



# A novel pixel neighborhood differential statistic feature for pedestrian and face detection

Jifeng Shen<sup>a,\*</sup>, Xin Zuo<sup>b</sup>, Jun Li<sup>c</sup>, Wankou Yang<sup>c</sup>, Haibin Ling<sup>d</sup>

<sup>a</sup> School of Electronic and Informatics Engineering, Jiangsu University, Zhenjiang, Jiangsu, 212013, China

<sup>b</sup> School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu, 212003, China

<sup>c</sup> School of Automation, Southeast University, Nanjing, Jiangsu, 210096, China

<sup>d</sup> Dept. of Computer & Information Sciences, Temple University, Philadelphia, PA, 19122, USA

## ARTICLE INFO

### Keywords:

Pedestrian detection

Face detection

PDF

Neighborhood differential statistic patterns

## ABSTRACT

Motivated by the successful application of Local Binary Pattern (LBP), in this paper we propose a novel pixel neighborhood differential statistic feature for pedestrian and face detection based on the multiple channel maps. The calculation of LBP comprises of two steps, Pixel Differential Feature (PDF) calculation and PDF sign encoding. The PDF distills discriminative information of local region that can improve the performance of the pedestrian detector, but the encoding step degrades the performance due to the quantization error. Although PDF is more discriminative than original channel maps, it has a much higher dimension than the original feature maps, and consequently requiring large computation cost. To address this issue, the pixel neighborhood differential pattern is learned with both supervised and unsupervised learning methods, which allow discovering discriminative pixel differential patterns in local area and achieving state-of-the-art results. Specifically, our method firstly aggregates the image channel maps into cell maps with max pooling. Then, pixel neighborhood differential feature based on each channel cell maps are calculated which contributes to encoding discriminative information in each local area and benefits the performance improvements. In addition, we attempt to learn discriminative differential statistic patterns by using linear discriminative analysis (LDA) and principle component analysis (PCA) for further performance improvement. Two sets of experiments are conducted on pedestrian detection and face detection respectively. The INRIA, Caltech, and ETH datasets are used for pedestrian detection, and the FDDB and AFW datasets for multi-view face detection. The experimental results show that our method achieves superior performance in comparison with the state-of-the-arts while running at 20 fps for 480×640 images.

## 1. Introduction

As a representative for the object detection problem, human detection is an active topic in computer vision. It poses great challenges due to large variability in deformation, illumination and occlusion present in images. In recent years, a great amount of attention has been paid to real-time human detection in applications such as vehicle autonomous driving, video surveillance and human activity understanding.

Despite a large body of work devoted to human detection in the last decades, it is still an open problem. Existing state-of-the-art methods [1–3] can be roughly categorized into two groups, those using hand-crafted features [4] versus those using data-driven features [5,6]. The former comprises of a set of manually designed features that are suitable for human detection, such as HOG [7], LBP [8], color descriptor [9], ICF [4], ACF [10], LDCF [11] and IHF [12,13], etc.

By contrast, the latter learns discriminative feature from the training data using Deep Learning models [5] or by constructing a set of over-complete dictionaries to encode the human parts [6]. Albeit computationally effective, the hand-craft features are not optimized for the human detection tasks. By contrast, the learned deep feature using Deep Learning models, such as CNN [14] or DBN [15] is very discriminative, but demands high computational costs. In this paper, a new pixel neighborhood differential statistic feature is proposed to discover the discriminative local variation patterns for the human channel maps. Compared with state-of-the-arts, the proposed feature exhibits solid advantage in computation efficiency, training speed and discriminating power. In particular, it runs at 20 fps for pedestrian detection for images of size 480×640.

In this paper, we first revisit the LBP feature [16] and demonstrate the connection between our proposed feature and CNN model. Then we present our proposed method with four different filtering schemes,

\* Corresponding author.

which can be further improved to learn discriminative patterns in both supervised and unsupervised learning framework. The final evaluations of the proposed feature on the public benchmarks reveal that our method achieves state-of-the-art results for both pedestrian and face detection tasks.

The contribution of our work is four-fold as follow:

1. We investigate the connection between our methods and the CNN model and demonstrate that our method can be viewed as a simplified one-stage CNN which achieves state-of-the-art results.
2. We propose a multi-scale pixel neighborhood differential feature with four different filter schemes, which is aiming at mining the underlying discriminative information to obtain the intrinsic structure of the pedestrian.
3. We propose an unsupervised feature learning method to reduce the redundancy of the pixel local differential feature. Meanwhile, we discover discriminative differential statistic patterns for improving both the accuracy and efficiency of pedestrian detection.
4. We propose a supervised feature learning approach which produces compact and informative feature.

The rest of the paper is organized as follows. After reviewing the related work in Section 2, we elaborate our method in Section 3. Next we provide comprehensive experimental evaluation on public benchmarks of pedestrian and face detection in Section 4. The paper is finally concluded in Section 5.

## 2. Related work

### 2.1. Local Binary Feature (LBP)

LBP is a widely and successfully used descriptor for texture classification, face detection and recognition, etc. It encodes the difference between the referred center pixel and its surrounding neighborhood in a circular sequence manner, and represents the local image patch as a binary string. Roughly speaking, a LBP feature characterizes the local spatial structure of image, as formulated in Eq. (1).

$$f_r = \sum_{i=0}^{N-1} s(p_i - p_c)2^i, \quad s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (1)$$

where  $p_i$  is one of the  $N$  neighbor pixels around the center pixel  $p_c$ , on a circle with radius  $r$  or a square with side length  $r$ . LBP is built on point-wise coding of the image intensities in the local region and thus is sensitive to local texture variation. The computation of LBP feature mainly comprises of three steps (shown in Fig. 1). First, the Pixel Differential Feature (PDF)  $a$  for each reference pixel is calculated, which yields an 8 dimension PDF descriptor (Fig. 1(a)). Second, a non-linear gating function  $S(a)$  is employed for mapping the PDF descriptor to a binary vector  $x$  of the same size (Fig. 1(b)). Finally, all these binary string bits are merged into an integer  $b$  by encoding function  $C(x)$  (Fig. 1(c)).

Fig. 1(d) gives an example of generating a pattern for one pixel in a

$3 \times 3$  local area. In particular, the non-linear gating function and encoding function are respectively defined as  $S(a) = \begin{cases} 0, & a < 0 \\ 1, & a \geq 0 \end{cases}$ , and  $C(x) = w^T x$ , where  $w = [2^0, 2^1, \dots, 2^7]$ .

There are two key parameters in LBP computation, the stride between two different sampling pixels  $s$  and the pooling radius  $r$  for each pixel. In Fig. 2(a), the sampling stride  $s$  equals to 4 and in Fig. 2(b), the radius  $r$  equals to 2. Fig. 2(c) shows a multi-scale LBP with different radius  $r=1, 2, 3$  simultaneously. In each radius  $r$ , we only consider 8 different orientations for raw PDF calculation. For an image of size  $W \times H$ , LBP feature is extracted from  $(W - r) \times (H - r) / (s * s)$  different anchor pixels, where each one is in a block with  $3 \times 3$  pixels.

### 2.2. Revisiting LBP

In this section, we explore LBP feature from a different viewpoint. We treat LBP as similar to the structure of a one-stage CNN that consists of convolution, Rectification Linear Unit (ReLU) and pooling operations. The procedure of calculating LBP feature on a human image is shown in Fig. 3. Firstly, the input image is a color image with three RGB channels, which is performed convolution operation with a  $3 \times 3 \times 8$  filter bank. Each filter reflects the variation direction between referenced point and its surrounding points. Then, the filtered result is fed into a ReLU operation, which is used to cut off the negative values to zero. Next, a pooling step is applied to encode the positive values to 1 and 0 for others. This operation only keeps the sign of the filtered result. Finally, the binary codes are transformed into integers by a convolution operation. The filter in the last stage is an  $8 \times 1$  vector  $[2^0, 2^1, \dots, 2^7]^T$ . The LBP can be treated as a swallow CNN with only one-layer and no Fully Connected (FC) layers. Besides, the conventional LBP feature significantly differs from this swallow CNN in its fixed weights in each stage. It does not involve in any model training with BP algorithms.

