Research on Pedestrian Detection Algorithm Based on Weighted DPM Model

Weidong Li, Chunjiang Pang

School of North China Electric Power University, Baoding, Hebei 071003, China.

Abstract

With the rapid development of artificial intelligence, computer vision, etc., pedestrian detection technology in the people's life self-respecting the complex scene behind the 643 people, the occlusion between pedestrians has increased the difficulty of pedestrian detection. The DPM model algorithm has a good effect on pedestrian detection. Therefore, an improved method based on the traditional DPM model is proposed, which can compensate for the lack of image texture features in pedestrian detection. The basic idea is that the classic DPM model algorithm adopts the HOG feature. In this paper, we propose the fusion of HOG and LBP describing the texture features of the image and propose a weighted component model for the DPM model. Different weights can be attached to different components, which can distinguish different components and make the component model play a bigger role. The role. Then the detection results on the INRIA database show that the weighted pedestrian detection based on feature fusion improves the accuracy of the detection results while not affecting the detection speed, which provides a better basis for the pedestrians in the picture to do further analysis.

Keywords

Pedestrian Detection; HOG features; LBP features; DPM model; Weighed Part Model.

1. Introduction

The main purpose of pedestrian detection is to determine whether there is a pedestrian in the image to be detected, and if a pedestrian is detected, the specific location of the pedestrian is marked. At present, pedestrian detection has become an important research direction in the fields of automotive intelligent assisted driving, video surveillance, robot vision analysis, and human behavior analysis [1, 2]. The commonly used description features operators for pedestrian detection have a directional gradient histogram (HOG) [3, 4, 5], a local binary mode (LBP) [6], and a Haar feature [7], while human features are usually composed of underlying features. Features and deep learning features. The underlying features usually refer to the basic features of images such as texture features, color features, shape features, edge features, spatial relationship features, etc.; and combined features are combinations of multiple underlying features, or high-order statistical features of underlying features; depth the learning feature is a feature that is learned from the raw data of the image through deep learning. HOG is widely used in the field of pedestrian detection, which forms features by histograms of gradient directions of images. The DPM component model [8] is an improved algorithm for Felzenszwalb et al. to add a component model based on HOG. Because of the lack of image model in the image texture, an algorithm combining LBP and LOG features and weighting the classic DPM is proposed. The test results show that the pedestrian detection method can improve the accuracy of the detection results while not affecting the detection speed and has strong robustness to the environment and illumination.

2. Algorithm Theory

2.1 HOG Feature Extraction

HOG (Histogram of Oriented Gradient) refers to a directional gradient histogram, which is a feature descriptor used for object detection in the fields of image processing and machine vision. It consists of statistical pixel gradient direction histograms and image calculations. The HOG extraction feature flow is shown in Figure 1 below:

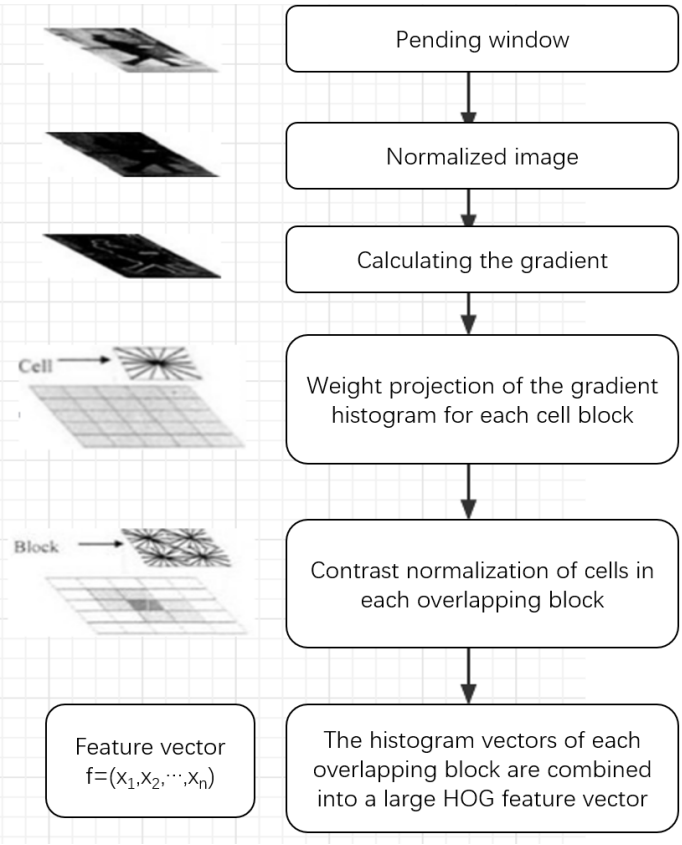


Fig 1. HOG feature extraction flow chart

(1) Standardized gamma space and color space

In order to reduce the influence of lighting and other factors, the input image is first normalized. Because color information is not very useful, it is usually first converted into a grayscale image. Gamma compression formula:

$$\mathrm{I}(\mathrm{x},\mathrm{y}) = \mathrm{I}(\mathrm{x},\mathrm{y})^{gamma} \tag{1}$$

Gamma=1/2

(2) Calculate the image gradient

The gradient of the pixel points (x, y) in the image is:

$$G_x(x,y) = H(x+1,y) - H(x-1,y) \tag{2}$$

$$G_y(x,y) = H(x,y+1) - H(x,y-1) \tag{3}$$

$$\mathrm{G}(\mathrm{x},\mathrm{y}) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{4}$$

$$\alpha(\mathrm{x},\mathrm{y}) = \tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \tag{5}$$

G_x (x, y) is the horizontal gradient at (x, y), G_y (x, y) is the vertical gradient at (x, y), and H (x, y) represents the pixel in the input image (x, y) pixel value, G (x, y) is the gradient magnitude of (x, y), and α (x, y) is the gradient of the point (x, y).

(3) Construct a gradient direction histogram

The image is divided into several cells. In this paper, the histograms of 9 bins are used to count the gradient information, that is, the gradient direction is 0-180 degrees divided into 9 intervals to obtain a 9-dimensional eigenvector. Then, using the trilinear interpolation method, the gradient direction is diffused to the adjacent bin of the cell in each block, and then the histogram is counted until all the histograms are connected in series to form a gradient direction histogram of the entire image. Finally, the characteristics of all the blocks are connected in series to obtain the characteristics of the human body. For example, for a 64*128 image, each 16*16 pixel constitutes a cell, and every 2*2 cell form a block. Since each cell has 9 features, there are 4*9= in each block. The 36 features, in steps of 8 pixels, will have 7 scan windows in the horizontal direction and 15 scan windows in the vertical direction. For example, a 64*128 image can be calculated to have a total of 36*7*15=3780 features.

2.2 LBP Feature Extraction

The LBP feature descriptor can express the texture features of the image well. It has the characteristics of scale, gray scale and rotation invariance. It has good robustness to complex background, illumination and occlusion, etc. It can solve HOG well. Difficulties in lighting, etc. The original LBP operator is calculated on the 3*3 window, and the gray value of the pixel at the center of the window is the threshold value, and then compared with the gray value of the adjacent 8 pixels, if the gray value of the surrounding pixels the position is marked as 1 if it is greater than the gray value of the center pixel, and 0 otherwise. Then, the eight binary numbers are grouped into an 8-bit binary number in a clockwise order, and then converted into a decimal number (usually converted to a decimal number or LBP code, a total of 256), the resulting decimal number is the 3*3 The LBP value of the center pixel of the window and use this value to reflect the texture information of the image in the area. as shown in picture 2.

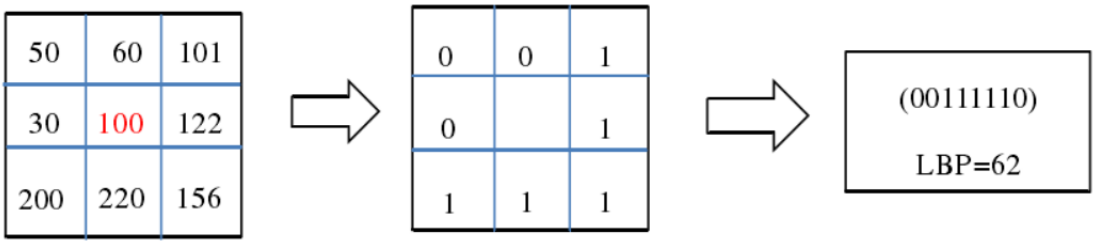


Fig 2. Schematic diagram of the LBP operator

(1) Circular LBP operator

It can be seen from the above steps that the traditional LBP operator is insufficient. It only covers a square area within a fixed radius and cannot be expanded. It obviously cannot meet the needs of textures of different sizes and frequencies. In order to adapt to different sizes of texture features, and to be able to achieve scale, rotation invariance and grayscale requirements, Ojala et al. improved the original local binary mode operator and replaced the original square neighborhood with a circular neighborhood. The improved LBP operator allows multiple sample points to be taken within a circular neighborhood of radius R. Therefore, several LBP operators containing P sample points as shown in Fig 3 are obtained.

