

Pedestrian Detection Inspired by Appearance Constancy and Shape Symmetry

Jiale Cao¹, Yanwei Pang^{1*}, Xuelong Li²

¹School of Electronic Information Engineering, Tianjin University

²Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences

connor@tju.edu.cn, pyw@tju.edu.cn, xuelong_li@opt.ac.cn

Abstract

The discrimination and simplicity of features are very important for effective and efficient pedestrian detection. However, most state-of-the-art methods are unable to achieve good tradeoff between accuracy and efficiency. Inspired by some simple inherent attributes of pedestrians (i.e., appearance constancy and shape symmetry), we propose two new types of non-neighboring features (NNF): side-inner difference features (SIDF) and symmetrical similarity features (SSF). SIDF can characterize the difference between the background and pedestrian and the difference between the pedestrian contour and its inner part. SSF can capture the symmetrical similarity of pedestrian shape. However, it's difficult for neighboring features to have such above characterization abilities. Finally, we propose to combine both non-neighboring and neighboring features for pedestrian detection. It's found that non-neighboring features can further decrease the average miss rate by 4.44%. Experimental results on INRIA and Caltech pedestrian datasets demonstrate the effectiveness and efficiency of the proposed method. Compared to the state-of-the-art methods without using CNN, our method achieves the best detection performance on Caltech, outperforming the second best method (i.e., Checkerboards) by 1.63%.

1. Introduction

Pedestrian detection is a premise in many computer vision tasks including gait recognition, behavior analysis, action recognition, and camera-based driver assistance. Generally speaking, the performance of pedestrian detection is determined by the performance of feature extraction and classification. This paper focuses on feature extraction.

There are three manners for feature extraction: (1) completely Hand-Crafted (HC) features, (2) Hand-Crafted candidate features followed by Learning Algorithms (HCLA), and (3) Deep Learning (DL) based features. Due to simplicity and robustness, it is much more possible for HCLA to achieve good tradeoff between efficiency and accuracy. So

this paper concentrates on HCLA.

Usually, the input of HCLA for pedestrian detection is CIE-LUV color channels, gradient histogram channels, gradient magnitude channel, etc. Once the channels are specified, the question remained is how to generate candidate features from the channels. Most of state-of-the-art methods generate the candidate features by using local (e.g., local mean features) or neighboring features (e.g., haar features). In fact, some inherent attributes of pedestrians can be also used for feature design. Inspired by appearance constancy and shape symmetry of pedestrians, we design two types of non-neighboring features: side-inner difference features (SIDF) and symmetrical similarity features (SSF). The contributions of the paper are as follows:

1) Appearance constancy and shape symmetry can be seen as the inherent attributes of pedestrians. Inspired by these attributes, we propose side-inner difference features (SIDF) and symmetrical similarity features (SSF), respectively. Compared to some state-of-the-art features, our features are oriented non-neighboring features. SIDF can characterize the difference between the background and pedestrian and the difference between the pedestrian contour and its inner part. SSF can capture the symmetrical similarity of pedestrian shape. However, it's difficult for neighboring features to have such above characterization abilities.

2) We propose to employ non-neighboring and neighboring features for pedestrian detection. Among all the selected features, about 70% are neighboring features and 30% of them are non-neighboring ones. So the non-neighboring features are complementary to the neighboring ones.

3) Compared to the state-of-the-art methods without using CNN, we achieve the best detection performance (i.e., 16.84% miss rate on Caltech). Meanwhile, our methods achieve the best performance tradeoff between detection efficiency and log-average miss rate only by common CPU. Moreover, SIDF and SSF may also be combined with CNN features to further boost the detection performance.

The rest of the paper is organized as follows. We review related work in Section 2. The proposed method is given in Section 3. Experimental results are provided in Section 4.

We then conclude in Section 5.

2. Related work

Pedestrian detection methods can be divided into three families [4]: DPM (Deformable Part Detectors) variants [13, 14, 23, 27], deep networks [15, 17, 28] and decision forests [2, 11, 32]. Our method can be categorized into the family of decision forests. Specifically, the process of this kind of methods is as follows: 1) a set of channel images are generated from an input image; 2) then, features are extracted from patches of the channels; and 3) finally, the features are fed into a decision forest learned via AdaBoost [37]. Feature extraction is a very important step.

Integral Channel Features (ICF) [11] is one of the most successful feature extraction method four years after Histograms of Oriented Gradients (HOG) [8] was proposed. In ICF, Dollár *et al.* [11] proposed to combine three types of channels: LUV color channels, normalized gradient magnitude, and histogram of oriented gradients (6 channels). First-order and higher-order features are then generated from the channel images [11]. Soft cascade [5, 35] is then used for learning discriminative features [11]. Note that ICF is also known as ChnFtrs.

Aggregated Channel Features (ACF) [9], SquaresChnFtrs [3], InformedHaar [36], Locally Decorrelated Channel Features (LDCF) [21], and Checkerboards [37] employ the same channel images as ICF. In ACF, the pixel sum of each block in each channel is computed and then the resulting lower resolution channels are smoothed [9, 10]. SquaresChnFtrs [3] is simpler than ICF because only the local sum of squares in each channel image is used as features. InformedHaar [36] is specifically designed for pedestrian detection where a pool of rectangular templates are tailored to the statistical model of the up-right human body across the channels. By using the technique of Linear Discriminant Analysis (LDA) [16, 26], the LDCF features are decorrelated so that they are suited for orthogonal decision trees [21]. The decorrelation can be achieved by convolution with a PCA-like filter bank. Checkerboards [37] generalizes ICF by using filter banks to compute features from channel images. Six types of filters are considered: InformedFilters, CheckerboardsFilters, RandomFilters, SquaresChntrs filters, LDCF8 filters, and PcaForeground filters.

