*Sensors and Materials*, Vol. 32, No. 6 (2020) 1997–2006
MYU Tokyo

S & M 2235

# Application of Convolutional Neural Network (CNN)–AdaBoost Algorithm in Pedestrian Detection

Guiyuan Li,[1,2] Changfu Zong,[1*] Guangfeng Liu,[1] and Tianjun Zhu[3]

[1]State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130025, P. R. China
[2]School of Automobile and Traffic Engineering, Liaoning University of Technology, Jinzhou 121001, P. R. China
[3]Department of Mechanical and Automotive Engineering, Zhaoqing University, Zhaoqing 526061, P. R. China

(Received December 26, 2019; accepted April 8, 2020)

***Keywords:*** pedestrian detection, autonomous driving, aggregate channel feature, convolutional neural networks

Pedestrian detection based on vision sensors is a hot and difficult issue in the field of autonomous driving. The large amount of data processing leads to high requirements for the robustness and real-time performance of the employed algorithm. The aggregate channel feature (ACF) algorithm is one of the widely recognized fast pedestrian detection algorithms, but there are many missed detections when the target is occluded or small. In response to this problem, we propose a pedestrian detection algorithm based on a combination of a five-layer convolutional neural network structure and an AdaBoost classifier (CNN–AdaBoost). The model was trained using Caltech and INRIA datasets, and detection experiments were performed using collected videos. The results show that the error detection rate of the proposed algorithm is greatly reduced compared with that of the ACF algorithm, but the detection speed is basically unchanged. Compared with the locally decorrelated channel features (LDCF) algorithm, the proposed algorithm achieves similar detection accuracy but the detection efficiency is greatly improved.

## 1. Introduction

Vision sensors can provide high-resolution color information, which can more accurately reflect the details of complex changes in light. Therefore, pedestrian detection based on vision sensors has wide application in many fields such as the military, traffic, and security fields. Because a pedestrian has the characteristics of scale, motion, and pose variation, and the appearance is easily influenced by factors, such as clothes, sunlight, shielding, and viewing angle, pedestrian detection is a difficult and hot issue with major challenges.

The key factor restricting the application of pedestrian detection methods in intelligent driving is the large amount of data processing, leading to high requirements for the robustness and real-time performance of the employed algorithm. Currently, the basic pedestrian detection methods can be divided into two categories from the perspective of the feature acquisition

[*]Corresponding author: e-mail: 2609582477@qq.com

https://doi.org/10.18494/SAM.2020.2787

method:[1,2] one is the traditional machine learning method based on artificial features, and the other is the deep learning method based on convolutional neural network (CNN) features. The basic framework of traditional machine learning methods includes feature extraction and classifiers. Features here mainly include the histogram of the oriented gradient (HOG),[3] local binary pattern (LBP),[4] deformable part model (DPM),[5] and aggregate channel feature (ACF).[6] Classifiers include the support vector machine (SVM), decision tree (DT), random forest (RF), and AdaBoost. The basic framework of deep learning methods includes a deep CNN and a classifier, which uses the deep CNN for feature extraction, and in typical structures such as GoogleNet, ZFNet, AlexNet, VGGNet, and ResNet, the classifier is generally an ordinary fully connected neural network. R-CNN, YOLO, and other deep learning detection frameworks have better pedestrian detection performance than traditional machine learning,[7,8] but the training of their models requires hardware with high computing power and massive datasets. The training is time-consuming, and it is difficult to perform training tasks using ordinary PCs. Morevoer, large datasets are not easy to obtain. Owing to the lack of a theoretical foundation, the design of a network's hyperparameters is also a considerable challenge. For target detection with a small dataset, traditional machine learning methods are usually better than deep learning. The ACF algorithm proposed by Dollar *et al.* is one of the widely recognized fast pedestrian detection algorithms.[6] The ACF algorithm is based on integral channel features (ICF),[9] and an AdaBoost classifier composed of 2048 two-layer DTs is used in the algorithm. The locally decorrelated channel features (LDCF) algorithm is based on the ACF algorithm and uses linear discriminant analysis (LDA) to obtain the final LDCF features.[10] The weak classifier used is a DT with a depth of five layers, and the total number of cascaded weak classifiers is 4096. The missed detection rate of the LDCF algorithm tested on the Caltech dataset reached 29.8%, about 16.2% less than that of the ACF algorithm. However, its missed detection rate was still large, especially when there were small or occluded pedestrian targets in the test. Ma and Gao proposed a combination of the LDCF algorithm and a CNN,[11] with the LDCF algorithm used to obtain region proposals, then the CNN used to extract features, and an SVM used to classify the extracted features. Zhang *et al.* used a region proposal network (RPN) in a faster R-CNN to extract region proposals,[12] and then used boosted forests to classify features. Mao *et al.* added a VGG-16 network on the front end of a faster R-CNN to obtain additional channel features.[13] Ouyang and Wang proposed the joint deep method,[14] which uses an SVM as the first-level detector and a CNN to further determine its detection results. The above method uses a deep CNN to improve the detection accuracy, but also requires a large number of datasets and long-term training. The real-time performance of algorithms also requires advanced hardware support.

