2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing

A New Pedestrian Detect Method in Crowded Scenes

Hou Xin, Zhang Hong, Yuan Ding
Image Processing Center, School of Astronautics
BeiHang University
Beijing, China
hx173149@gmail.com, dmrzhang@buaa.edu.cn

Abstract—**Most existing pedestrian detection methods always focus on improving detect accuracy of single pedestrian detection, but in this paper we focus on detect crowded pedestrians and recognizing adjacent or overlapped pedestrian exactly. We propose a dissimilarity model to represent difference between adjacent pedestrians by utilizing relative spatial information, body part information, color difference, and crowd density information. Through this model we can accurately distinct every pedestrian in a dense crowd. A deep architecture neural network is used in our model, deep belief network. Its low-level feature learning characteristic makes our model have a more intelligent performance. Some optimization measures are used to make our algorithm more efficient. Experiments on an authority dataset have proved the method's effectiveness.**

Keywords—pedestrian detection; crowded scenes; dissimilarity; deep belief network.

I. Introduction

The ability to detect pedestrian is crucial in real-world scenes, such as video surveillance or automatic driver-assistance systems in vehicles, and the technology can also be used in some other domain including pedestrian counting and pedestrian tracking. So the pedestrian detect algorithm's real-time performance, accuracy and robust is becoming more crucial. Some recent algorithms emphasized on rise to these challenges, for example Rodrigo Benenson et al. [21] can make the pedestrian detection at 100 frames per second. And in the real application, the video scenes may be more complicated than normal scenes, such as occluding and overlapping. In this paper we mainly attempt to overcome these difficulties.

While previous classical pedestrian detection algorithms [1, 2] have mainly aim at the normal scenes. But in crowded scenes, pedestrians always are occluded or overlapped by each other, so previous methods cannot achieve a good performance. An intelligent and robust algorithm we proposed just wants to solve the problem. Our goal is to detect all pedestrians in crowded scenes, divide two or more adjacent pedestrians exactly and locate each pedestrian's location in the image. Most classical pedestrian detect algorithms propose a template window, and used some image features [1, 3, 4, 9] to judge the template window weather there is a pedestrian or not. Especially, the Felzenszwalb et al use a part detector model to detect object is a popular and effective method. While the occluded and overlapped pedestrian's image feature may have a big difference with the alone pedestrian's image feature, the template window may not be effective in crowd. In this case, some people have attempted to solve it, like Bastian Leibe et al. used a probabilistic top-down segmentation method [5] to divide two adjacent pedestrians; Junjie Yan et al. proposed a global view method [6] to solve the problem. But there is still have room to improved, in our proposed method, we not only take the alone template window into consideration but also consider the adjacent pedestrians' information.

This method mainly makes three contributions: first, we propose a pedestrian's detect difference model, named Dissimilarity Model (DS model) which contains both the alone pedestrian's appearance information and distinct pedestrian's difference information. Second, we define the distinct pedestrian's difference information is composed of relative spatial information, body part information, crowded dense information and the distinct pedestrian's color difference. Finally, in the training stage we use a deep belief network to automatically extract and combine these low-level features to a high-level feature, and use the DS model to detect the adjacent pedestrians. The experiment's result has proved our algorithm is effective on pedestrian detection in crowded scenes.

The rest of the paper is organized as follows: Section II introduces some related work. Section III illustrates our method's work procedure and discussed our distinct pedestrian's dissimilarity model. Section IV introduces the deep belief network. Several optimization methods about our algorithm are given in Section V. Section VI shows the experiments and in Section VII we conclude the paper.

II. Related work

The classical pedestrian detect method which uses HOG feature and SVM classifier[1] can make a good effect on detecting alone pedestrian, then some work have been done for improving it, such as[2, 7, 8]. Unlike the most traditional template based method which aim at judging whether the template window is a pedestrian or not, our algorithm is pay attention to find the difference between the adjacent pedestrians. So we just define a DS model that contains both the two templates hypotheses' difference information which may represent the two templates hypotheses is a whole pedestrian, two distinct pedestrians or none pedestrian. To make template hypothesis' feature have more efficient difference information, we adopt some new image features.