## 3. Our method

Our method is motivated by the close correlation between the CNN model and state-of-the-art hand-crafted features, e.g., SIFT, HOG and LBP. These hand-crafted features can be used as input for the CNN model at the first layer and the following operations can be concatenated to them which lead to much deeper structure. In our implementation, we use ten channel features as input, which include the LUV color channels, six gradient orientation channels and a magnitude channel. It is observed that the superior performance for almost all the hand-craft features can be achieved by simply adding one layer CNN operations based on the channel maps. In the following section, we introduce four different filter banks which are applied to the channel feature maps.

### 3.1. Pixel neighborhood differential feature

#### 3.1.1. Local Pixel Differential Features (LPDF)

The LBP feature allows encoding the local structure in the image

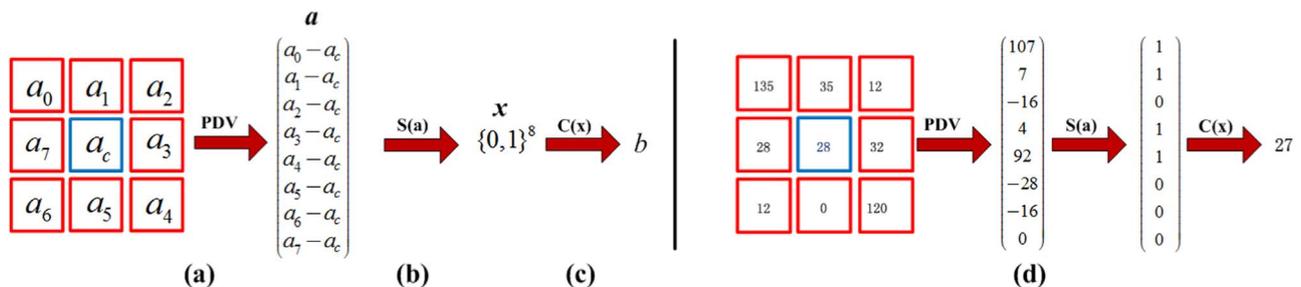


Fig. 1. PDF and LBP computation procedure. ((a–c) Computation procedure of LBP; (d) A toy sample for a single pixel).

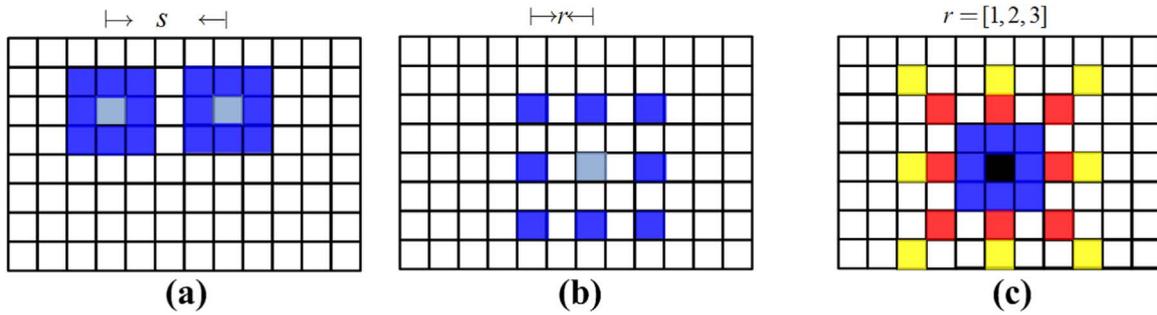


Fig. 2. Stride of LBP, radius of LBP and LBP with three different radius.

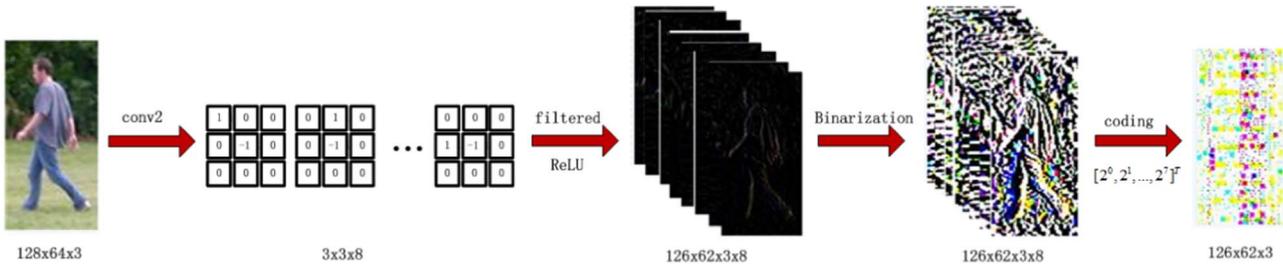


Fig. 3. different view of LBP operator.

with sufficient discriminating power to distinguish between different but similar image patches. Inspired by the success of the LBP, we impose raw LBP on the channel maps and observe significant performance drop (average miss rate about 5% higher than ACF on the INRIA dataset). With a deep insight into the LBP generation process, only partial order information is adopted and the pixel values of the filtered channel map is discarded in the last two steps (linearization and coding steps) of LBP computation. By removing the last two steps and using the filtered channel maps as our local differential features, the performance gain is reported with an average miss rate about 2% lower than ACF on the INRIA dataset, which indicates the partial order information insufficiently encodes the discriminative information of the pedestrians.

As shown in Fig. 4, the procedure of generating local differential feature is analogous to the CNN structure comprising a series of pooling and convolution steps. Nevertheless, the filter in the CNN is retrained many times for deriving the optimal weights. In our cases, the filter is fixed with simple horizontal, vertical and diagonal differential filters. As the filter bank in CNN plays a critical role in the detection performance, they should be carefully chosen and designed in our scenarios. Toward this end, we present following three filtering methods which aim to discover useful discriminative information for pedestrian detection.

### 3.1.2. Symmetrical Pixel Differential Features (SPDF)

As shown in Fig. 5, the filter bank of LPDF only uses the eight differential values of the surrounding eight pixels around its center pixel. However, in terms of the average channel map of pedestrians, it

is readily observed that the 45 and 135 degree slopes at the left and right shoulders are very discriminative areas. Therefore, more discriminative information is encoded in the symmetric pixel differential feature in the horizontal, vertical and diagonal directions across the center pixel, which we term SPDF for short. Fig. 6 gives the procedure of calculating SPDF. It is shown that the 12-dimension SPDF encodes additional symmetric information across the center pixel, compared with the 8-dimensional LPDF descriptor.

### 3.1.3. Circular Pixel Differential Features (CPDF)

In addition to the above mentioned feature, we also consider a circular pixel differential feature, which captures the information of surrounding pixels such as human head. As shown in Fig. 7, with the same size of LPDF, CPDF focus on its left-right or top-down positioned neighbor pixels instead of the center pixel.

### 3.1.4. Total Pixel differential Features (TPDF)

By comparing the three aforementioned features, one can see that none of them captures the pairwise relationship among arbitrary two pixels in a local area. Thus motivated, we propose the Total Pixel differential Features (TPDF) that generalizes the pixel differential feature. Specifically, when  $r = 1, 2, 3$  the dimensions  $K$  of the TPDF are computed as  $C_9^2 = 36$ ,  $C_{25}^2 = 300$  and  $C_{47}^2 = 1170$  respectively. Taking into account all the scenarios described above, the three features can be viewed as the special case of TPDF. In this paper, we only consider two scenarios with radius  $r = 1, 2$  for the sake of affordable memory size.

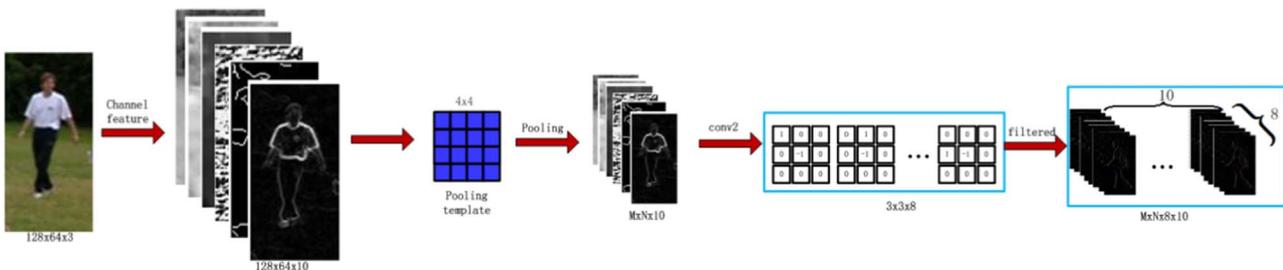


Fig. 4. Procedure of calculating local differential features.