On the basis of the above research, the combination of a deep CNN and traditional machine learning to improve the performance of pedestrian detection is currently the most popular technical route. However, a problem with this approach is how to effectively reduce the depth of the CNN while improving the detection accuracy in order to reduce the dependence of the algorithm on the dataset and hardware. In response to this problem, we propose a pedestrian detection method (CNN–AdaBoost) based on an AdaBoost classifier combined with a CNN feature extractor. First, we refer to the fast R-CNN framework to improve the detection efficiency.[15] In view of the high miss rate of the AdaBoost classifier in the ACF algorithm, we propose a negative sample retrieval strategy to improve it. Second, we design a five-layer CNN, which is used as a feature extractor to improve the detection rate of small pedestrian targets. The rest of the paper is organized as follows. In Sect. 2.1, we introduce the overall framework, detection, and training process, and then in Sect. 2.2, we introduce the negative sample retrieval strategy and the structure of the five-layer CNN. The experimental results are given in Sect. 3. Conclusions are given in Sect. 4.

## 2. CNN–AdaBoost Detection Algorithm

### 2.1 Basic framework of the algorithm

The overall architecture of the CNN–AdaBoost algorithm proposed in this paper is shown in Fig. 1. It mainly includes four parts: a fast feature pyramid part, a region proposal selection part, a CNN feature extraction part, and a feature processing part.

In the detection phase, a color image is first calculated through a fast feature pyramid with multiscale AFC features. The region proposal section in the upper branch of Fig. 1 uses a fixed-size sliding window (the red rectangular frame on the fast feature pyramids picture in Fig. 1) to extract the ACF features layer by layer from the bottom to the top of the feature pyramid. The ACF features are expanded into a feature vector, and the target and non-targets are filtered step by step through an AdaBoost classifier. For the non-targets, we use a negative sample retrieval strategy to make the position of each non-target a candidate region again. This will not affect the overall detection efficiency of the AdaBoost classifier and, at the same time, it can effectively reduce its false detection rate. The CNN feature extraction part in the lower branch of Fig. 1 extracts the L color space features (the green rectangular frame in the fast feature pyramids picture in Fig. 1) of the LUV color space layer by layer from the bottom to the top of the pyramid, and the features extracted by the CNN have better expressive power for small targets than ACF features. The task of region of interest (ROI) feature extraction is to obtain the data of the feature map in the corresponding position of the proposal region, and finally, the feature is classified by the fully connected layer and Softmax.

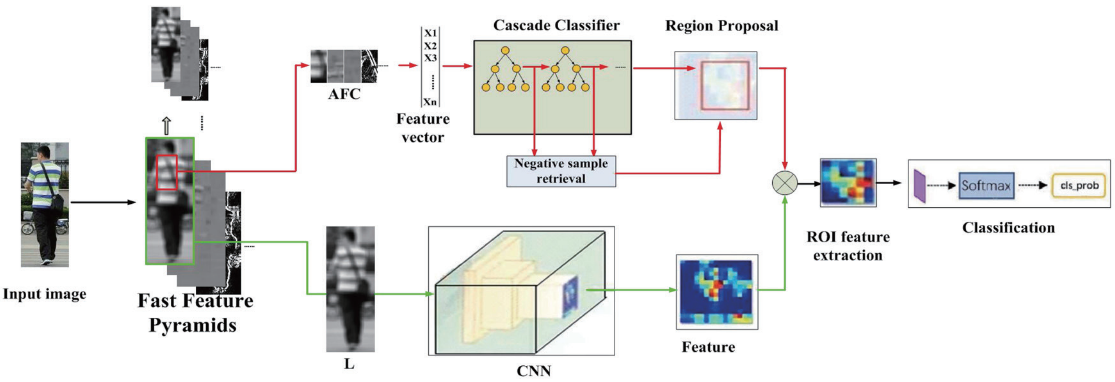


Fig. 1. (Color online) Overall architecture of the CNN–AdaBoost algorithm.