In order to access a higher reliable template hypothesis window, we use Felzenszwalb's Deformable Part Models (DPM) [9], which can detect all the pedestrian's parts score. And these parts score can make a second-level feature for pedestrian's body part information. It's efficient to overlapping and deformation.

978-0-7695-5046-6/13 $26.00 © 2013 IEEE
DOI 10.1109/GreenCom-iThings-CPSCom.2013.337

IEEE computer society

A more complicated and smarter classifier is used in our method. Deep Belief Network (DBN), proposed by Geoffrey Hinton [10, 11, 12], its deep architecture likes the brain's neural network because the signal is transformed layer by layer. And the kind of network has been proven that it is a so intelligent machine learning method in the low-level feature space. Deep learning method can train a low-level feature and composition them to a high-level feature in an unsupervised way which just like a thinking process. So we can just use the high-level feature to do a more accurate classify. Finally, we have made some experiments to prove that the deep learning method is better than the traditional classifier.

III. DS MODEL

In this section we will illustrate our system's work procedure. And the Dissimilarity Model will be introduced. First, The HOG feature and SVM classifier method is used to detect the all pedestrian part's scores shows in Fig.1.

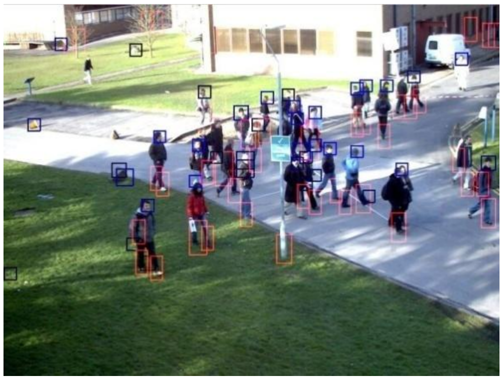

Fig. 1. HOG+SVM detect pedestrian's part, blue window is head part and red window is leg part.

Second, the DPM model is used to detect some template hypotheses windows for possible pedestrian's location shows in Fig.2.

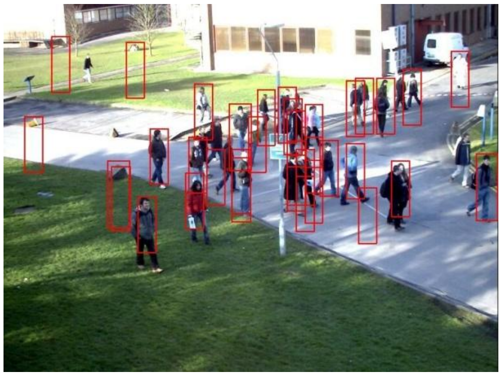

Fig. 2. template hypotheses windows

Finally, the DS model can be used to judge every hypothesis and its nearest hypothesis' truthfulness about whether the hypothesis window has pedestrian shows in Fig.3.

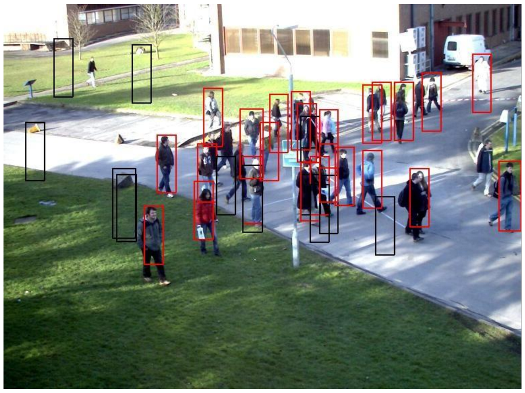

Fig. 3. DS model judged adjacent hypotheses' truthfulness, red window is true pedestrian, black window is false pedestrian

DS model:

$$(S_1, S_2) = \mathrm{DS}(\mathbf{X_1}, \mathbf{X_2}) \tag{1}$$

Where $\mathbf{X}$ is a low-level feature of template hypothesis window and S is template hypothesis window's judged result, which can judge hypothesis window's truthfulness. The model's judged result is illustrated by the TABLE I.