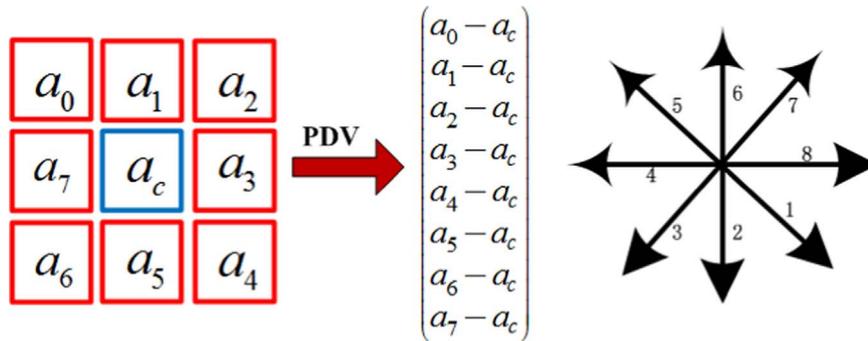


Fig. 5. Local Pixel Differential Features.

3.1.5. Comparison of different top pixel neighborhood differential feature

In our comparative study, we evaluate four features trained and tested on the INRIA dataset. Besides, different filter banks are learned based on the AdaBoost learning framework. Fig. 8 demonstrates the four top PDF features corresponding to the depth-2 decision tree classifier selected by the AdaBoost algorithms. The top left corner in each window shows the corresponding channel index, and the top left corner of the inner black rectangle in this window indicates the bin index of PDF based feature accordingly. The red and green boxes indicate the position of pixels in the corresponding local area of the channel map. It is observed that the most discriminative channel includes 1, 2, 4, 5 and 6 respectively corresponding to the LU channel of LUV color space, gradient magnitude channel and 90 degree gradient orientation channel.

As shown in Fig. 8, the most discriminative areas focus on the shoulders, legs and feet which agree with our human perception. In terms of LPDF, SPDF and CPDF, the first selected features are mainly distributed around the feet area in the channel 5. By contrast, their TPDF counterparts transferred to the head area in the channel 1. This can be explained by more discriminative information encoded in the TPDF than the other ones.

3.1.6. Explanation on the effect of our feature

In order to demonstrate the fundamental rational behind these pixel differential feature, we visualize the average filtering results of LPDF on the training data due to its relatively low dimension, as shown in Fig. 9. The left column with red bounding box indicates the average channel map on the positive training data, while the right eight columns offer the filtering result for each filter. As can be seen in Fig. 9, the details of the human shape are highlighted, which implies that our proposed feature can capture more discriminative information for the pedestrians. It apparently works in the magnitude and gradient angle channels. Meanwhile, our approach can also be seemed as a feature space transform which is used for mapping the original feature

space to a more readily separable space.

3.2. Pixel Differential Pattern Learning

In the previous section, we have introduced four different PDFs which can all be trained in the AdaBoost learning framework. Apparently, there're only a small number of variations in each local area, to reflect the structure of the image. Therefore, a natural idea arises to reduce the dimension of the PDF. In this paper, we aim to learn discriminative PDF patterns in both the unsupervised and supervised learning framework, which will be illustrated in the following section.

3.2.1. Unsupervised PDF pattern learning

Considering the local variation in a specified position, it is a standard practice to employ PCA for obtaining the major variation in the feature space. Suppose  $b$  is the number of blocks in a detection window,  $n$  is the number of scales,  $C$  is the number of channels for the feature map ( $C=10$  in this paper),  $K$  is the dimension of PDF, then the dimension of this multi-scale local differential channel map is  $K \times n \times b \times C$ .

In this paper, two different methods are exploited for dimension reduction, which are shown in Fig. 10. One performs dimension reduction on each channel separately (Fig. 10(b)), while the other one operates on all the channels as a whole (Fig. 10(a)). The former one aims to reduce the feature dimension from  $K \times n \times b \times C$  to  $d \times b \times C$  for each channel and each pixel, which we name pixel-wise PCA (Pi-PCA). The latter one is to reduce the dimension of  $K \times n \times b \times C$  to  $d \times b$  for all pixels with  $C$  channel features concatenated altogether, which we coin channel-wise PCA (Ch-PCA). It reveals that Pi-PCA contributes to the performance boost for the detector due to the transformation of original feature space to new orthogonal spaces found by the PCA. For example, the dimension of LPDF  $K$  for pedestrian equals 8. As shown in Fig. 10,  $b = W \times H$ ,  $D'=D \times C$ , where  $W=14$ ,  $H=30$ ,  $D=K$ ,  $C=10$ ,  $d=2$  and  $d'=20$ .

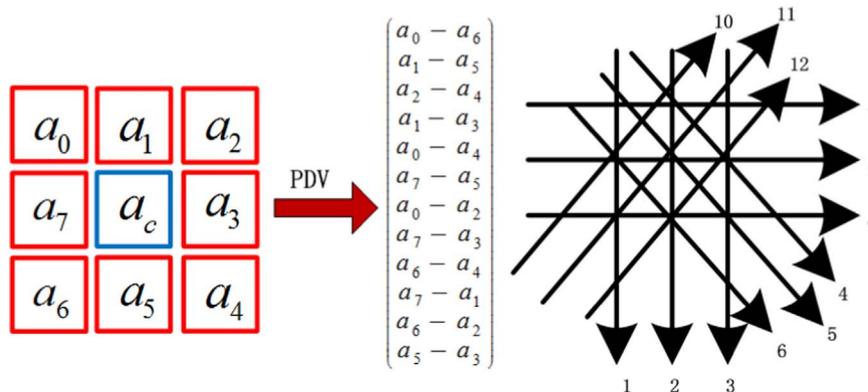


Fig. 6. Symmetrical local differential feature.

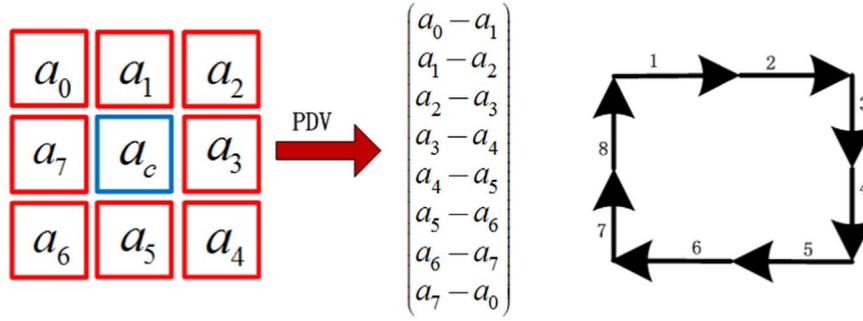


Fig. 7. Circular local differential feature.