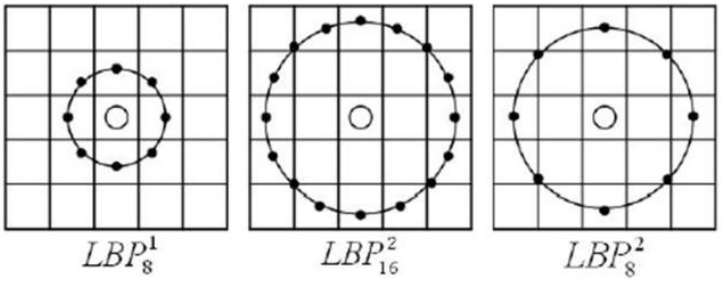


Fig 3. Several LBP operators

The formula for calculating the LBP operator is shown in equation (6).

$$LBP_{P,R} = \sum_{p=0}^{p-1} s(g_p - g_c)\, 2^p \tag{6}$$

$$s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x \leq 0 \end{cases} \quad (7)$$

Among them, LBPP, R means that there are P sampling points in the circular neighborhood with radius R, gp (p=0,1...p-1) is the gray value of the pixel in the circular neighborhood, and gc is the central pixel. Gray value.

(2) Equivalent mode LBP

In a circular area of radius R, the binary codes are arranged in different order, and features containing p sample points will produce 2P binary patterns. In order to improve its performance and reduce the dimension of the feature histogram, Ojala et al. [9] proposed a subset of the Uniform Pattern in LBP multiple binary models. An LBP operator is called "equivalent" when it jumps from 0 to 1 or the opposite bitwise position at most twice. The equivalent mode LBP consists of an equivalence mode class and a non-equivalent mode class, and its calculation formula is as shown in equation (8).

$$LBP_{P,R}^{u2} = \left|s(g_{p-1} - g_c) - s(g_0 - g_c)\right| + \sum_{p=1}^{p-1}\left|s(g_{p-1} - g_c) - s(g_p - g_c)\right| \quad (8)$$

U2 represents the equivalence mode, $LBP_{P,R}$ is changed to $LBP_{P,R}^{u2}$, , and the number of modes is reduced from the original 2P to P(P-1) +2, where P refers to the neighbor the number of sample points within the domain set. For the eight sampling points in the neighborhood, the number of binary patterns is reduced from the original 256 to 58, which makes the dimension of the feature vector less, and can reduce the impact of noise.

(3) Rotation-invariant LBP

The LBP operator is gray-scale invariant, but not rotation-invariant. Image rotation will result in different LBP values. Maenpaa et al. [10] extended the LBP operator and proposed an LBP operator with rotation invariance. Compare the LBP value for each time and let the minimum value be the LBP value of the series rotation. Its formula is as shown in equation (9).

$$LBP_{P,R}^{ri} = \min(\mathrm{ROR}(LBP_{P,R}^{ri}, \mathrm{i})\mathrm{i} = 0,1,..\mathrm{p}-1) \quad (9)$$

The ri represents the rotation invariant mode, and the rotation function ROR (x, i) function represents the operation of shifting the bitwise right shift, shifting the x loop right by i(i<p) bits. And it has a good effect on dimension reduction and image rotation.

(4) Rotation-invariant equivalent mode LBP

Rotating the equivalent mode LBP can obtain the rotation-invariant equivalent mode LBP, and its calculation formula is as shown in (10).

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{p-1} s(g_p - g_c) & LBP_{P,R}^{u2} \leq 2 \\ p+1 & Other\ situations \end{cases} \quad (10)$$

Where riu2 represents an equivalent mode with rotation invariance

The improved LBP has both gray-scale invariance and rotation invariance, and the number of modes is also much less. The texture features of the images used in this paper are based on the LBP operator, and the rotation-invariant equivalent mode is used.