SpatialPooling+ [25, 24] does not take channel images as input. Instead, it applies the operator of spatial pooling (e.g., max-pooling) on covariance descriptor and Local Binary Pattern (LBP). Blob-like operator [19] and Torque operator [22] can be also used for image abstraction.

According to [4] and our experimental results, the performance of the above methods can be summarized as follows: On the Caltech pedestrian dataset [1, 12], the miss rates of the above methods are $ICF > ACF > SquaresChnFtrs > InformedHaar > LDCF > SpatialPooling+ >$

Checkerboards. Loosely speaking, the detection speeds of these methods are $SpatialPooling+ < ICF < SquaresChnFtrs < Checkerboards < InformedHaar < LDCF < ACF$. It can be concluded that no method can simultaneously obtain the best log-average miss rate and detection speed. That is, these methods are unable to achieve satisfying tradeoff between accuracy and efficiency.

Recently, the methods based on CNN have achieved very good performance [6, 20, 30, 31, 34]. For example, Tian *et al.* [30] proposed DeepParts to improve the detection performance by handling occlusion with an extensive part pool. Though the methods based CNN can achieve the best performance, it needs the relatively expensive device (i.e., GPU). On the other hand, the simple feature design can also be complementary to CNN. For example, by combining the simple local features (e.g., ACF [9], Checkerboards [37], and LDCF [21]) and very deep CNN features (e.g., VGG [29] and AlexNet [18]), Cai *et al.* [6] could decrease the miss rate from 18.9% to 11.70%. So in this paper, we focus the feature design in the traditional methods.

3. Our methods

3.1. Appearance constancy and shape symmetry

Most state-of-the-art features for pedestrian detection are designed to describe the local image region. Thus, they don't make full use of the inherent attributes of pedestrians. In fact, some inherent attributes of pedestrians can be used to further improve detection performance. For example, Zhang *et al.* [36] incorporate the common sense that pedestrians usually appear up-right into the design of simple and efficient haar-like features. In this paper, we incorporate the appearance constancy and shape symmetry into the feature design. First of all, we give the explanations of appearance constancy and shape symmetry. Fig. 1 gives some examples of the cropped pedestrians.

1) *Appearance Constancy*. The appearances of pedestrians are usually contrast to the surrounding background. Meanwhile, pedestrians can be seen as three main different parts (i.e., head, upper body, and legs). The appearances inside these parts are usually constancy. For example, the woman wears the sky-blue coat and black pants in Fig. 1(a). We call this inherent attribute of pedestrians *appearance constancy*. Thus, the regions located inside the pedestrians (e.g., patches *B* in Figs. 1(a) and (b)) are contrast to that located in the background (e.g., patches *A* in Figs. 1(a) and (b)). Note that patches *A* and *B* lie in the same horizontal. Patches *B* are called the inner patches, and patches *A* are called the side patches.

2) *Shape Symmetry*. As stated in [36], pedestrians usually appear up-right. Thus, the pedestrian shape is loosely symmetrical in the horizontal direction. For example, the symmetrical region (patches *A* and *A'*) in the Figs. 1(c) and

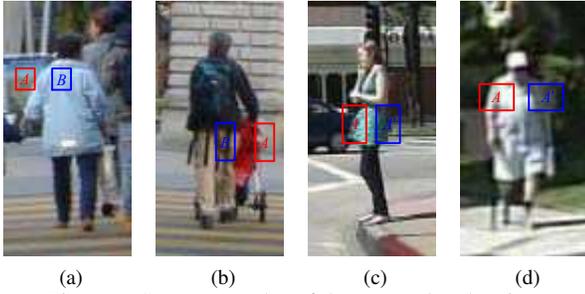


Figure 1. Some examples of the cropped pedestrians.

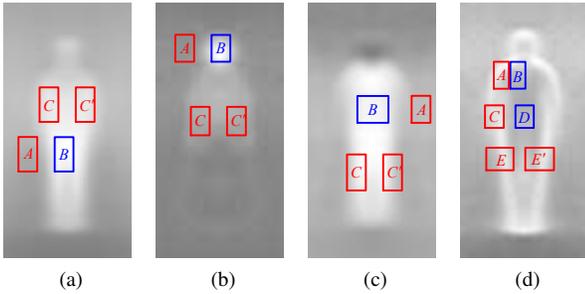


Figure 2. Average values of channel images . (a) Inversed L (Luminance) channel. (b) U channel. (c) Inversed V channel. (d) G channel.

(d) have the similar characteristic. This inherent attribute is called *shape symmetry*.

Inspired by the above appearance constancy and shape symmetry, we can design two types of non-neighboring features. It can be explained by the average appearance of pedestrians in channel images such as L, U, V, and G. Fig. 2 gives the average values of above four channel images. Due to the appearance constancy, the pixel values of pedestrians in L, U, and V channel images are similar in the same horizontal, which are different from that of the two-side regions. Meanwhile, the pixel values of the inner part of pedestrians in G channel image are constantly small, and the pixel values of pedestrian contour in G channel image are relatively large. Thus, the large difference in G channel image can be characterized by not only the neighboring feature formed by patches *A* and *B* but also the non-neighboring feature formed by patches *C* and *D* in Fig. 2(d). Though there is little difference between the inner part and contour in V channel (Fig. 2(c)), the difference between the inner part and its two side background can be characterized by the non-neighboring feature formed by patches *A* and *B*. Due to shape symmetry, the symmetrical regions of pedestrians in the horizontal direction have the similar characteristic. For example, the symmetrical patches *E* and *E'* in Fig. 2(d) describe the similar edge characteristic, while patches *C* and *C'* in Fig. 2(c) are both bright. Figs. 2(a) and (b) also support the above two conclusions.