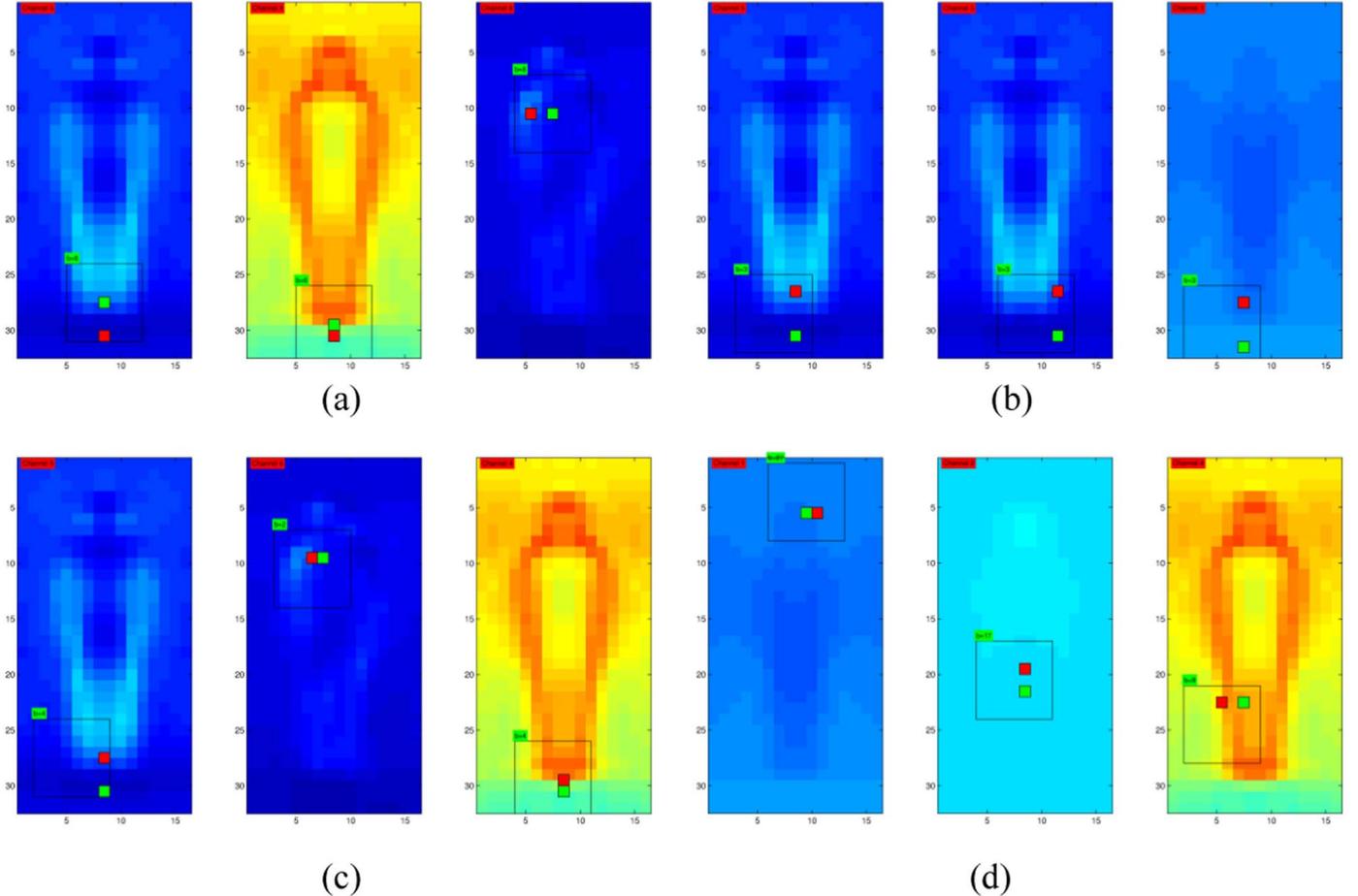


Fig. 8. Top one selected pixel neighborhood differential feature. ((a) Top left three windows for LPDF, (b) Top right three windows for SPDF, (c) Bottom left three windows for CPDF, (d) Bottom right three windows for TPDF).

Suppose there are  $N$  training images  $\{I_n\}_{n=1}^N$ ,  $I_n \in R^{w \times h \times 3}$ , and the channel maps for the training image are represented as  $\{M_n\}_{n=1}^N$ ,  $M_n \in R^{C \times W \times H}$ , where  $w$ ,  $h$ ,  $C$ ,  $W$  and  $H$  are the width and height of raw input image, the number of channels, the width and height of channel map respectively. For each channel map, we extract local differential vector with radius  $r$ , and the result is denoted as  $\bar{x}_{i,j} \in R^{KC}$ , where  $K=8$ ,  $C=10$  for LPDF. The  $\bar{x}_{i,j}$  is a  $KC$  dimension local differential vector, where  $i$  is the index of the training image,  $j$  is the index of blocks in an image and there're  $b$  local differential vectors totally in one image. So  $i^{th}$  image can be represented as a matrix  $\bar{X}_i = [\bar{x}_{i,1}, \bar{x}_{i,2}, \dots, \bar{x}_{i,b}]^T$ . Thus, the training feature are obtained by concatenating all the  $N$  input images together, which can be formulated in Eq. (2)

$$X = [\bar{X}_1, \bar{X}_2, \dots, \bar{X}_N] \in R^{KC \times Nb} \quad (2)$$

By only considering the  $L$  filters of the original feature, PCA minimizes the reconstruction error with top  $L$  eigenvectors of the

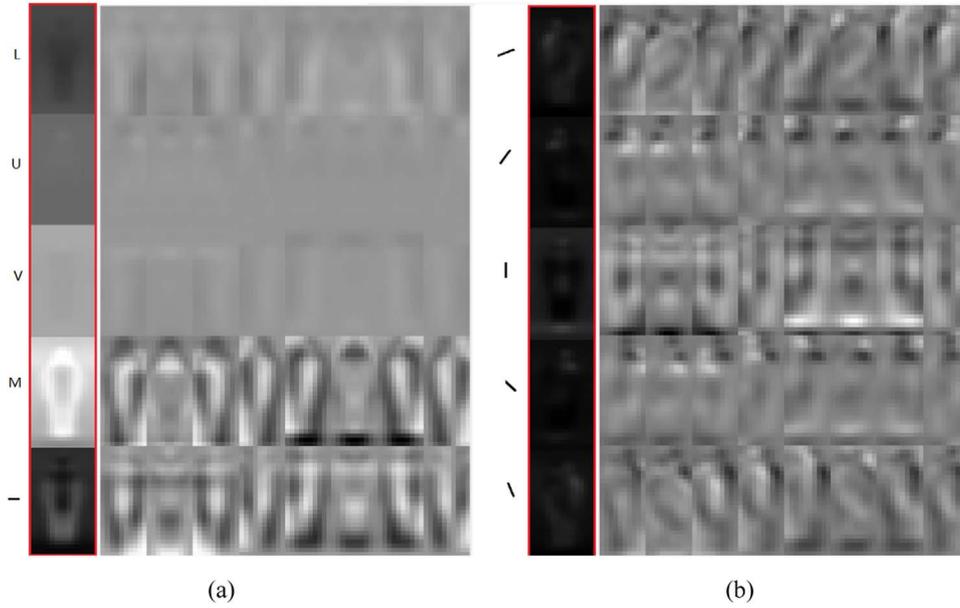
covariance matrix  $XX^T$ , which is formulated in Eq. (3).

$$\min_{V \in R^{KC \times L}} \|X - VV^T X\|_F^2, \text{ s. t. } V^T V = I_L \quad (3)$$

The solution to Eq. (3) is the  $L$  eigenvectors corresponding to the top  $L$  eigenvalues of matrix  $XX^T$ , which can be easily calculated by the SVD decomposition. The detailed numerical comparisons on the datasets are shown in Section 4.

### 3.2.2. Supervised PDF pattern learning

In this section, we will make use of the labels of the training data to discover discriminative patterns for pedestrian detection. In order to distinguish pedestrian from the negative patches, the conventional method consists in finding an optimal projection vector for each pixel coordinate, such that the high inter-class variance and low inner-class variance are obtained simultaneously. Mathematically, it is formulated



**Fig. 9.** Average channel map and average LPDF filtering results for the training data. (First columns of Fig. 9(a–b) with red box represent the 10 average channel map; Eight right columns of Fig. 9(a–b) represent the 8 average LPDF in each channel). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

in Eq. (4).

$$w_f = \arg \max_w \frac{w^T S_b w}{w^T S_w w} \quad (4)$$

where  $S_b$  is the between-class scatter matrix and  $S_w$  is the within-class scatter matrix. The solution of Eq. (4) can be written in a closed form  $w^* = S_w^{-1}(m_1 - m_2)$ , where  $m_1$  and  $m_2$  is the mean of the positive and negative PDF respectively. As shown in Fig. 11, there also exist two different learning strategies similar to the approach introduced in Section 3.2.1, i.e., pixel-wise LDA (Pi-LDA) and channel-wise LDA (Ch-LDA). The difference between them is the resulting feature dimensionalities, which are  $W \times H$  and  $W \times H \times C$  for Ch-LDA and Pi-LDA respectively. In addition, it is observed that this Pi-LDA based feature significantly profits the performance gains of the detector. In the experiments, we will conduct comparative study for comparing these features.