In the training phase, AdaBoost and the CNN are trained separately using conventional methods. AdaBoost is trained using ACF features, and the CNN is trained using L color space features.

### 2.2 Negative sample retrieval strategy and CNN structure

Ohn-Bar and Trivedi showed that AdaBoost is different from the CNN, and it is difficult to further improve its detection performance by increasing the size of the trained pedestrian dataset and the depth of the DT.(16) To maintain the depth of the ACF algorithm's DT, a negative sample retrieval strategy is proposed in this paper. The basic idea of the strategy is to reselect the regions that have been detected as negative samples as proposal regions using the AdaBoost classifier. Specifically, a scale threshold and a rank threshold are set. The scale threshold is set according to the number of layers of the fast feature pyramid. A value smaller than the threshold indicates that the scale is close to the top of the pyramid, and the rank threshold is set according to the number of cascaded AdaBoost strong classifiers. A value greater than the threshold indicates that the rank is near the end classifier. When the scale of pyramid layers that the sliding window is on is less than the scale threshold and the rank of activated strong classifiers is greater than the rank threshold, if the result of the strong classifier is a non-pedestrian target, the sliding window position of the non-pedestrians is used as the proposal region.

For pedestrian detection using on-board cameras, we hope to find a CNN with a simple structure, an easy-to-use small scale for training on ordinary PCs, and a detection speed that can meet real-time requirements. Here, we employ a five-layer CNN, where the size of the convolution kernel is 9 × 9 and the neighborhood of the maximum pooling method is 2 × 2. By adjusting the number of convolution kernels in the third layer, we obtain four different CNNs, After the preliminary training, a test experiment is carried out. The training mean square error curve of each CNN is shown in Fig. 2 and the recognition rate is shown in Table 1. It can be seen that CNN2 has the highest speed and accuracy, so CNN2 is subsequently used in this study.

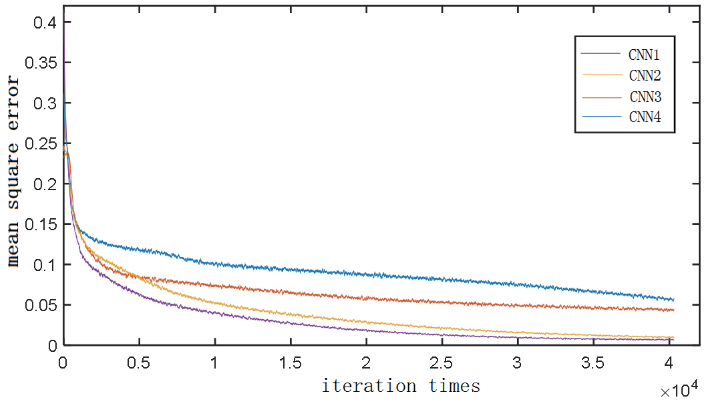


Fig. 2. (Color online) Mean square error curves of four CNNs.

Table 1
Four CNN structures and test results on Caltech and INRIA datasets.

| | | | | | |
|---|---|---|---|---|---|
| Input | | 64 × 32<br>L of the LUV color space | | | |
| Layer 1 | | Conv layer (convolution kernel) 3rd-order tensor C: 6 × 9 × 9 | | | |
| Layer 2 | | Pooling layer (maximum pooling) matrix P: 2 × 2 | | | |
| Layer 3 | | C: 5 × 9 × 9 | C: 12 × 9 × 9 | C: 18 × 9 × 9 | C: 27 × 9 × 9 |
| Layer 4 | | P: 2 × 2 | | | |
| FC layer (dimensions of feature) | | 100 | 240 | 360 | 540 |
| Rate of detection (%) | INRIA | 93.74 | 95.12 | 94.05 | 94.75 |
| | Caltech | 79.92 | 80.6 | 80.37 | 80.22 |
| CNN | | CNN1 | CNN2 | CNN3 | CNN4 |