Tab 1. Comparison of different LBP dimensions

	Original LBP	Equivalent mode LBP	Rotation-invariant equivalent mode LBP
$LBP_{8.1}$	256	59	10
$LBP_{16.2}$	65536	243	18
$LBP_{24.3}$	16777216	555	26

HOG is a directional gradient histogram feature, LBP is an image texture feature, and two features respectively describe different information of the image, and the two features are combined to complement each other. In this paper, the direct fusion method is used to directly connect two feature vectors to form a new feature vector. For each picture, the HL feature is expressed as:

$$HL=[HOG, LBP_{P,R}^{riu2}] \quad (11)$$

After the comparison, the merged features are less than the original HOG feature dimension.

3. Weighted Component Model

(1) DPM is a very successful target detection algorithm. It has continuously obtained the VOC (Visual Object Class) 07, 08, and 2009 test champions, and has become an important part of many classifiers, human body posture and behavior classification, segmentation and so on. In 2010, Felzenszwalb was awarded the "Lifetime Achievement Award" by VOC. The reason for choosing DPM model is the difference between individuals and the diversity of human posture. The traditional fixed model has great shortcomings for the matching of complex human body. However, DPM introduces deformable components based on the framework of pedestrian detection [11]. Ability to adapt to changes in objects.

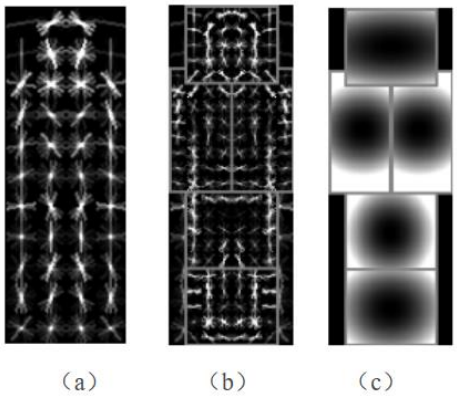


Fig 4. DPM pedestrian model

DPM consists mainly of two parts, one for the root filter covering the entire image, and one for the high-resolution component filter that describes the various parts of each component. As shown in Fig. 4(a), the model of the root filter is relatively ambiguous, generally showing the upright front or back of a pedestrian. As shown in Fig. 4(b), the model of the component filter is clearer. Its resolution is twice the resolution of the root filter model, which can better describe the information of different parts of the pedestrian and better detect pedestrians. Figures 4(a) and 4(b) are axisymmetric in order to reduce the difficulty of the model and facilitate pedestrian detection. Figure 4(c) depicts the deviation loss of the component filter model. The brighter the picture, the greater the deviation loss, and the ideal positional deviation loss of the component filter model is zero.

(2) The detection process of DPM. DPM uses traditional sliding window detection methods to search for scales by building scale pyramids. Figure 5 shows the matching process of the DPM model at a certain scale, that is, the matching process of the pedestrian model. For any input image, extract the feature map of its DPM, and then extract the original image into the Gaussian pyramid and extract its DPM feature map. A convolution operation is performed on the DPM feature map of the original image and the trained root filter to obtain a response map of the root filter. For the DPM feature map of the 2x image, a convolution operation is performed with the trained component filter to obtain a response graph of the component filter. Then down sample the operation of its fine Gaussian pyramid. The response plot of the root filter and the response plot of the component filter have the same resolution. It is then weighted averaged to give the final response map. The greater the brightness, the greater the response value.

In the DPM model, when performing pedestrian detection, the comprehensive score of each position is calculated based on the optimal position of the component. The calculation formula is:

$$\text{score}(p_0) = \max_{p_1,p_2 \ldots p_n} \text{score}(p_0, p_1 \ldots p_n) \tag{12}$$

Where score(p_0) is the response score of the root filter position. In the DPM model, the target assumes z = (p_0, ..., p_n), and gives each filter at each position of the feature pyramid, p_i= (x_i, yi, li) represents the layer where the ith filter is located. And the position coordinates of the upper left corner point, $F_0 F_1 \ldots F_i \ldots F_n$ represents each filter in the deformable part model, where F_0 is the root filter and F_i is the i-th part filter, d_i represents the spatial deformation cost of each possible position of the

component filter relative to the anchor point position. The eigenvector of the filter F arranged in series is F', H is the feature pyramid, and φ (H, pi) represents the pyramid. P_i is the eigenvector of the i-th filter size in the pyramid. The resulting response to the target hypothesis is the response of each filter at its respective position minus the deformation cost at this position relative to the root position plus the deviation value.