## 4. Experiment

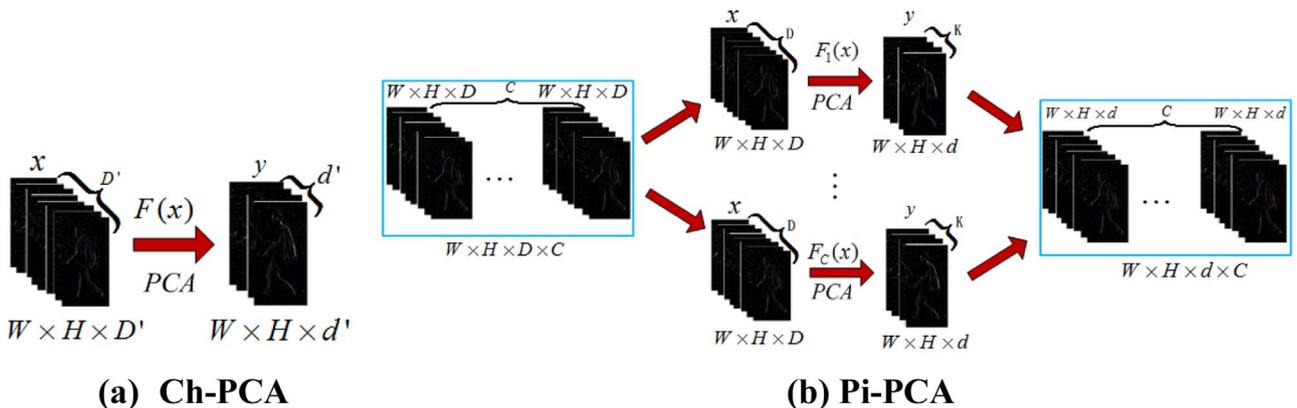
### 4.1. Experiment setting

In order to validate the effectiveness of our method, we carry out extensive experiments on three public datasets for pedestrian detection

task and on two datasets for multi-view face detection task, which are briefly described in Table 1 and 2.

The detailed experiment setting for pedestrian detection is described as follows. The size of the pedestrian window is set to  $128 \times 64$ , and each positive sample is cropped from the annotated image. Each annotation of pedestrian is jittered to mitigate the misalignment problem. The total number of positive samples is about 24,740 for the INRIA dataset and 24,498 for the Caltech dataset. The pooling template size is of  $4 \times 4$  pixels, which shrinks the original channel maps (size  $128 \times 64 \times 10$ ) into pooled channel maps (size  $32 \times 16 \times 10$ ). We make use of AdaBoost algorithm to perform feature selection, with depth-2 decision tree as weak classifiers. The AdaBoost training is conducted by five rounds (32, 128, 512, 2048, 4096), each of which is trained with 10,000 negatives which are bootstrapped from a large negative pools. We use public available Piotr's toolbox [17] to calculate the channel features and utilize the evaluation code [17] to evaluate the detector.

Following the setting in [17], our multi-view face detector is also trained on the AFLW database. The size of the face window is  $80 \times 80$ . We have trained six face detectors according to different yaw angles which is divided into  $[-90, -60]$ ,  $[-60, -30]$ ,  $[-30, 0]$ ,  $[0, 30]$ ,  $[30, 60]$ ,  $[60, 90]$  respectively. The pitch and roll angle is limited to  $[-22.5, 22.5]$ . The number of positive training samples for each detector is 3726,



**Fig. 10.** Feature dimension reduction with PCA in two different methods.

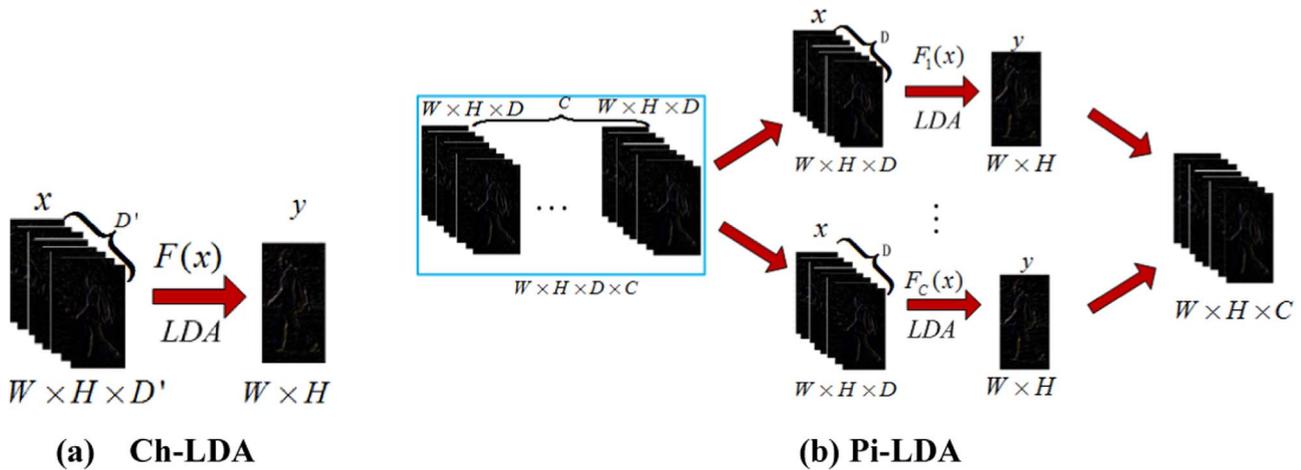


Fig. 11. Feature information composition with LDA in two different methods.

Table 1  
Datasets for pedestrian detection.

Dataset name	Description
INRIA	<a href="http://pascal.inrialpes.fr/data/human/">http://pascal.inrialpes.fr/data/human/</a> Training data: 2416 human annotations in 614 and 1218 non-human images Testing data: 1132 human annotations in 288 and 453 non-human images
Caltech-USA	<a href="http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/">http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/</a> Training data: 6325 pedestrian annotations in 4250 images (set00–set05) Testing data: 5051 pedestrian annotations in 4024 images (set06–set10)
ETH	<a href="http://www.vision.ee.ethz.ch/~aess/dataset/">http://www.vision.ee.ethz.ch/~aess/dataset/</a> Training data: 3780 pedestrian annotations in 853 images Testing data: 14,167 pedestrian annotations in 1804 images

Table 2  
Datasets for multi-view face detection.

Dataset name	Description
AFLW	<a href="http://lrs.icg.tugraz.at/research/aflw/">http://lrs.icg.tugraz.at/research/aflw/</a> Training data: about 25 k annotated faces in real-world images
FDDB	<a href="http://vis-www.cs.umass.edu/fddb/">http://vis-www.cs.umass.edu/fddb/</a> Testing data: 5171 annotated faces in 2845 images
AFW	<a href="http://www.ics.uci.edu/~xzhu/face/AFW.zip">http://www.ics.uci.edu/~xzhu/face/AFW.zip</a> Testing data: 468 annotated faces in 205 images

4024, 4636, 5069, 4024, and 3726 according to the yaw angles. The detector is trained with AdaBoost algorithm with four rounds and the final detector comprises 2048 weak classifiers. A total number of 4881 negative images without any faces for bootstrapping are collected from VOC 2007 dataset.

#### 4.2. Comparison of PDF with different parameters

In order to derive the optimal parameter of stride(S) and radius(R), we have conducted a set of experiments on the INRIA dataset. As can be seen from Fig. 12, we can see that, the optimal parameter for the LPDF is achieved when the stride equals to 1(S=1) and radius of the encode area is 3(R=3). It is observed that this parameter configuration is also best for the other two features. Besides, the SPDF is superior to

other two features, with the lowest miss rate of 14.46%. So we can infer that symmetric information helps to improve the performance of pedestrian detection.