## 3. Results and Discussion

The experimental hardware platform is an Intel (R) Core (TM) i3-2370M CPU (2.4 GHz, 6 GB RAM) and the experimental software platform is the Windows 7 operating system with MATLAB R2015b. The color vehicular camera used has a frame rate of 24 fps and a resolution of 640 × 480. The experimental training test data is shown in Table 2. The training of the AdaBoost classifier uses the Caltech training set, and the training of the CNN uses a combination of the Caltech training set and the INRIA training set (Table 2). The test set uses the INRIA test set, Caltech test set, and collected videos. The video test set we used is obtained on the campus through a color vehicular camera. The ACF and LDCF algorithms are used for comparison.

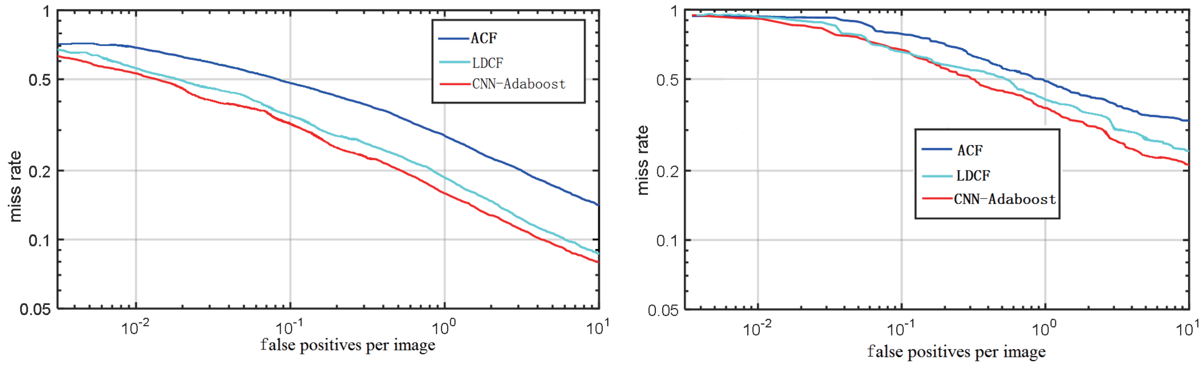
Figure 3 shows the miss rate–false positives per image (FPPI) curves of the Caltech test set and INRIA test set. Figures 4–6 show the test results for different scenes in the video test set. The left column is the detection result of the ACF algorithm, the middle column is the detection result of the LDCF algorithm, and the right column is the detection result of the proposed algorithm. It can be seen from Fig. 3 that the CNN–AdaBoost algorithm performs better on the Caltech dataset than on the INRIA dataset, and its detection performance for the two datasets is generally better than those of the ACF and LDCF algorithms.

In Fig. 4, the pedestrians have different sizes. For large pedestrian targets, the three methods can correctly detect larger pedestrians, but the ACF algorithm misses the detection of small targets. The detection of small targets by the LDCF algorithm is improved compared with the ACF algorithm, but when the small targets are closer, they are sometimes not detected. However, the proposed method can still distinguish different targets in this case. In Fig. 5, there are mutually occluded targets. The ACF and LDCF algorithms can generally detect large occluded targets, but in the case of small occluded targets, missed detection and false detection occur. However, the proposed method can still detect the occluded targets in these cases. There are deformed targets in Fig. 6. Although all three methods can correctly detect deformed targets at a short distance, the ACF and LDCF algorithms generally fail to detect small deformed targets, while the proposed method can still detect them.

The missed detection rate, the average number of false frame detections, and the detection efficiency are evaluated for the video set, and the results are shown in Table 3. The ACF