TABLE I. MEANING OF THE DS MODEL'S JUDGED RESULT

(S_1, S_2)	**Defined Meanings**
(0,0)	All of the hypotheses are false pedestrians
(1,0)	First hypothesis is true pedestrian, second is false pedestrian
(0,1)	First hypothesis is false pedestrian, second is true pedestrian
(1,1)	All of the hypotheses are true pedestrians (and they are different pedestrians)

What features in our model can discriminate two adjacent hypotheses in our DS model will be introduced as flow:

A. Part Spatial Feature

The part spatial feature is used to describe pedestrian's head, body and leg part. So we can infer the pedestrian's truthfulness and their location from pedestrian's part information in two adjacent template hypotheses windows. For example, from Fig.4(a) it can easily be judged that the yellow hypothesis window is a false pedestrian and the blue hypothesis window is a true pedestrian. As the same reason, we can judge whether the two adjacent hypotheses represent a same pedestrian, two different pedestrians or none. In our model, three kinds of scale hypothesis window have been selected to represent probable pedestrian. Every template hypothecs' spatial feature is made up of 5*5 head windows score $\mathbf{S_{head}} = (sh_1, sh_2 ......sh_{25})$, 7*5 leg windows score $\mathbf{S_{leg}} = (sl_1, sl_2 ......sl_{35})$ and its location information $\mathbf{L} = (x, y, scale)$. The selected head and leg detector windows' location and size are showed in Fig.4(b).

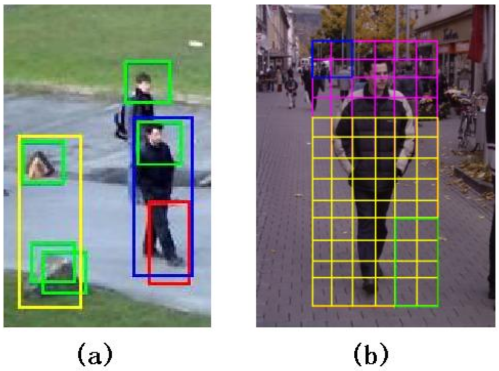

Fig. 4. (a) green windows have a high head score and red windows have a high leg score, but the yellow window is false pedestrian and blue window is true pedestrian. (b) red windows' location are head windows score S_{head} location, yellow windows' location are leg windows score S_{leg} location. Blue and green windows' size represent head and leg detector's size

B. Color Dissimilarity Feature

Under normal circumstance, people will wear the different clothes, even they are on the same cloth, and the background will have little difference. Fig.5 shows, while two pedestrians stand very close and the part spatial feature satisfy the condition which can judge two hypotheses is a whole true pedestrian, the two pedestrians are easily treated as one pedestrian. So in two different pedestrians hypotheses window the color information may have a big difference to discriminate them, shows in Fig.6. Hence we just take the hypothesis' color feature into consideration to discriminate two adjacent hypotheses. A simply color feature was used: color histogram [14]:

$$\mathbf{Col} = (\mathbf{R}, \mathbf{G}, \mathbf{B}) \tag{2}$$

$\mathbf{R}, \mathbf{G}, \mathbf{B}$ are three color channels' histogram feature. Experiment's experience shows that top 20 color components can represent the major information in color histogram, so we just use the top 20 components at every channel.

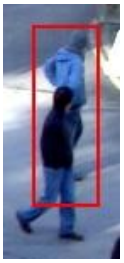

Fig. 5. Two adjacent pedestrians may be detect as a one pedestrian only judged by part spatial feature.

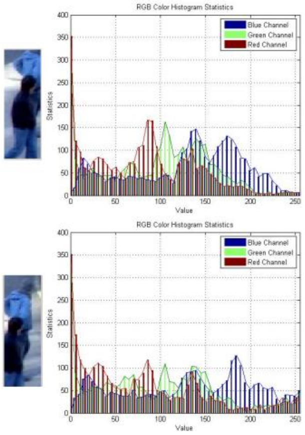

Fig. 6. two adjacent pedestrian hypothesis' color histogram

C. Crowded Dense Feature

Generally, the pedestrian detection is always used in a video scene [20]. So the background and foreground information can help us access the crowded dense information. The Gaussian Mixture Model (GMM) method [13] is used to extract hypothesis' background and foreground information shows in Fig.7.