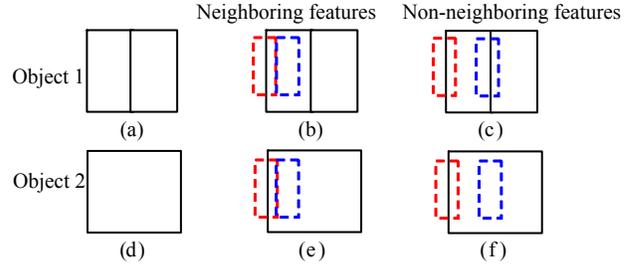


Figure 3. Demonstration of the discrimination and usefulness of non-neighboring features. (a) and (d) show Object 1 and Object 2, respectively. In (b) and (e), neighboring features are extracted. In (c) and (f), non-neighboring features are extracted.

The discrimination and usefulness of non-neighboring features are graphically supported by Fig. 3. In Fig. 3, there are two objects (classes) to be classified. We call the object in Fig. 3(a) Object 1 and the object in Fig. 3(d) Object 2. There is a line in the middle of Object 1 whereas the inner part of Object 2 is flat. In both Figs. 3(b) and (e), two neighboring dashed rectangles form a feature. We can see that this neighboring feature is unable to distinguish between Object 1 and Object 2 because the values of neighboring features in Object 1 (i.e., Fig. 3(b)) and Object 2 (i.e., Fig. 3(e)) are equal. Now we use in Figs. 3(c) and (f) two non-neighboring patches to form a feature. Because the blue dashed patch in Fig. 3(c) contains a line whereas the blue dashed patch in Fig. 3(f) contains nothing, the non-neighboring features in Object 1 (i.e., Fig. 3(c)) and Object 2 (i.e., Fig. 3(f)) have different values. The two objects can be correctly classified according to the different values. This demonstrates the discrimination and usefulness of non-neighboring features.

3.2. Side-inner difference features inspired by appearance constancy

Inspired by appearance constancy, we design the non-neighboring difference features in the same horizontal. We call this oriented non-neighboring difference features Side-Inner Difference Features (SIDF). Fig. 4 gives some possible forms of SIDF. Fig. 4(a) shows that the distance $d(A, B)$ of non-neighboring patches *A* and *B* in SIDF can be different. Theoretically, the distance can be arbitrary. But it results that the number of all possible SIDF is very large. Because a pedestrian is horizontally symmetrical in a loose sense, we restrict the location $l(B)$ of *B* in the interval of the locations $l(A)$ and $l(A')$ where A' is the horizontal mirror of *A*. That is, $l(B) \in [l(A), l(A')]$. As demonstrated in Fig. 5, $l(B)$ is randomly sampled from $[l(A), l(A')]$ in our experiments.

Both Figs. 4(b) and (c) show varying sizes of patches. But in Fig. 4(b) both two non-neighboring patches equally vary with size (scale) whereas in Fig. 4(c) only one patch

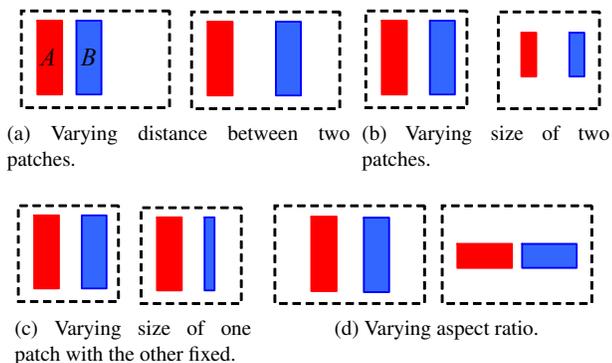


Figure 4. Some possible forms of side-inner difference features (SIDF).

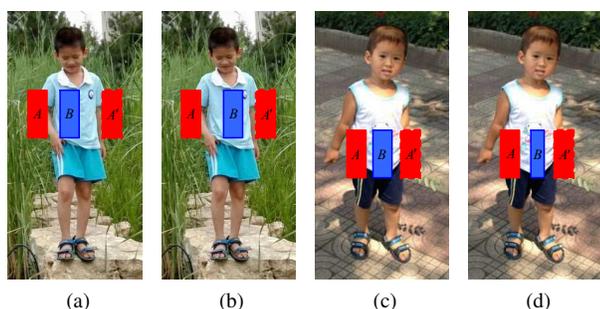


Figure 5. The patch B is randomly located between the patch A and its horizontal mirror A' . The locations of patch B in (a) and (b) are different. But they are both among A and its mirror A' . (c) and (d) show that the width of patch B can be changed.

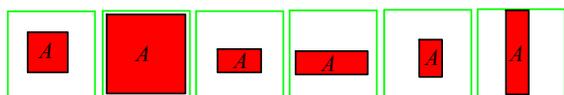


Figure 6. The size of patch A is allowed to change inside the maximum square indicated by green squares.

varies its size. It's good enough for letting A and B have the different width but the same height. Figs. 5(c) and (d) also give an example of the different widths of patches A and B . Fig. 4(d) shows SIDF with varying aspect ratio.