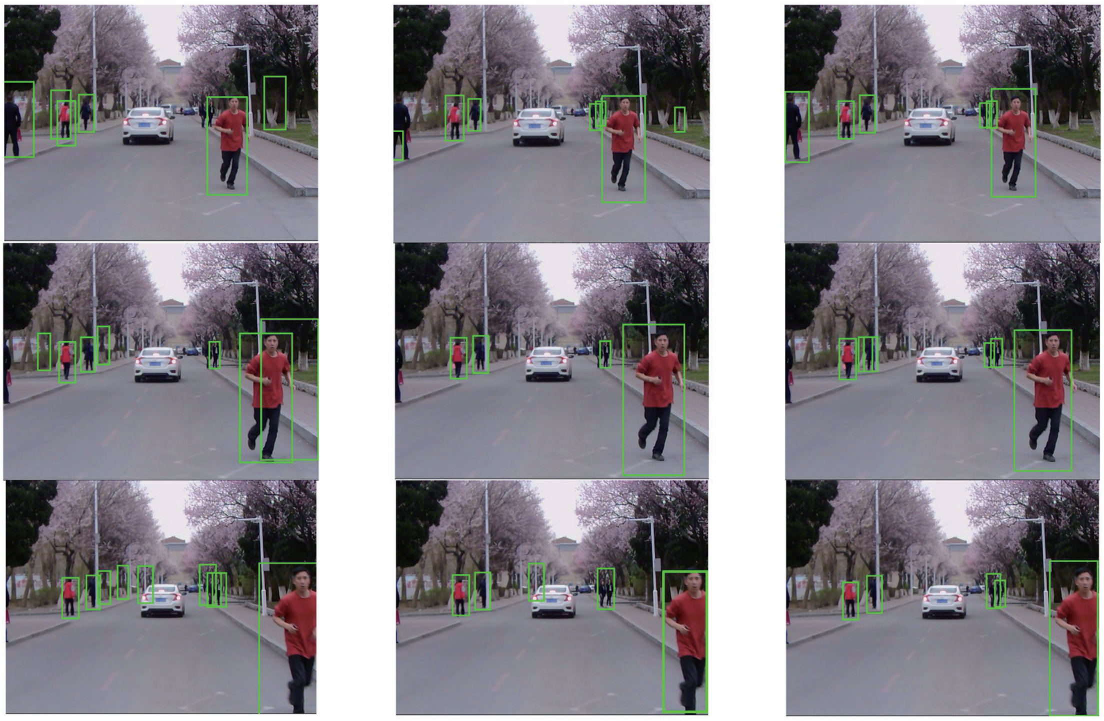
Table 2
Training data.

| | | Positive samples | Negative samples | Total number | |
|---|---|---|---|---|---|
| Training set | Caltech | 20000 | 50000 | 70000 | 78700 |
| | INRIA | 2200 | 6500 | 8700 | |
| Test set | Caltech | 15000 | 50000 | 65000 | 72700 |
| | INRIA | 1200 | 6500 | 7700 | |



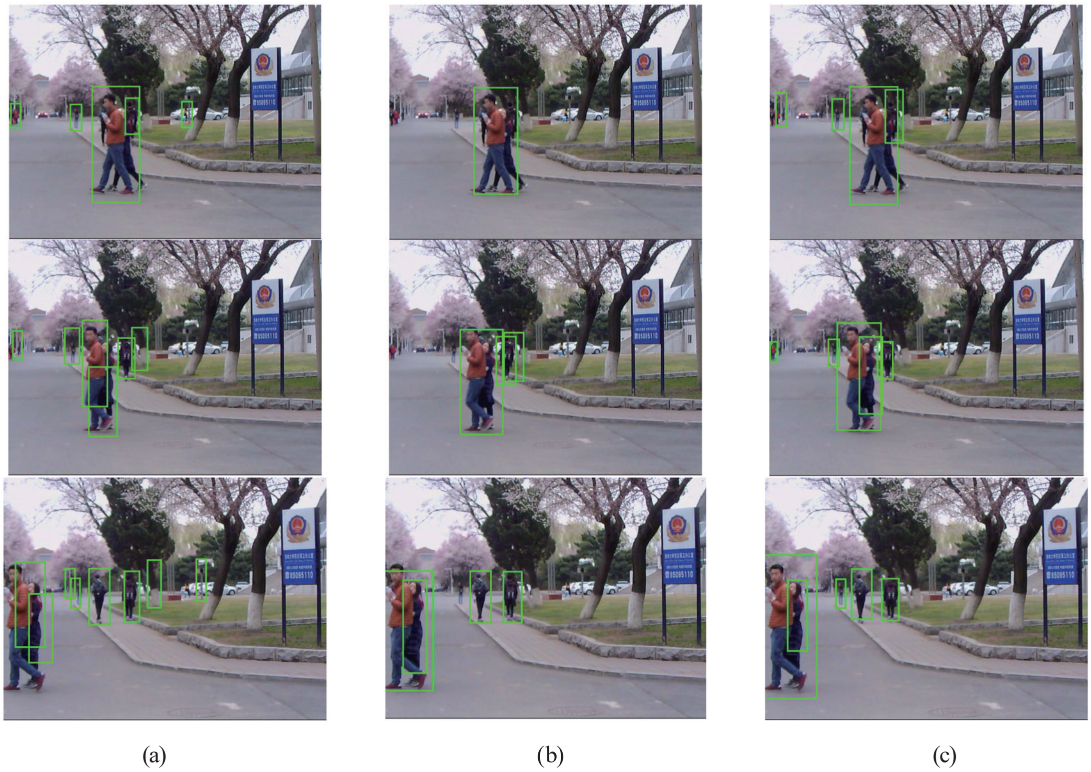
(a) (b)