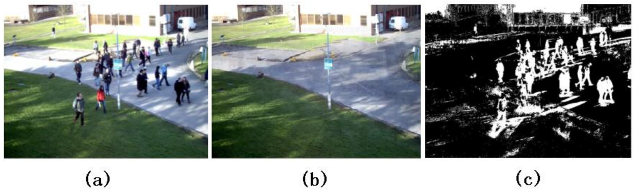

Fig. 7. Extract background and foreground information

We define the hypothesis window's crowded dense ture F as:

$$F = \frac{\sum\limits_{px \in (hypotheces \cap foreground)} 1}{\sum\limits_{px \in hypotheces} 1} \tag{3}$$

In the equation (3), px represent a pixel in hypothesis' window.

D. Features Combination

The feature vector $\mathbf{X}$ can be described at equation (4):

$$\mathbf{X} = (\mathbf{L}, \mathbf{S_{head}}, \mathbf{S_{leg}}, \mathbf{Col}, F) \tag{4}$$

For every two adjacent hypotheses, the DS model is used to judge $(S_i, S_j) = \mathrm{DS}(\mathbf{X_i}, \mathbf{X_j})$. But if one hypothesis does not have an adjacent hypothesis, the DS model will be used in a same feature $\mathbf{X}$ as $(\mathbf{X_i}, \mathbf{X_i})$. The condition to judge two hypotheses are adjacent relies on two hypotheses' relative distance:

$$d_x < thresholdx \, \&\& d_y < thresholdy \tag{5}$$

where d_x and d_y are two hypotheses' horizontal and vertical distance. From our experimental experience shows the *thresholdx* always be $3*h_w$, and *thresholdy* always be $3*h_l$. h_w and h_l are hypothesis window's width and length.

IV. Deep Learning used in DS model

Another important component about our method will be introduced in this section. A wide range of classifiers have been proposed for pedestrian detection. Most popular classifier choices are linear SVM and AdaBoost[15, 16], these algorithms have a good effect on normal scenes. But we will apply a deep architecture neural network to do the classify work for our model. Deep learning methods aim at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower feature. Automatically learning features at multiple levels of abstraction allows a system to learn complex functions mapping the input to output directly from data, without depending completely on human-crafted feature. And the experiments' result will confirm that deep learning's effect is better than major previous classifiers'.

A. The Restricted Boltzmann Machine (RBM)

As RBM is a basic and important building block of deep learning model, so we will give a brief introduction on it [10]. Express the stochastic input visible vector by $\mathbf{X}=[x_1,x_2......x_n]$, and denote the hidden variables by $\mathbf{H}=[h_1,h_2......h_m]$.The RBM defines a probability distribution over h and x as:

$$\mathrm{p}(\mathbf{X},\mathbf{H}) \propto e^{[\mathbf{X^T W H + c^T H + b^T X}]} \tag{6}$$

$\mathbf{X}$ form the visible layer and H forms hidden layer. There are symmetric connections $\mathbf{W}$ between the visible layer and the hidden layer, but no connection for variables within the same layer. The graphical model of RBM is show in Fig.8. This particular configuration makes it easy to compute the conditional probability distributions:

$$\mathrm{p}(x_n = 1 \mid \mathbf{H}) = \sigma(\mathbf{W_{n,*}}\mathbf{H} + b_n) \tag{7}$$

$$\mathrm{p}(h_i = 1 \mid \mathbf{X}) = \sigma(\mathbf{X^T}\mathbf{W_{*,i}} + c_i) \tag{8}$$

Where $\mathbf{w_{n,*}}$ is the nth row of $\mathbf{w}$, $\mathbf{w_{i,*}}$ is the ith column of $\mathbf{w}$ and $\mathrm{s}(t) = (1+\exp(-t))^{-1}$ is the logistic function. The contrastive divergence in [17] is used for learning the parameters $\mathbf{w}$, c and b in (7) and (8). The RBM's learning process has the function that learning the low-level feature to high-level feature automatically.

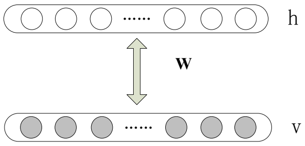

Fig. 8. RBM's architecture

B. The Deep Belief Network (DBN)

DBN[18] is a multilayer networks, we can train each of two adjacent layers with a RBM. And make the previous hidden layer as the input layer for the next RBM, finally use a one-layer BP network to fine tune the whole deep network. Since the RBM can do the unsupervised learning work, we consider previous low layers have function that extract and composed low-level feature.