The size of a patch (e.g., patch A in Fig. 5(a)) is allowed to change in a reasonable range. In this paper, the variation of a patch is limited to a maximum square. In other words, the sizes of both patches A and B are allowed to be not larger than that of the maximum square. The green squares in Fig. 6 are maximum squares and patches have to be inside them. A typical maximum square is of size 8×8 cells (1 cell = 2×2 pixels).

Suppose that the side-inner difference feature $f(A, B)$ consists of two patches A and B (see Fig. 5(a)). The number of pixels of A and B are denoted by N_A and N_B . Let S_A and S_B be the pixel sum in A and B in each channel, respectively. Then the side-inner difference feature $f(A, B)$

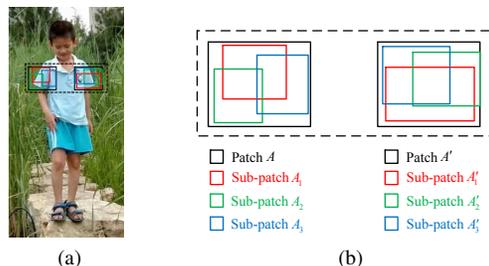


Figure 7. Examples of symmetrical similarity features. (a) is an example of SSF located in the pedestrian. (b) shows a specific form of SSF.

can be calculated by

$$f(A, B) = \frac{S_A}{N_A} - \frac{S_B}{N_B}, \quad (1)$$

where N_A and N_B are used for normalization.

3.3. Symmetrical similarity features inspired by shape symmetry

As stated in Section 3.1, the shape of pedestrian is symmetrical. Thus, patches A and A' in Fig. 5 have the similar characteristic. The symmetrical similarity features $f(A, A')$ of patches A and A' can be calculated by the following equation:

$$f(A, A') = |f_A - f_{A'}|, \quad (2)$$

where f_A and $f_{A'}$ represent the features of patches A and A' (e.g., histogram features and local mean features). For the computational efficiency, we just use the local mean features to represent the patches. Namely, $f_A = S_A/N_A$ and $f_{A'} = S_{A'}/N_{A'}$. Then, Eq. (2) can be written as the following:

$$f(A, A') = \left| \frac{S_A}{N_A} - \frac{S_{A'}}{N_{A'}} \right|. \quad (3)$$

However, due to the changes of the pedestrian posture, the pedestrian symmetry is relatively loose. It results that Eq. (3) is very sensitive to the pedestrian deformation.

To eliminate the above influence caused by pedestrian deformation, we replace the mean features of patches by the max-pooling features [33]. In Fig. 7, two symmetrical patches A and A' are represented by three different color sub-patches, respectively. For examples, patch A consists of three sub-patches A_1 , A_2 , and A_3 . The sub-patches are randomly generated inside the patch A . The size and aspect ratio of them can arbitrary, whereas the area of them should be larger than half of patch A . Then, the feature value of patch A is set as the maximum of mean values of three sub-patches. It can be expressed as:

$$f_M(A) = \max_{i=1,2,3} \frac{S_i}{N_i}. \quad (4)$$

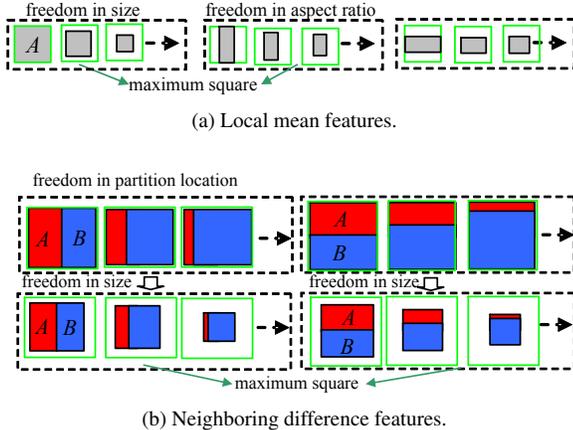


Figure 8. Some possible forms of neighboring features. The green squares are called maximum squares.

Note that the maximum is replaced by minimum in L and V channel images. Then, the symmetrical symmetry features $f(A, A')$ of patches A and A' is calculated by the following equation:

$$f(A, A') = |f_M(A) - f_M(A')|. \quad (5)$$

The size of the symmetrical patches A and A' is allowed to change in a reasonable range, which varies from 6×6 cells to 12×12 cells. As the symmetry in pedestrians mainly exists in L, U, V, and G channel images, we only use the above channel images to generate SSF.

3.4. Neighboring features

In fact, both non-neighboring and neighboring features are crucial for pedestrian detection. In this section, we propose to form the pool of neighboring features by using local mean features (see Fig. 8(a)) and neighboring difference features (see Fig. 8(b)) with enough freedom in size, aspect ratio, patch direction, and partition location. The left portion of Fig. 8(a) shows that the size of a feature is allowed to vary in a large extent. Patch direction is either vertical or horizontal. The patch direction in the middle of Fig. 8(a) and the left portion of Fig. 8(b) is vertical whereas the direction in the right portion of Fig. 8(a) and the right portion of Fig. 8(b) is horizontal.

Partition location is illustrated in Fig. 8(b) which is defined as the location where two neighboring patches intersect. Assigning freedom in partition location strengthens representative and discriminative ability of the features.