#### 4.3. Analysis of learned filters with PCA and LDA

In order to get insight from the learned weights, we visualize the top features which are selected by our classifiers. We have test four different learning modes (Ch-PCA, Pi-PCA, Ch-LDA and Pi-LDA), which is shown in Fig. 13(a–d) respectively. The channel-wise feature learning means the filter is learned across channels that only depend on the position of the pixels, whereas the pixel-wise filter learning means it is learned on each channel of the training samples independently.

As shown in Fig. 13(a), we can see that the first learned eigenvector of Ch-PCA(first column in Fig. 13(a)) has large absolute weight values on the 1st, 3rd, 5th and 7th dimension of PDF corresponding to the four diagonal directions (shown in Fig. 5). The second and third eigenvector of the filter corresponds to the 8th and 6th dimension of PDF (largest weight), which represent the horizontal and vertical directions. The position of the corresponding PDF is shown in Fig. 13(e), from which we observe that PCA can distill orthogonal directions for local differentiation patterns (corresponding PDF is located in the left shoulder in Fig. 13(e)).

We have also demonstrated the top filter learned by Pi-PCA. The position of the learned filter is shown in Fig. 13(f) and the weight of the filter is shown in Fig. 13(b). We can see that it is quite different from the filters in Fig. 13(a) which learn all the channels as a whole. In the top eigenvector (first column of Fig. 13(b)), we observed that the largest absolute weight values focus on the direction 5th, 6th, 7th and 8th dimension of the PDF (corresponding PDF located in the bottom of the left leg in Fig. 13(f)).

As shown in Fig. 13(c), the Ch-LDA filter put more weights on the 3th and 5th dimension of PDF in the 8th channel map. This is also consistent with the feature selected in Fig. 13(g), which suggests the largest contract values are presented in the orientation of 45 and 135°.

Fig. 13(d) gives the Pi-LDA filter in the 5th channel, while Fig. 13(h) provides the PDF in the 5th channel maps which located in the region of legs. It is easy to observe that this region seems to be much brighter than other surrounding regions. As can be seen in Fig. 13(d), the largest two weights are in the 5th and 6th dimension of the PDF.

In Fig. 14, we observe that Pi-LDA and Pi-PCA enable promoting the performance of the original TPDF features. However, the Ch-LDA and Ch-PCA degrade the performance of the detector. It is also shown that TPDF exhibits superior performance over LDF, CPDF and SPDF, all of which works better than ACF feature on which our feature is built. Therefore, we only consider pixel-wise scenario for feature learning in

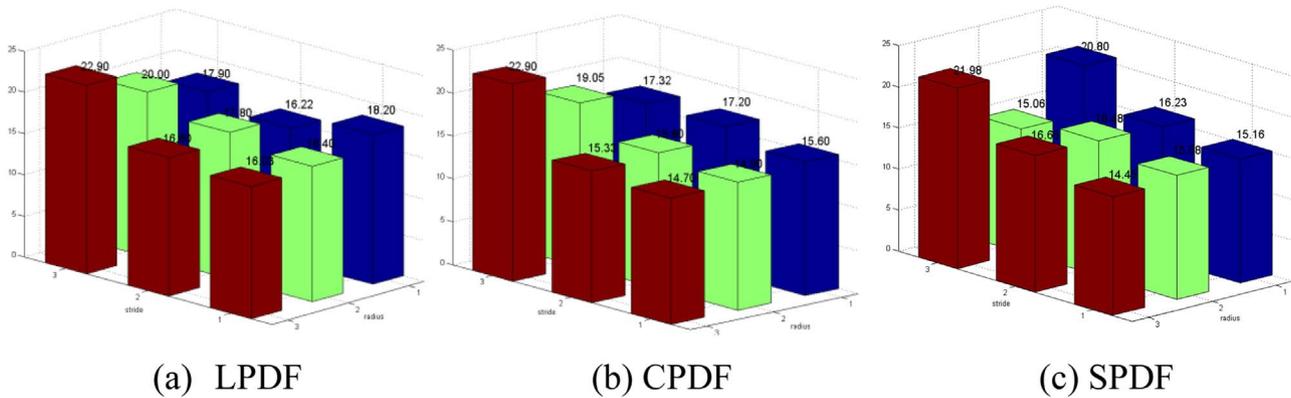


Fig. 12. Optimal parameter selection for LPDF, CPDF and SPDF.

the following section. Besides, we also find that Pi-LDA performs slightly better than Pi-PCA method.

#### 4.4. Comparisons with state-of-the-art algorithms for pedestrian detection

##### 4.4.1. Evaluation on the INRIA dataset

In this paper, we compare our proposed method with the state-of-the-art methods. We use public available Piotr's toolbox [3] for evaluation. The state-of-the-art methods include the ACF [10], LDCF [11] and IHF [12] which are closely related to our proposed method. Other methods in our comparative study include ConvNet [5], Sketch Tokens [18], and two baseline methods (VJ [19] and HOG [7]). The comparison results are shown in Fig. 15. It can be seen that, our method achieves state-of-the-art result at a miss rate approximately 4.7% lower than ACF and 0.8% lower than LDCF, which substantially suggests that local neighbor pixel differential information is very discriminative in channel maps.

##### 4.4.2. Experiments on the Caltech dataset

We also evaluate our method with extensive comparative study on the Caltech dataset. Fig. 16 offers the performance of different methods. Similar to the results achieved on the INRIA dataset, our method outperforms all the hand-crafted features and reveals the consistent superiority on the Caltech dataset. In particular, our method significantly outperforms ACF by reporting a 5% lower average miss rate. Besides, the marginal superiority is also observed against LDCF. In addition to the detection accuracy, our method is also advantageous in real-time running and fast training speed.

##### 4.4.3. Experiments on ETH dataset

Due to involving dramatic variance in human scale ranging from  $13 \times 25$  to  $239 \times 478$ , two octaves of the test image in this dataset is upscaled to detect the smaller pedestrian. Fig. 17 provides the experimental results of different methods on ETH dataset. It demonstrates our method considerably outperforms ACF and LDCF (10% lower than ACF and 4.7% lower than LDCF). Interestingly, our method even achieves comparable performance with the best result thus far which is obtained by deep models. This can be explained by the higher images resolution and smaller pedestrian pose variation on ETH dataset. Consequently, the local neighbor differential feature on the channel maps is capable of describing the shape of the pedestrians more preferably.

#### 4.5. Comparisons with state-of-the-art algorithms for face detection

##### 4.5.1. Experiments on FDDB dataset

In this section, we also evaluate our method on the FDDB dataset for face detection task (Fig. 18). We follow the evaluation protocol in

[20] and use the average discrete ROC and continuous ROC as the performance measure. In the comparative study, we compare our method with seven state-of-the-art approaches. On par with the best hand-crafted feature [17], our method performs comparable to the ACF-multiscale for multi-view face detection.

The Yan's [21] and the HeadHunter [22] methods make use of DPM to model the variation of face units such as eyes and nose that can get better detection results. The other three [23–25] methods learn the discriminative features based on the Deep Learning methods. However, our method is much efficient to train and also achieves competitive results.

##### 4.5.2. Experiments on the AFW dataset

The experimental results on the AFW dataset are shown in Fig. 19(a). Our method achieves average precision of 92.45 which is 5% lower than the best HeadHunter method (Fig. 19(b), Fig. 11 from [22]). In this experiment, we cannot reproduce the result of ACF-multiscale [15], which is still 4.4% lower than the published result due probably to slightly different number of training data reported in the paper. But in our experimental setting, our TPDF method is superior to our implement of the ACF method.

#### 4.6. Runtime comparison

All the experiments are implemented with Matlab R2014b and visual studio 2012 on DELL Precision T7610 workstation (16 core dual CPU E5-2650, 2.6GHZ, 64G). It takes about 6 h to train a four-stage pedestrian detector with approximately 20 fps detection speed for a  $640 \times 480$  image. The comparison results are shown in Table 3. Average reject number [26] is also utilized to evaluate the speed of detectors with different features, which is shown in Table 3. Note that our detector can be further accelerated with GPU or other parallel computing techniques.

