
<th>MR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fc</td>
<td>62.38</td>
</tr>
<tr>
<td>Conv</td>
<td>41.32</td>
</tr>
<tr>
<td>Conv max pooling</td>
<td>65.4</td>
</tr>
</tbody>
</table>

Table 1. Log-average miss rate(%) of fc, conv and conv max pooling feature represents in Caltech-test dataset, all results are obtained by SVM classifiers covered on the related feature vector.

### Table 2

<table>
<thead>
<tr>
<th>Detector</th>
<th>detectors</th>
<th>MR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Holistic</td>
<td>300</td>
<td>46.32</td>
</tr>
<tr>
<td>UMD</td>
<td>600</td>
<td>41.16</td>
</tr>
<tr>
<td>UMD+Scales</td>
<td>600</td>
<td>39.34</td>
</tr>
</tbody>
</table>

Table 2. The effectiveness of detector sets and their combinations based on 3 mining methods. The final pedestrian scores are combined by linear SVMs. In this table, we can see that the part detectors constructed by the UpMidDown subset mining have better performance than the original mining set, and the UMD+Scales detector sets achieve a better pedestrian detection performance. MR: Log-average miss rate(%).

### Table 3

<table>
<thead>
<tr>
<th>Model</th>
<th>reasonable</th>
<th>partial occl</th>
<th>heavy occl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shallow</td>
<td>39.34</td>
<td>48.21</td>
<td>82.33</td>
</tr>
<tr>
<td>VGG19</td>
<td>18.10</td>
<td>28.48</td>
<td>65.46</td>
</tr>
<tr>
<td>ConvDP</td>
<td>17.47</td>
<td>30.73</td>
<td>66.58</td>
</tr>
<tr>
<td>FcDP</td>
<td>17.14</td>
<td>28.12</td>
<td>62.33</td>
</tr>
</tbody>
</table>

Table 3. The Log-average miss rate(%) of different deep methods on Caltech-usa dataset. Compare 2 deep methods with the VGG19 fine-tuning and the shallow method.

than detectors from the holistic set. In addition, more detailed the subclasses are per-installed, better performance the part detectors achieved. The part detectors mined from UpMidDown+Scales’s 6 subclasses get the best result of 39.34%. The effectiveness of complementary part detectors mined from different pedestrian attributes is proved in this experiment.

### 5.3. Evaluation on deep method

In this subsection, we evaluate two deep methods. We evaluate the pedestrian detection performance of those methods on Caltech-usa reasonable, Partial occlusion and Heavy occlusion dataset. We use Caltech10x, which takes every 3rd frame of video, to train our models. The detection results of our model are shown in Table.3. By the way, the part detectors we use to construct deep method are all from the 'UpMidDown+Scales'-set which achieved the best performance in the shallow method experiments.

Compare the shallow method with all the deep method. We can obviously see the optimization effect. The best deep method FcDP outperforms shallow method 22.2% MR in Reasonable subset, and improves MR on

### 5.4. Overall Evaluation

We compare our approach with the existing best-performing methods, including VJ[28], HOG[7], ACF+SDT[22], Jointdeep[20], SDN[19], LDCF[29], SCF+AlexNet[15], Katamari[3], SpatialPooling+[21], TACNN[25] and Deepparts[26]. The evaluation results on the reasonable, Partial Occlusion and Heavy Occlusion are shown in Fig.4, Fig.5 and Fig.6. Our approach achieves 17.14% Log-average miss rate, outperforms most of the existing methods. Although our approach slightly underperformed the state-of-the-art method Deepparts, cause we just implement one time of CNN computing, our method is 50 times faster than theirs.
6. Conclusion

In this paper, we proposed the DP-CNN to handling occlusion problems with pedestrian detection. We first propose a mining method which can mine discriminative body parts with the pedestrian bounding box images only. Secondly, we propose a brand new approach to transform the part detectors to CNN middle layers and train the CNN model to get an astonishing optimization. Our method shows robust occlusion handling and lowers computational cost comparing the traditional CNN fine-tuning method. Our future work will focus on the network structures and other object recognition applications.

Acknowledgement This research is supported by the JSPS Grant-in-Aid for Scientific Research B (No. 26280057), the JSPS Grant-in-Aid for Challenging Exploratory Research (No.16K12460), and the JST Center of Innovation Program.

References