To avoid the large number of features, we specify a maximum square. The sizes of local mean features and neighboring difference features are allowed to be not larger than the size of the maximum square. The green squares in Fig. 8 are maximum squares. As stated in Section 3.2, a typical size of the maximum square is 8×8 cells.

The neighboring features illustrated in Fig. 8 are suitable to be computed with integral image. Hence the feature extraction process is very efficient. Note that neighboring difference features can be calculated using the same formula (i.e., Eq. (1)) of non-neighboring features.

In our method, both the neighboring (i.e., local mean features and neighboring difference features) and non-neighboring features (i.e., SIDF and SSF) are used as input of decision forests and AdaBoost.

4. Experiments

The public Caltech pedestrian dataset [1, 12] and INRIA dataset [8] are employed for evaluation. In the INRIA dataset, there are 1237 pedestrian images used for training and 288 pedestrian images used for evaluation.

The Caltech pedestrian dataset is more challenging than the INRIA dataset and hence has become a benchmark. It consists of approximately 10 hours of 640×480 30Hz video [1]. The 10 hours data consists of 11 videos with the first 6 videos are used for training and the last 5 videos for testing. The standard positive training data is formed by sampling one image out of each 30 sequential frames. To enlarge the number of training samples, we sample a frame from every 15 or 3 frames instead of every 30 frames. The resulting training sets are called Caltech 2x and Caltech 10x [37]. Whenever Caltech 2x training set or Caltech 10x training set is used, the test dataset is the same. The testing dataset consists of 4024 frames among which there are 1014 positive images.

4.1. Self-comparison using the Caltech 2x training data

Before comparing with state-of-the-art methods, experimental results on Caltech 2x dataset are reported to show how the proposed method works and the importance of each component of the proposed method. Note that the Caltech 2x training set instead of Caltech10x training set is used.

The experimental setup is as follows. Classical 10 channel images (i.e., HOG+LUV) are used for generating features. The final classifier consists of 4096 level-2 decision trees. The classifier is learned by five rounds, where the numbers of trees in subsequent rounds are 32, 128, 512, 2048, and 4096, respectively. Each tree is built by randomly sampling $1/32$ of features from the large pool of features. 5000 hard negatives are added after each round and the cumulative negatives are limited to 15000. The stride of sliding windows is 4 pixels. The model size is 64×128 , which consists of 2048 cells (1 cell= 2×2 pixels). As the pedestrian is generally taller than 50 pixels, each testing image is upsampled by one octave.

In NNF (a.k.a. NNF+NF), both Non-Neighboring Features (NNF) and Neighboring Features (NF) are employed. In the NNF, there are two types of non-neighboring features:

Channel	L	U	V	G	6 Oriented gradients
Method	$\frac{x-\mu}{\sigma}$	x			$\frac{x}{\mu_G}$

Table 1. Channel-Specific Normalization

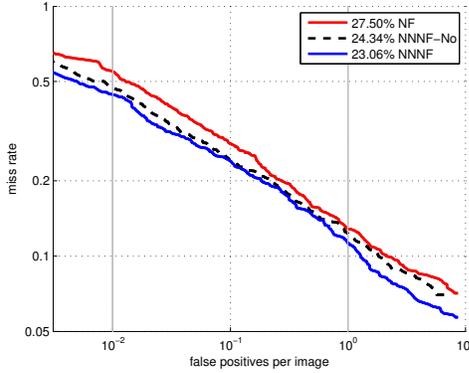


Figure 9. Self-comparison: ROC curves of NF, and NNNF-No, and NNNF on the Caltech dataset.

Method	MR	Δ MR
NF	27.50%	N/A
NF+SIDF	25.67%	+1.83%
NF+SSF	25.20%	+2.30%
NNNF-No	24.34%	+3.16%
NNNF	23.06%	+4.44%

Table 2. Comparison of Log-average Miss Rates

SIDF and SSF. NF+SIDF or NF+SSF mean that the neighboring features are combined with only one type of non-neighboring features (i.e., SIDF or SSF). In SIDF and NF, the channel-specific normalization can be used in Table 1. In Table 1, x is a feature in a detection window. μ and σ are the mean and variance of the features in the detection window. μ_G is the mean of G channel. Because U and V channels are relatively stable to variations in illumination, we do not perform normalization. We denote NNNF-No the method which is the same as NNNF except that no normalization is conducted in SIDF and NF.

The ROC curves of NF, NNNF-No and NNNF are shown in Fig. 9. It is seen that the performance of NNNF is systematically better than that of NF, meaning that incorporating NNF is useful for improving detection performance. Meanwhile, one can observe that NNNF-No is inferior to NNNF. NNNF employs channel-specific normalization in NF and SIDF whereas NNNF-No does not perform normalization. So it is concluded that pedestrian detection benefits from the proposed channel-specific normalization.

The above observation can also be seen from Table 2 where the log-average miss rates are given. The miss rates of NNNF (i.e., NNF+NF), NNNF-No, and NF are 23.06%, 24.34%, and 27.50%, respectively. The miss rate

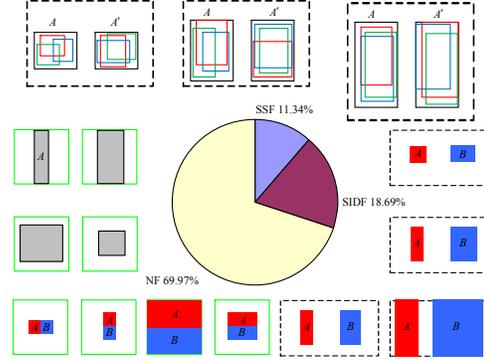


Figure 10. Among all the selected features, about 30% are non-neighboring features and 70% are neighboring features. Some representative non-neighboring and neighboring features also shown.