## 5. Conclusion

We propose a novel pixel neighborhood differential statistic feature which makes use of the differential statistic information of local pixel and its neighborhood to detect pedestrians effectively and efficiently. Our proposed method takes into account the local differential information instead of the only partial order for the LBP, which significantly contributes to the performance gains. Experimental results for pedestrian detection based on the INRIA, Caltech and ETH datasets, and face detection based on the FDDB and AFW datasets show that our method can achieve state-of-the-art results. Besides that, our method consistently outperforms other competing hand-crafted features for pedestrian detection. Meanwhile, our method can be readily and efficiently implemented with a real-time running speed for  $640 \times 480$  images.



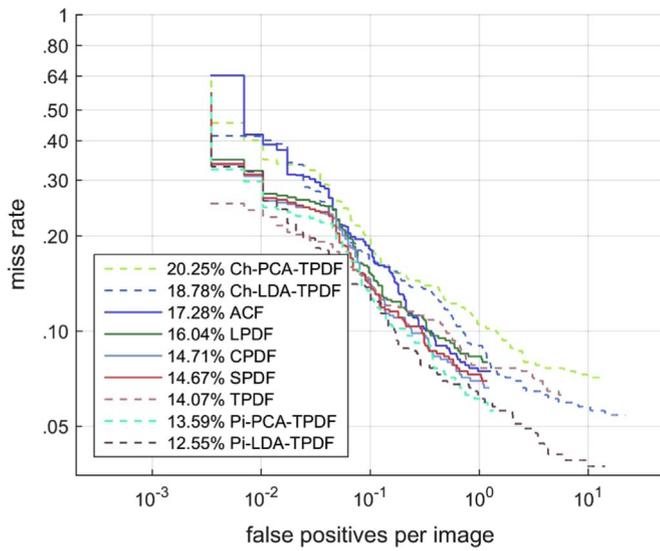


Fig. 14. Comparison between PDF and filtered feature on INRIA dataset.

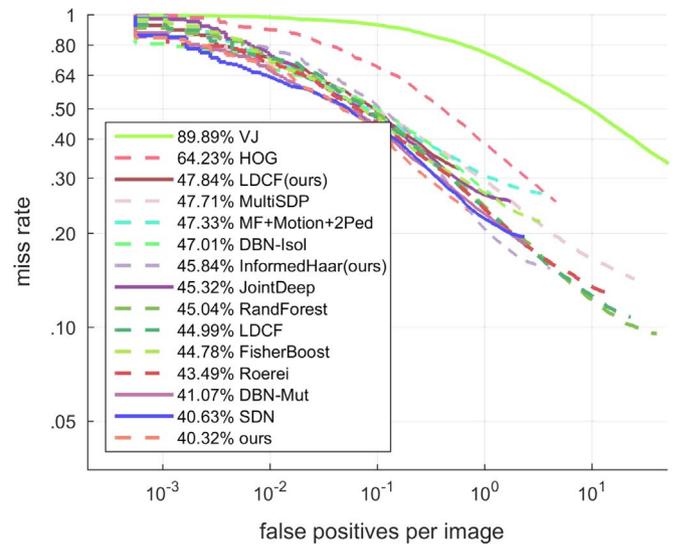


Fig. 17. Results on ETH dataset.

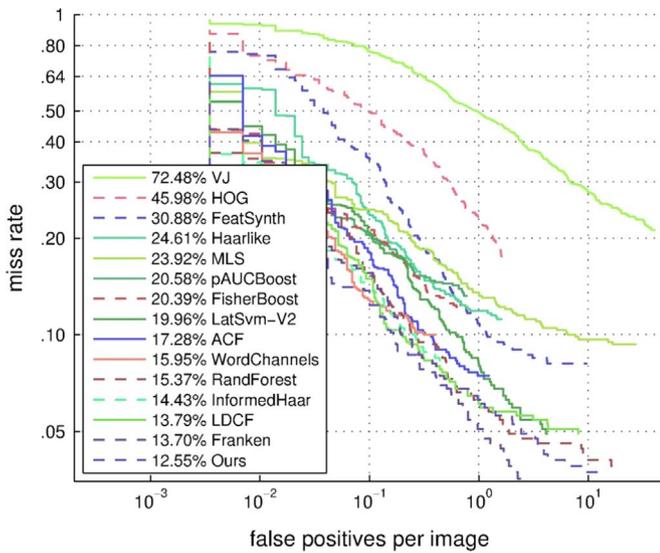


Fig. 15. Results on INRIA dataset.

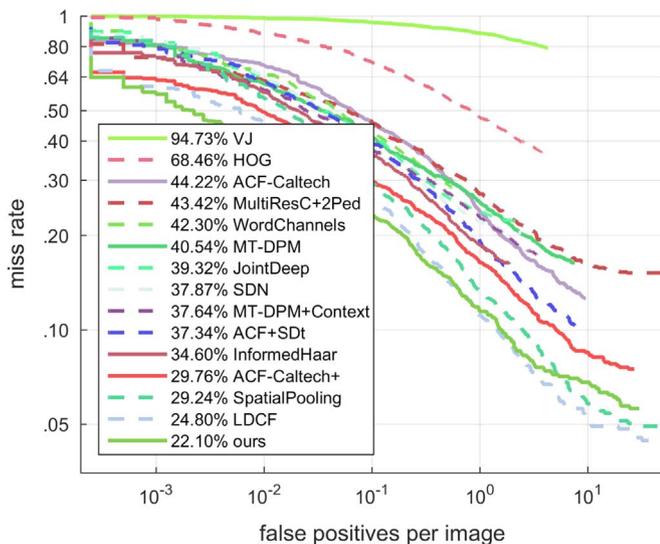
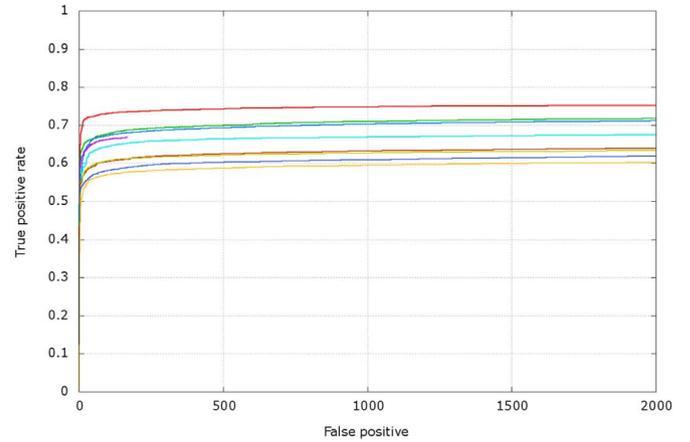
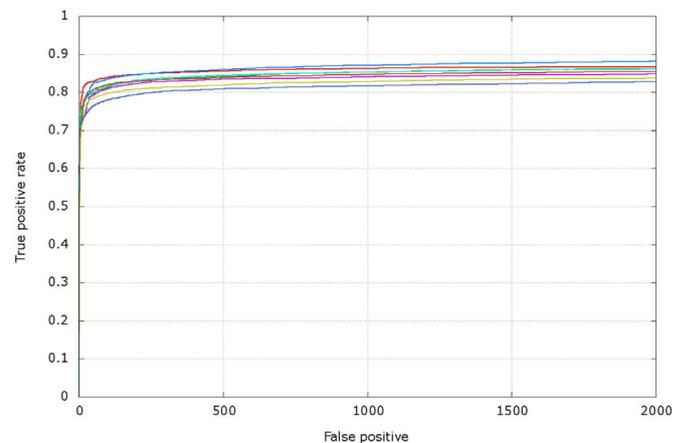


Fig. 16. Results on Caltech dataset.



(a) Continues score



(b) Discrete score

Fig. 18. Comparison with the discrete and continues score on Fddb dataset.