$$\text{score}(p_0, \ldots p_n) = \sum_{i=0}^{n} F_i' \cdot \varphi(\text{H}, \text{p}_i) - \sum_{i=1}^{n} \text{d}_i \cdot \varphi_d(dx_i, dy_i) + b \quad (13)$$

Where $(dx_i, dy_i) = (x_i, y_i) - (2(x_0, y_0) + v_i)$ (14) represents the displacement of the i-th component filter relative to its original standard position. (x_0, y_0) represents the coordinates of the root filter at the pyramid layer where it is located. Because of the low resolution, it is necessary to multiply 2 to the resolution of the component filter to perform the subsequent calculations. V_i indicates the offset from the original standard position of the root filter when the i-th component filter is not deformed, so $2(x_0, y_0)$ indicates the absolute coordinates of the component i when no deformation occurs. $\varphi_d(dx, dy) = (dx, dy, dx^2, dy^2)$ (14) represents the deformation cost of the component filter. b is the deviation between the different model components, plus this value to align with the model.

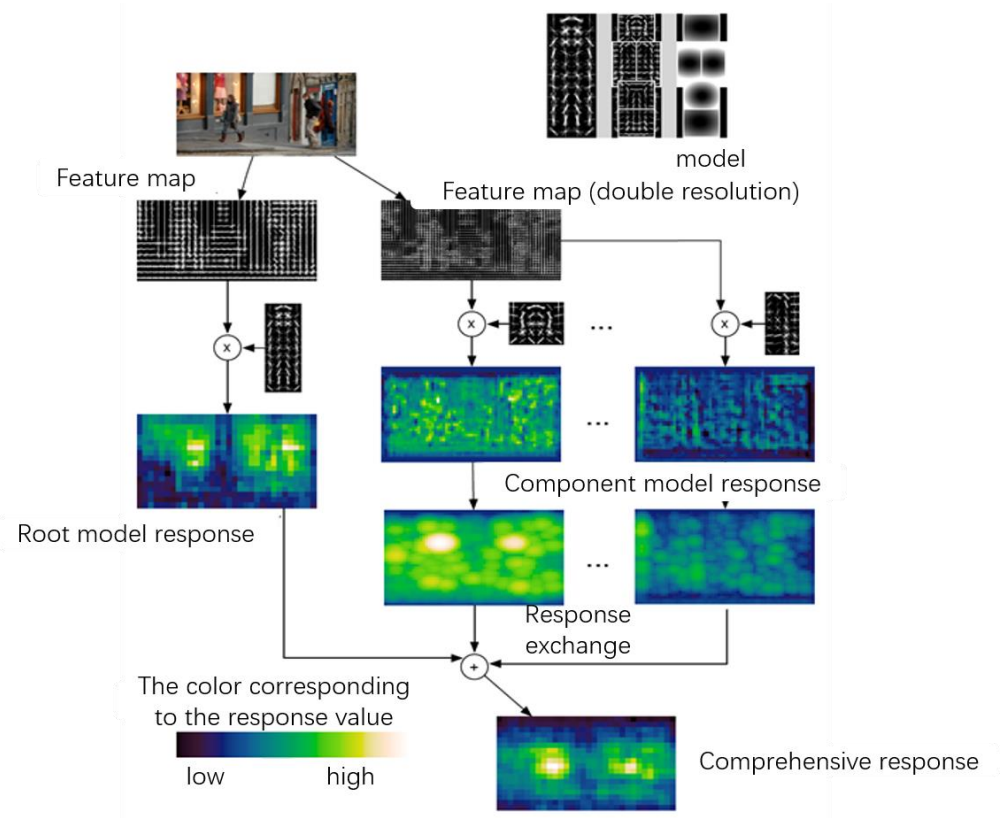


Fig 5. DPM inspection process

(3) Weighted DPM model. In the DPM model, the root filter represents the overall information of the image, the component filter is the information representing the different parts of the image, and the information described by the different components is also different, so the role played in the detection process is also different. According to the traditional DPM does not consider the importance of different components, this paper proposes a weighted component model, which adds different weights to different components, which can distinguish different components and make the component model play a greater role.