Fig. 3. (Color online) Miss rate–FPPI curves. (a) Miss rate–FPPI curve of Caltech test set. (b) Miss rate–FPPI curve of INRIA test set.



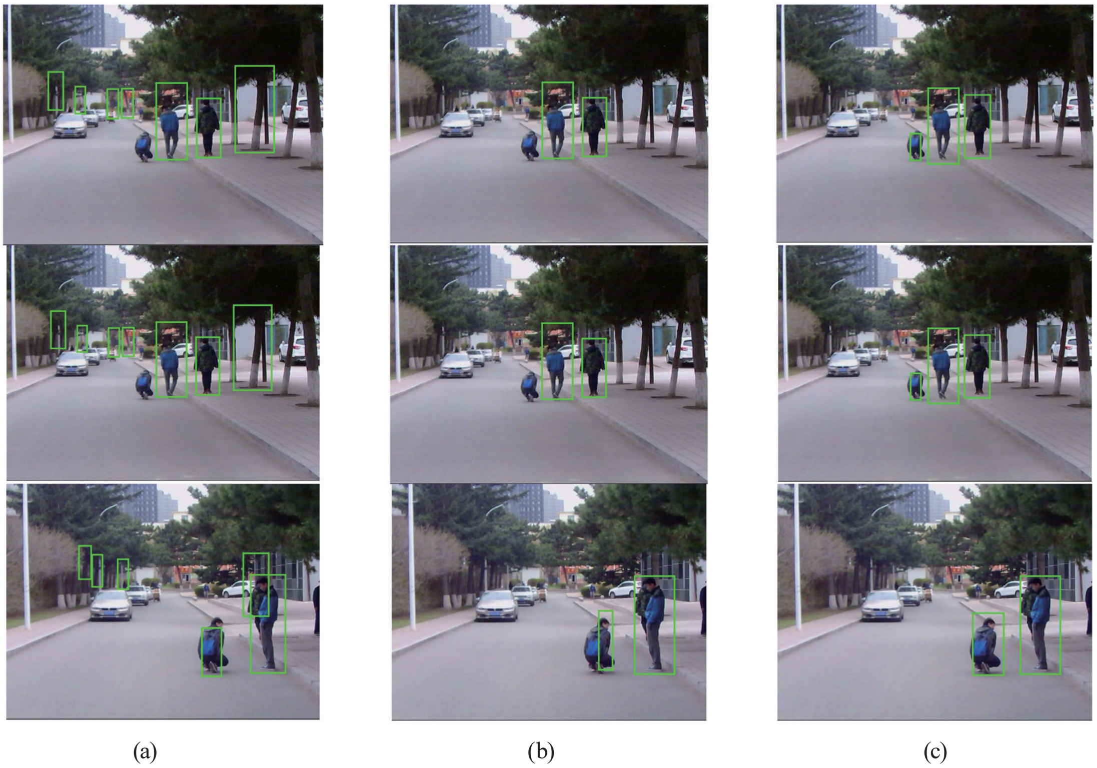
(a) (b) (c)

Fig. 4. (Color online) Video sequences of pedestrians with different scales. (a) ACF. (b) LDCF. (c) CNN–AdaBoost.



(a) (b) (c)

Fig. 5. (Color online) Video sequences of pedestrians with occlusion. (a) ACF. (b) LDCF. (c) CNN–AdaBoost.