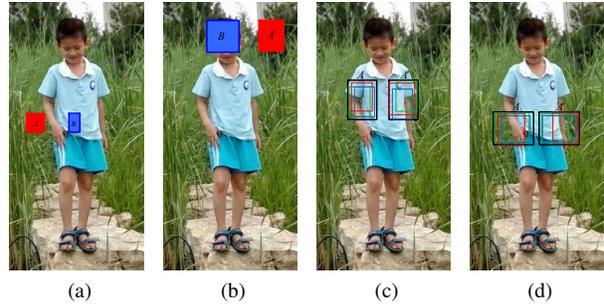


Figure 11. Several selected non-neighboring features. The first two features are SIDF, and the last two features are SSF.

of NNF+NF is 4.44% smaller than that of NF. So it is said that non-neighboring features contribute significantly for improving detection performance. Specifically, NF+SIDF and NF+SSF outperform NF by 1.83% and 2.30%, respectively. NNNF outperforms NNNF-No by 1.28%. Though the contribution of channel-specific normalization is not as significant as non-neighboring features, it is steadily helpful for improving detection performance.

Totally, 12288 features are selected, which consist of 3690 non-neighboring features and 8598 neighboring features. Among non-neighboring features, there are 2297 side-inner difference features (SIDF) and 1393 symmetrical similarity features (SSF). That is, the proportions of SIDF, SSF, and NF are approximately 18.69%, 11.34% and 69.97% (see Fig. 10). We can conclude that non-neighboring features are complementary to neighboring features. Several representative forms of non-neighboring (SIDF and SSF) and neighboring features (NF) are also shown in Figs. 10.

In Fig. 11, the representative non-neighboring features are also visualized on pedestrian images. The first two images show the side-inner difference features, and the last two images show the symmetrical similarity features.

In fact, SIDF features can be categorized into the follow-

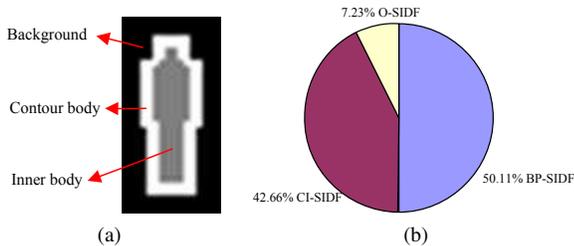


Figure 12. CI-SIFD, BP-SIFD, and O-SIFD features. (a) The ternary model of pedestrians. (b) The portions of SIFD.

ing three types: 1) A SIFD feature is called Contour-Inner SIFD (CI-SIFD) feature if one of its patch is located on the pedestrian contour and the other is located inside the pedestrian; 2) A SIFD feature is called Background-Pedestrian SIFD (BP-SIFD) feature if one of its patch is on the background and the other patch is inside or on the contour of a pedestrian; and 3) A SIFD feature different from CI-SIFD and BP-SIFD features is called Other SIFD (O-SIFD) feature. To know the proportions of the three types of SIFD features, a ternary model (Fig. 12(a)), consisting of background, contour body, and inner body, is created according to average appearance (e.g., Fig. 2(d)) of pedestrians. All the 2297 selected SIFD features are classified to CI-SIFD, BP-SIFD, and O-SIFD by computing the intersection of a SIFD feature and the ternary model. The results given in Fig. 12(b) indicate that the proportions of CI-SIFD, BP-SIFD, and O-SIFD are 42.66%, 50.11%, and 7.23%, respectively. Fig. 12(b) tells that SIFD features not only capture the difference the contour of a pedestrian and its inner part but also utilize the difference between the background and a pedestrian. Background can be regarded as context of a pedestrian image and hence context has been proved to be effective in object detection and recognition. It is difficult for neighboring features to utilize the context information.

4.2. Comparison with state-of-the-art methods on Caltech dataset

The proposed NNNF method can adopt different levels (depths) decision trees. In this section, NNNF-L2 stands for the NNNF method where level-2 trees are utilized. The Caltech 2x training data is used for NNNF-L2. All parameters in NNNF-L2 are the same as those in Section 4.1. In NNNF-L4, level-4 trees are employed. The Caltech 10x training data is used for NNNF-L4. The resulting classifier is composed of 4096 level-4 decision trees and each tree is built by randomly sampling 1/2 of features from the feature pool. The decision trees are obtained after five rounds. In each round, 20000 hard negatives are added and the cumulative negatives are limited to 50000. Other parameters are the same as those in Section 4.1.

Fig. 13 compares NNNF-L2 and NNNF-L4 with the

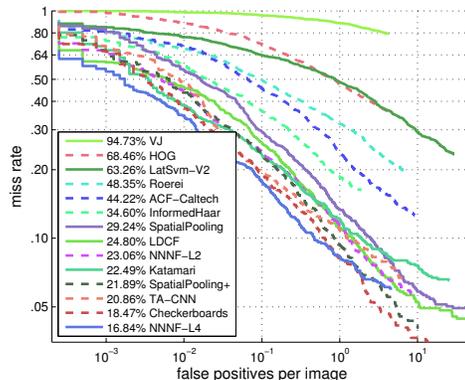


Figure 13. Comparison with state-of-the-art methods on the Caltech dataset.