Assume that each filter of the part model is represented by $F_0 F_1 \ldots F_i \ldots F_n$. F_0 represents the root filter, and F_i represents the component filter, and the weight of each component filter is ω_i. Therefore, the weighted component model can be expressed as $\omega_1 F_1 \ \ldots \ \omega_i F_i \ \ldots \ \omega_n F_n$. For the target hypothesis z= (p_0 …, p_n), pi= (x_i, y_i) represents the position of the i-th filter at a certain layer of the feature pyramid, then the response of the weighted component model can be expressed as:

$$\text{score}(p_i) = F_0' \cdot \varphi(\text{H}, \text{p}_0) + \sum_{i=1}^{n} w_i F_i \cdot \varphi(\text{H}, \text{p}_i) - \sum_{i=1}^{n} \text{d}_i \cdot \varphi_d(dx_i, dy_i) + b \quad (14)$$

Using the weighted component model for pedestrian detection can increase the detection score of the effective component, and the component detection score that is not very effective is relatively small, and the detection effect is improved.

The key of the weighted component model proposed in this paper is the choice of weights. Different weights will directly affect the effectiveness of pedestrian detection. The choice of weights is determined according to the contribution of different components in the pedestrian detection process. Assume that the model contains the root filter F_0, the component filter $F_1 \ldots F_i \ldots F_n$, and the weight value is calculated according to the following method:

(a) Keep the root filter F_0 unchanged, the F_i in the component filter is unchanged, and set the vector of the remaining component filters to zero;

(b) Test the sample with the modified filter, and record the result as score (F_i);

(c) Repeat the above operation, in turn, keep each component filter unchanged, and set other component filter vectors to zero detection;

(d) Calculate the weight of each component filter based on the detected score. The weight of the i-th component filter is

$$\omega_i = \frac{n * \text{score}(F_i)}{\sum \text{score}(F_i)} \quad (15)$$

The method measures the importance of the component filter based on the detection result. Through these we can analyze the difference in importance between different components, and finally normalize the weight of the obtained weight.

4. Experimental Results and Analysis

The data set used in the experiment is the INRIA data set, which is a static pedestrian detection data set currently used more. The training data set contains 2416 pedestrians, 614 positive samples with marked annotations, and 1218 negative samples; the number of positive samples in the test set is 288, including 1126 pedestrians and 453 negative samples. Some training samples are shown in Figure 6.

(a) Positive sample

(b) Negative sample

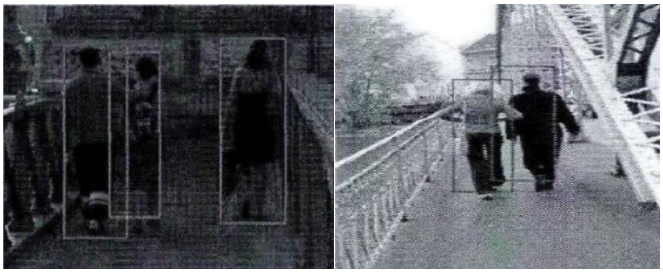
Fig 6. Part of the training sample

Most of the pedestrians in the images in the INRIA dataset are standing, wearing and posing differently, while the data set includes complex backgrounds, a certain occlusion and a different illumination changing environment. The data samples for this experiment are shown in the following table:

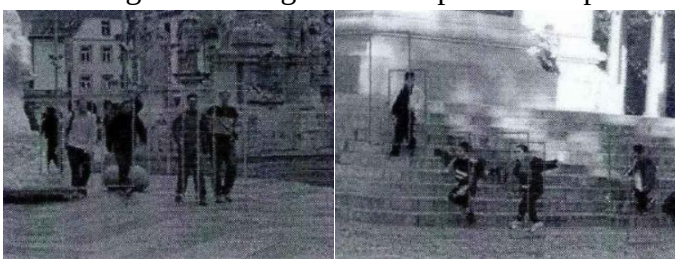
Tab 2. Number of experimental data samples

	Positive sample	Negative sample
Training set	2416	6120
Test set	1126	3000

Shown below is the effect of positive, false and missed detection of weighted DPM model pedestrian detection:



Effect diagram of weighted DPM positive inspection



Effect diagram of weighted DPM false detection

Effect diagram of weighted DPM miss detection

In this paper, the average log-check-off rate curve is used as the performance evaluation standard. Its abscissa is the average false detection rate FPPI of each picture, and the ordinate is the missed detection rate. The number of pedestrian samples that are judged as background in the test picture accounts for the pedestrians. The proportion of the total, the discriminant formula is as follows:

$$\frac{area(B_g \cap B_d)}{area(B_g \cup B_d)} > 0.5 \quad (16)$$

When the above formula is satisfied, the result is judged as positive, otherwise it is falsely detected. Where Bg is the detection result and Bd is the position of the marked pedestrian. Experiments show that the method used in this paper has some progress compared with the traditional DPM, as shown in Figure 7.

When the above formula is satisfied, the result is judged as positive, otherwise it is falsely detected. Where Bg is the detection result and Bd is the position of the marked pedestrian. Experiments show that the method used in this paper has some progress compared with the traditional DPM, as shown in Figure 7.

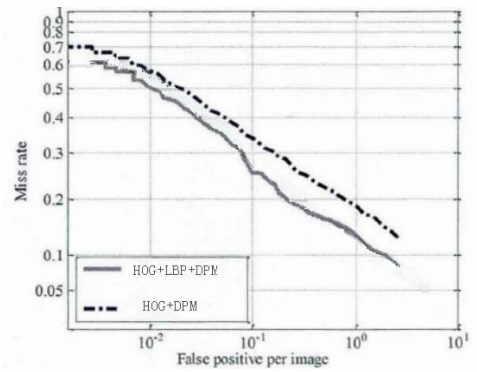


Fig 7. Comparison of experimental results

5. Conclusion

In this paper, the difficulty of pedestrian detection is analyzed. The fusion of HOG and LBP describing the texture features of the image is proposed, and a weighted component model is proposed for the DPM model. Experiments show that in the INRIA data set, the improvement made in this paper has a certain improvement on the effect of pedestrian detection.

References

[1] Dollar P, Wojek C, Schiele B, et al. Pedestrian detection: an evaluation of the state of the art. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 4 (2012) No. 34, p. 741-761.

[2] Su Songzhi, Li Shaozi, Chen Shuyuan, et al. Review of pedestrian detection. Chinese Journal of Electronics, Vol.4 (2012) No. 40, p. 814-820.

[3] Tian Xianxian, Bao Wei, Xu Cheng A pedestrian detection algorithm for improving HOG features. Computer Science, Vol.9(2014) No. 41, p. 320-324.

[4] Hao Xi, Chen Shurong, Yin Daosu. Particle filter pedestrian tracking algorithm combining HOG and color features. Microcomputers and Applications, Vol.6(2014) No.33, p. 40-43.

[5] Dalal N Triggs B. Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition 2005. CVPR2005. IEEE Computer Society Conference on IEEE, Vol.7(2005) No.32, p. 886 - 893.

[6] Chen Rui, Wang Min, Chen Xiao. Pedestrian detection based on PCA dimension reduction HOG and LBP fusion. Information Technology, Vol.2(2015) No.23, p. 101-105.

[7] Paisitkriangkrai S, Shen C, Hengel AV D. Efficient Pedestrian Detection by Directly Optimizing the Partial Area under the ROC Curve. Computer Vision (ICCV), 2013 IEEE International Conference on. IEEE, Vol. 1 (2013) No.23, p. 1057-1064.

[8] P F Felzenszwalb, R B Girshick, D McAllester, et al. Object detection with discriminatively trained part-based models. Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol.9 (2010) No.32, p. 1627-1645.

[9] T Ojala, M Pietikäinen, T Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis & Machine Intelligence, Vol.7 (2002) No.24, p. 971-987.

[10] T Ahonen, J Matas, C He, et al. Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features. Image Analysis, 16th Scandinavian Conference, SCIA 2009, Oslo, Norway, June 15-18, 2009 Proceedings. Vol.1 (2009) No.35, p. 61-71.

[11] Jiménez PG, Bascon SM, Moreno HG, et al. Traffic sign shape classification and localization based on the normalized FFT of the signature of blobs and 2D homographies. Signal Processing, Vol.12(2008) No.32, p. 2943- 2955.