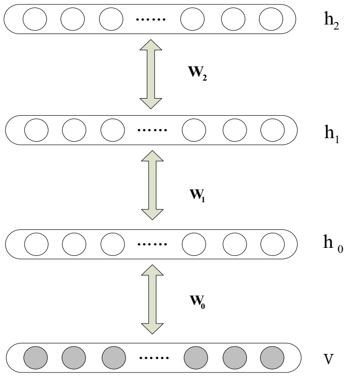

Fig. 9. DBN's architecture

The system is used to our model as follows: the first input layer of the DBN is the two combinational features $(\mathbf{X_i},\mathbf{X_j})$. And the highest BP network layer is our model's judged result (S_1,S_2) . The DBN's layers number is selected from 3 to 6.

V. Optimization

The optimization about our algorithm includes three aspects: how to estimate the hypothesis window as accurately as possible, how to use our model to do a correct judgment, and how to access the optimal parameters.

The DPM model is used to detect the pedestrian's part, but we just detect the head part and leg part. Because in crowded scenes pedestrian's size is usually small, so it is hard to detect all body part of a pedestrian. In addition, the non-maxima suppression (NMS) [19, 9] is used to filter some heavily overlapped part windows. Besides, in order to have a high efficiency to estimate the hypothesis template, each head was regarded as a hypothesis. Because we think head is the most crucial and has the least deformable in all body's part.

Sometimes a hypothesis window may be judged in DS model multi-times. So the hypothesis may have several scores, compute its average score to judge the hypothesis window's truthfulness is an effective method.

At experiment stage, some experience about training DBN have been concluded that at low layers, the hidden nodes number always more than input layer nodes, because in the low layers, features are always mess and low-level. But in the higher layers, the hidden nodes always decrease layer by layer, because the features have become high-level after the low layers' learning and combination.

VI. Experiments

In this part we compare proposed method with the classical method HOG+SVM method [1] and Integral HOG-LBP+SVM detect method [2] which is state-of-the-art on general pedestrian detection. We select PETS_2009 [22] as the test set, which is a well-known crowd database. In our experiment we select S1_L1 for training, S2_L2 with medium density crowd for test. S1_L1 contains 220 frames and 4502 pedestrians; S2_L2 contains 436 frames and 8927 pedestrians;

Fig.10 shows the recall-precision curves about three algorithms in the S2_L2 and some result example image are showed in Fig.11 and Fig.12. Our algorithm have a higher precision at the same recall or have a higher recall at the same precision in compared with others algorithms, showed in Fig.10. In addition, the detection result example Fig.11 and Fig.12 also illustrate that the pedestrian's location can be located exactly.

In these experiments, it can be proved that our method can have a better performance on pedestrian detection in crowded scenes.