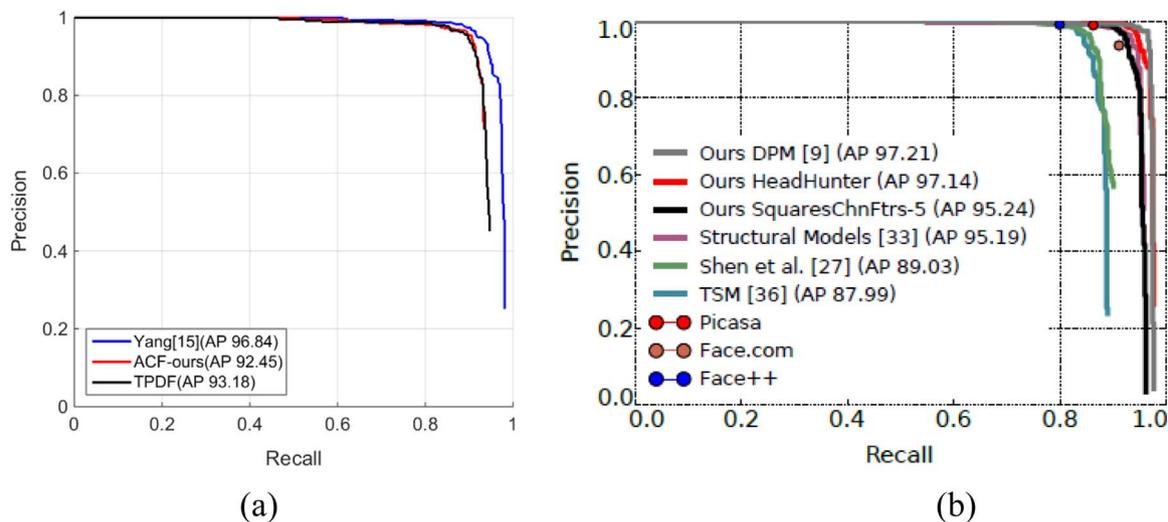


Fig. 19. Result on the AFW dataset.

Table 3  
Runtime comparison.

Name	Running speed (fps)	Average reject number
ACF	30	5.24
TPDF	20	3.47

**Acknowledgments**

This project is supported by the NSF of Jiangsu Province (Grants No. BK20140566, BK20150470, BK20130471), the NSF of China (61473086, 61305058), the Fundamental Research Funds for the Jiangsu University (13JDG093), the NSF of the Jiangsu Higher Education Institutes of China (15KJB520008), and the China Postdoctoral science Foundation (2014M561586).

**References**

[1] S. Munder, D.M. Gavrilu, An experimental study on pedestrian classification, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (11) (2006) 1863–1868.  
 [2] D. Geronimo, A.M. Lopez, A.D. Sappa, Survey of pedestrian detection for advanced driver assistance systems, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (7) (2010) 1239–1258.  
 [3] P. Dollár, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: an evaluation of the state of the art, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (4) (2012) 743–761.  
 [4] P. Dollar, Z. Tu, P. Perona, S. Belongie, Integral Channel Features, *British Machine Vision Conference 2009*, London, England, 2009.  
 [5] P. Sermanet, K. Kavukcuoglu, S. Chintala, Y. LeCun, Pedestrian detection with unsupervised multi-stage feature learning, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3626–3633.  
 [6] C. Yao, X. Bai, W. Liu, L.J. Latecki, Human detection using learned part alphabet and pose dictionary, in: *Proceedings of the 13th European Conference on Computer Vision*, 2014, pp. 251–266.  
 [7] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.  
 [8] X. Wang, T. Han, S. Yan, An HOG-LBP Human detector with partial occlusion handling, in: *Proceedings of the IEEE International Conference on Computer Vision*, 2009.  
 [9] K. van de Sande, T. Gevers, C. Snoek, Evaluating color descriptors for object and scene recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (9) (2010) 1582–1596.  
 [10] P. Dollár, R. Appel, S. Belongie, P. Perona, Fast feature pyramids for object detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 36 (8) (2014) 1532–1545.  
 [11] W. Nam, P. Dollar, J.H. Han, Local decorrelation for improved pedestrian detection, *Adv. Neural Inf. Process. Syst.* 27 (2014) 424–432.  
 [12] S. Zhang, C. Bauckhage, A.B. Cremers, Informed haar-like features improve pedestrian detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.  
 [13] S. Zhang, R. Benenson, B. Schiele, Filtered channel features for pedestrian detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1751–1760.

[14] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE* 86 (11) (1998) 2278–2324.  
 [15] G.E. Hinton, S. Osindero, Y. Teh, A fast learning algorithm for deep belief nets, *Neural Comput.* (2006) (2006) 18.  
 [16] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (7) (2002) 971–987.  
 [17] B. Yang, J. Yan, Z. Lei, Stan Z. Li, Aggregated channel features for multi-view face detection, in: *Proceedings of the IEEE International Joint Conference on Biometrics*, 2014, pp. 1–8.  
 [18] J.J. Lim, C.L. Zitnick, P. Dollar, A learned mid-level representation for contour and object detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.  
 [19] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.  
 [20] V. Jain, E. Learned-Miller, FDDB: A benchmark for face detection in unconstrained settings, in: *Technical Report UM-CS-2010-0092010*, Dept. of Computer Science, University of Massachusetts.  
 [21] J. Yan, Z. Lei, L. Wen, S.Z. Li, The fastest deformable part model for object detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2497–2504  
 [22] M. Mathias, R. Benenson, M. Pedersoli, L. Van Gool, Face detection without bells and whistles, in: *Proceedings of the European Conference on Computer Vision*, 2014.  
 [23] H. Li, Z. Lin, X. Shen, J. Brandt, G. Hua, A convolutional neural network cascade for face detection, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.  
 [24] S.S. Farfadi, M. Saberian, Li-Jia Li, Multi-view face detection using deep convolutional neural networks, in: *Proceedings of the International Conference on Multimedia Retrieval*, 2015.  
 [25] B. Yang, J. Yan, Z. Lei, S.Z. Li, Convolutional channel features, in: *Proceedings of the IEEE International Conference on Computer Vision*, 2015.  
 [26] J. Shen, C. Sun, W. Yang, Z. Wang, Z. Sun, A novel distribution-based feature for rapid object detection, *Neurocomputing* 74 (17) (2011) 2767–2779.

**Jifeng Shen** received his M.S. and Ph.D degree in computer science from Jiangsu University of Science and Technology, Zhenjiang, China, and Southeast university, Nanjing, China, in 2006 and 2013 respectively. He is currently a faculty in School of Electrical and information Engineering at Jiangsu University, Zhenjiang, China since 2013. His research interests include object detection, scene classification and content-based image retrieval.

**XinZuo** received her B.S. and M.S. degree in the school of computer science from East China Shipbuilding Institute, and Jiangsu University of Science and Technology, Zhenjiang, China, in 2003 and 2007 respectively. She received her Ph.D. degree in the school of computer science and Engineering from Southeast University, Nanjing, in 2014. Her research interests include image retrieval and image registration.

**Jun Li** received the B.S. Degree in electrical engineering & automation from Nanjing Normal University, Nanjing, China, and the M.S. Degree in Control theory & engineering from Southeast University, Nanjing, China, in 2008 and 2011, respectively. He is currently working toward the Ph.D. Degree with School of Automation, Southeast University, Nanjing, China. His research interests include multimedia search and computer vision.

**Wankou Yang** received the B.S., M.S. and Ph.D degrees in the School of Computer Science and Technology, Nanjing University of Science and Technology (NUST), China,

respectively in 2002, 2004, and 2009. From July 2009 to Aug. 2011, he worked as a Postdoctoral Fellow in the School of Automation, Southeast University, China. From Aug. 2010 to Aug. 2011, he also worked as a Postdoctoral Fellow in Face Aging Group, UNC Wilmington, USA. Since Sep. 2011, he has been an assistant professor in School of Automation, Southeast University. His research interests include pattern recognition, computer vision and machine learning.

**Haibin Ling** received the BS and MS degrees from Peking University, China, in 1997 and 2000, respectively, and the Ph.D degree from the University of Maryland, College Park, in 2006. From 2006 to 2007, he worked as a postdoctoral scientist at UCLA. In 2008, he joined Temple University where he is now an associate professor. His research interests include computer vision and medical image analysis.