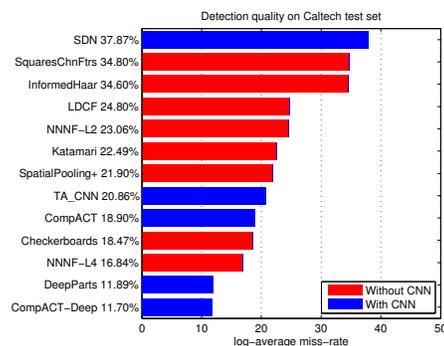


Figure 14. Miss rate of the state-of-the-art methods. The methods with blue bars are based on CNN. The methods with red bars are not using CNN.

state-of-the-art methods. The curves of ACF-Caltech [12] are obtained when they are trained on the Caltech training set. The models of VJ [32], HOG [8], LatSvm-V2 [14], and Roerei [3] are trained on the INRIA dataset. The curves of other methods are obtained when the training set is Caltech 10x. They all utilize the Caltech testing set for evaluation.

The following observations can be seen from Fig. 13. Even the small Caltech 2x training dataset is used, the proposed NNNF-L2 is better than LDCF [21] whose models are trained from the large Caltech10x dataset. Specifically, the log-average miss rate of NNNF-2 is 23.06%.

It can also be seen from Fig. 13 that the proposed NNNF-L4 is superior to all other methods. The log-average miss rate of NNNF-L4 is as small as 16.84% whereas the log-average miss rate of TA-CNN [31] and Checkerboards [37] are 20.86% and 18.47%, respectively. Though the proposed non-neighboring and neighboring features are much simpler than those in CNN and Checkerboards, they result in better detection results.

According to whether using CNN or not, Fig. 14 divides the state-of-the-art methods into two classes. The methods with red bars do not use CNN. NNNF-L4 achieves the

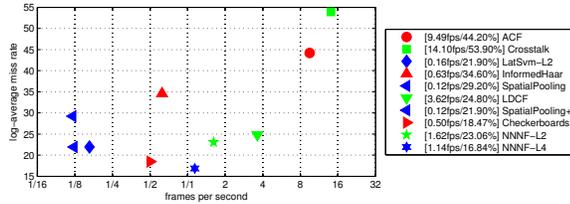


Figure 15. Log-average miss rate (MR) versus frames per second (FPS) on the Caltech.

best detection performance, outperforming Checkerboards [37] by 1.63%. The methods with blue bars are based on CNN. CompACT-Deep [6] achieves the lowest miss rate (i.e., 11.70%) by combination of some local channel features (e.g., ACF [12], Checkerboards [37], and LDCF [21]) and deep features (e.g., VGG [29]). Though CompACT-Deep [6] has a better performance than NNNF-L4, the improvement of CompACT-Deep are based on very deep CNN model (i.e., VGG). When only using the above local features and small CNN, CompACT can only achieve 18.9%, which is inferior to NNNF-L4. It means that NNNF-L4 are much more effective than the local features used in CompACT. With very deep CNN, NNNF can also boost the detection performance [7].

The log-average miss rates and frames per second of the methods without CNN are visualized in Fig. 15. It is desirable if miss rate is as small as possible and FPS is as large as possible. So Fig. 15 implies that the proposed NNNF-L4 achieves the best tradeoff between miss rate and FPS. The log-average rate of NNNF-L4 is superior to that of Checkerboards, and it is also 2.28 times faster than Checkerboards. Note that the detection speed is measured on a computer with an Intel i7 CPU and a 640×480 image with the height of a pedestrian not less than 50 pixels. GPU is not used.

4.3. Comparison with state-of-the-art methods on the INRIA dataset

Experiments are also conducted on the INRIA dataset. Because pedestrian height in both the training and testing sets are larger than 100 pixels, we train a model with 64×128 pixels. The model consists of 2048 level-3 decision trees. Other parameters are the same as those in Section 4.1.

Experimental results are shown in Fig. 16. It can be observed that NNNF achieves the best performance (log-average miss rate is 12.25%). The miss rate of NNNF is 7.71%, 5.03%, and 2.18% lower than that of LatSvm-V2 [14], ACF [9], and InformedHaar [36]. NNNF outperforms LDCF [21] by 1.54%. The advantage of NNNF over LDCF [21] and Roerei [3] is more remarkable for the complex Caltech dataset than for the simple INRIA dataset.

The comparison of detection speed and miss rate of different methods is given in Fig. 17. The image to be detected has 640×480 pixels and the height of a pedestrian

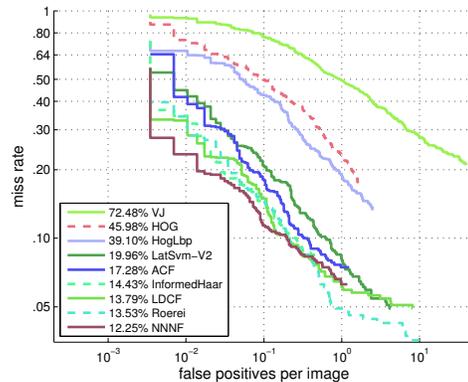


Figure 16. Comparison with state-of-the-art methods on the INRIA dataset.