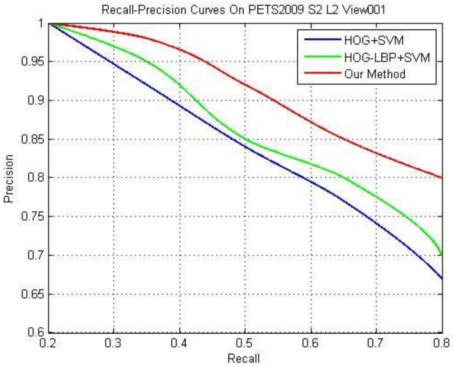

Fig. 10. Three methods' recall-precision compare curves

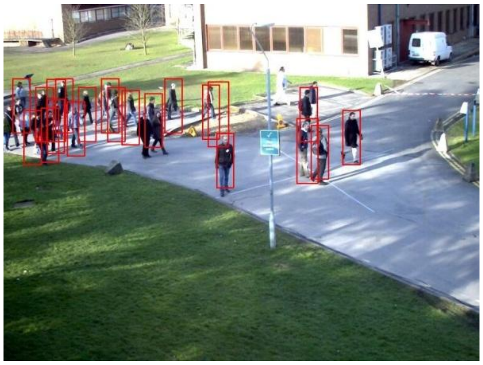

Fig. 11. Detection result example 1

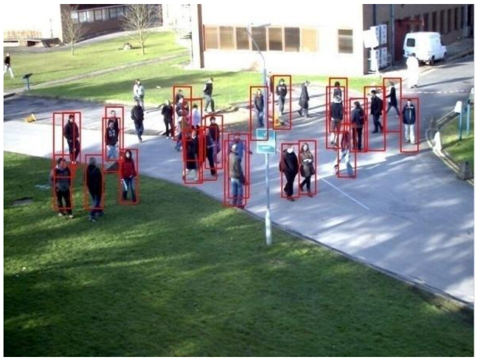

Fig. 12. Detection result example 2

VII. Conclusion

In this paper, we have presented a new method for pedestrian detection in crowded scenes. Our method is pay attention to design a model which can divide two adjacent pedestrians in crowded scenes. And it's also important to emphasize that our method used various features like color, video information and edge information. Besides, we use the deep belief network to replace the general classifier. Experiment result has prove that our method can improve the detect effect significantly.

However, our system's capability to robustly detect pedestrians in crowded scenes and how to estimate the hypothesis as accuracy as possible are still need to explore, we will make a further research on this challenge in future work.

References

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In CVPR, 2005.

[2] Xiaoyu wang, Tony X.Han, and Shuicheng Yan. An HOG-LBP Human Detector with Partial Occlusion Handling. In ICCV,2009

[3] Paul Viola, Michael J.Jones, and Daniel Snow. Detecting Pedestrians Using Patterns of Motion and Apperance. In ICCV, 2003

[4] Lubomir Bourdev and Jitendra Malik. Poselets: Body Part Detectors Trained Using 3D Human Pose Annotations. In ICCV 2009

[5] Bastian Leibe, Edgar Seemann, and Bernt Schiele, Pedestrian Detection in Crowded Scenes. In ICCV 2001

[6] Junjie Yan, Zhen Lei, Dong Yi, Stan Z. Li. Multi-Pedestrian Detection in Crowded Scenes: A Global View. In CVPR 2012.

[7] Lili cheng, Jianpei Zhang, Jing Yang, Jun Ma. An Improved AdaBoost Algorithm Based on Adaptive Weight Adjusting. Third International Conference,ADMA 2007,PP 625-632

[8] N. Dalal, B. Triggs and C. Schmid. Human detection using oriented histograms of flow and appearance. In ECCV, 2006.

[9] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester and Deva Ramanan. Object Detection with Discriminatively Trained Part-Based Models. In CVPR,2011

[10] Geoffrey Hinton, A Practical Guide to Training estricted Boltzmann Machines

[11] Yoshua Bengio, Learning Deep Architectures for AI

[12] Geoffrey Hinton and Salakhutdinov. Reducing the imensionality of data with neural networks. Science, 13(5786):504 – 507, July 2

[13] Douglas A. Reynolds, Robust text-independent speaker identification using Gaussian mixture speaker models, IEEE Trans. On Speech and Audio Processing, Vol.3, No.1, pp .72-83

[14] Carol L.Novak and Steven A.Shafer,Anatomy of a Color Histogram

[15] Paul Viola and Michael J.Jones, Robust Real-Time Face Detection.In IJCV 2004

[16] Corinna. Cortes and Vladimir Vapnik, Support-vector networks. Machine Learning September 1995,Volume 20,issue 3,pp,273-297

[17] Geoffrey Hinton, Training products of experts by minimizing contrastive divergence. Neural Computation, 14:1771–1800, 2002.

[18] Geoffrey Hinton, Simon Osindero and Yee-Whye The, A Fast Learning Algorithm for Deep Belief Nets.Neural Computation, pp.1527-1554 2006.18.7

[19] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. IJCV, 77(1):259–289, 2008

[20] Jinhao Deng, Juan Zhu, Research on Pedestrian Detection Algorithms Based on Video. In ICCDA 2010

[21] Rodrigo Benenson, Markus Mathias, Radu Timofte and Luc Van Gool, Pedestrian detection at 100 frames per second. In CVPR 2012.

[22] PETS 2009.http://www.cvg.rdg.ac.uk/PETS2009/a.html