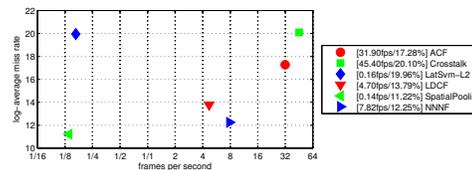


Figure 17. Log-average miss rate (MR) versus frames per second (FPS) on the INRIA.

is not less than 100 pixels. One can see that NNNF outperforms in terms of log-average miss rate all the methods except Spatialpool. Though slightly lower than the miss rate of Spatialpool, NNNF is 55.86 times faster than SpatialPool [24]. Therefore, our method is able to get the best tradeoff between miss rate and detection speed.

5. Conclusion

In this paper, we have presented an effective and efficient pedestrian detection method. The main contribution lies in the proposed two types of non-neighboring features (NNF): side-inner difference features (SIDF) and symmetrical similarity features (SSF) which were found to be complementary to the proposed neighboring features (NF). SIDF features characterize not only the difference between contour of a pedestrian and its inner part but also the difference of the background and pedestrian. SSF can capture the symmetrical similarity of pedestrian shape. Though the forms of the proposed NNF and NF features are very simple, combining them results in the best tradeoff between miss rate and frames per second.

6. Acknowledgment

This work was supported in part by the National Basic Research Program of China (Grant No. 2014CB340400) and the National Natural Science Foundation of China (Grant No. 61222109).

References

- [1] www.vision.caltech.edu/image_datasets/caltechpedestrians.
- [2] R. Appel, T. Fuchs, P. Dollár, and P. Perona. Boosting decision trees-pruning underachieving features early. *Proc. Int'l Conf. Machine Learning*, 2013.
- [3] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Gool. Seeking the strongest rigid detector. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013.
- [4] R. Benenson, M. Omran, J. Hosang, and B. Schiele. Ten years of pedestrian detection, what have we learned? *Proc. European Conf. Computer Vision*, 2014.
- [5] L. Bourdev and J. Brandt. Robust object detection via soft cascade. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005.
- [6] Z. Cai, M. Saberian, and N. Vasconcelos. Learning complexity-aware cascades for deep pedestrian detection. *Proc. IEEE Int'l Conf. Computer Vision*, 2015.
- [7] J. Cao, Y. Pang, and X. Li. Learning multilayer channel features for pedestrian detection. *CoRR abs/1603.00124*, 2016.
- [8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. European Conf. Computer Vision*, 2005.
- [9] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fastest feature pyramids for object detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545, 2014.
- [10] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. *Proc. British Machine Vision Conference*, 2010.
- [11] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. *Proc. British Machine Vision Conference*, 2009.
- [12] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 34(4):743–761, 2012.
- [13] P. Felzenszwalb, R. Girshick, and D. McAllester. Cascade object detection with deformable part models. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
- [14] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.
- [15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2014.
- [16] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. *Proc. European Conf. Computer Vision*, 2012.
- [17] J. Hosang, M. Omran, R. Benenson, and B. Schiele. Taking a deeper look at pedestrians. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2015.
- [18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Proc. Advances in Neural Information Processing Systems*, 2012.
- [19] T. Lindeberg. Detecting salient bloblike image structures and their scales with a scalespace primal sketch: A method for focusofattention. *Int'l J. Computer Vision*, 11(3):283–318, 1993.
- [20] P. Luo, Y. Tian, X. Wang, and X. Tang. Switchable deep network for pedestrian detection. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2014.
- [21] W. Nam, P. Dollár, and J. H. Han. Local decorrelation for improved pedestrian detection. *Proc. Advances in Neural Information Processing Systems*, 2014.
- [22] M. Nishigaki, C. Fermuller, and D. DeMenthon. The image torque operator: A new tool for mid-level vision. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012.
- [23] W. Ouyang and X. Wang. Single-pedestrian detection aided by multi-pedestrian detection. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013.
- [24] S. Paisitkriangkrai, C. Shen, and A. van den Hengel. Strengthening the effectiveness of pedestrian detection. *Proc. European Conf. Computer Vision*, 2014.
- [25] S. Paisitkriangkrai, C. Shen, and A. van den Hengel. Pedestrian detection with spatially pooled features and structured ensemble learning. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2015.
- [26] Y. Pang, S. Wang, and Y. Yuan. Learning regularized lda by clustering. *IEEE Trans. Neural Netw. Learning Syst.*, 25(12):2191–2201, 2014.
- [27] Y. Pang, H. Yan, Y. Yuan, and K. Wang. Robust cohog feature extraction in human-centered image/video management system. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(2):458–468, 2012.
- [28] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013.
- [29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR abs/1409.1556*, 2014.
- [30] Y. Tian, P. Luo, X. Wang, and X. Tang. Deep learning strong parts for pedestrian detection. *Proc. IEEE Int'l Conf. Computer Vision*, 2015.
- [31] Y. Tian, P. Luo, X. Wang, and X. Tang. Pedestrian detection aided by deep learning semantic tasks. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2015.
- [32] P. Viola and M. Jones. Robust real-time face detection. *Int'l J. Computer Vision*, 57(2):137–154, 2004.
- [33] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013.
- [34] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Convolutional channel features. *Proc. IEEE Int'l Conf. Computer Vision*, 2015.
- [35] C. Zhang and P. Viola. Multiple-instance pruning for learning efficient cascade detectors. *Proc. Advances in Neural Information Processing Systems*, 2007.
- [36] S. Zhang, C. Bauckhage, and A. B. Cremers. Informed haar-like features improve pedestrian detection. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2014.
- [37] S. Zhang, R. Benenson, and B. Schiele. Filtered channel features for pedestrian detection. *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2015.