(a) (b) (c)

Fig. 6. (Color online) Video sequences of pedestrians with pose variation. (a) ACF. (b) LDCF. (c) CNN–AdaBoost.

Table 3
Statistics of experimental results.

| Algorithm | Miss rate (%) | FPPI | Average detection time per frame (s) |
|---|---|---|---|
| ACF | 16.032 | 3.784 | 0.0611 |
| LDCF | 14.342 | 0.033 | 0.1255 |
| CNN–AdaBoost | 12.778 | 0.021 | 0.0809 |

algorithm has the highest detection speed, but the false detection rate is highest, with the average number of false detections per frame reaching 3.784. The LDCF algorithm has fewest false detections, but the missed detection rate is higher than that of the proposed method and its detection efficiency is low. The average detection time per frame is 0.1255 s. The proposed CNN–AdaBoost algorithm has a lower missed detection rate than the ACF algorithm while maintaining a higher detection speed. The average detection time per frame is 0.0809 s, which is close to the speed of the ACF algorithm.

From the experimental results, it is concluded that, in the proposed CNN–AdaBoost algorithm, the AdaBoost and CNN computations remain independent and parallel. The CNN increases the algorithm complexity compared with those of the ACF and LDCF algorithms, but we optimized the structure of the five-layer CNN by simplifying its input data (using the fast pyramid algorithm in the L color space) and performing feature extraction (using the sliding window method of AdaBoost to obtain candidate regions and obtaining feature vectors from the corresponding CNN output). These improvements enable the real-time performance of the algorithm. Because the probability of misclassification of AdaBoost's strong classifier is high when dealing with features near the top of the pyramid, the negative sample retrieval strategy is used to feed such false positives to the CNN to reidentify them, which overcomes the bottleneck due to the strategies used to improve the AdaBoost classifier performance (increasing the size of the trained pedestrian dataset and increasing the depth of the DT). This also makes the algorithm more robust to complex conditions such as occlusion and deformation.

## 4. Conclusions

In this paper, an AdaBoost classifier is combined with a CNN to realize a novel pedestrian detection method (CNN–AdaBoost). This method uses the fast feature pyramid of the ACF algorithm to calculate the features of each channel. In each layer of pyramid features, the CNN only extracts features of the L color space. Each proposal region is obtained through a fixed-size sliding window in AdaBoost combined with a negative sample retrieval strategy. This avoids the shortcoming of the CNN sliding window of a low efficiency of feature extraction. At the same time, it retains the advantage of the AdaBoost classifier of high efficiency and that of the CNN of strong classification performance. The experimental results show that the method has superior efficiency and accuracy to the ACF and LDCF algorithms in detecting small pedestrian targets, which illustrates the effectiveness of the CNN–AdaBoost method. The proposed method uses a phased training method, which increases the burden of the model training phase, so further methods for improving the method will be explored in future research.

## Acknowledgments

This work is supported by the Scientific Study Project for Institutes of Higher Learning of Liaoning Provincial Department of Education (JP2016018).

## References

1 P. Dollar, C. Wojek, B. Schiele, and P. Perona: IEEE Trans. Pattern Anal. Mach. Intell. **34** (2012) 4. https://doi.org/10.1109/TPAMI.2011.155

2 R. Benenson, M. Omran, J. Hosang, and B. Schiele: Proc. European Conf. Computer Vision (ECCV, 2014) 613. https://doi.org/10.1007/978-3-319-16181-5_47

3 N. Dalal and B. Triggs: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2005) 886. https://doi.org/10.1109/CVPR.2005.177

4 Y. D. Mu, S. C. Yan, Y. Liu, T. Huang, and B. F. Zhu: Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR, 2008) 1. https://doi.org/10.1109/CVPR.2008.4587800

5 P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, and D. Ramanan: IEEE Trans. Pattern Anal. Mach. Intell. **32** (2010) 9. https://doi.org/10.1109/TPAMI.2009.167

6 P. Dollar, R. Appel, S. Belongie, and P. Perona: IEEE Trans. Pattern Anal. Mach. Intell. **36** (2014) 8. https://doi.org/10.1109/TPAMI.2014.2300479

7 R. Girshick, J. Donahue, T. Darrell, and J. Malik: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2013) 580. https://doi.org/10.1109/CVPR.2014.81

8 J. Redmon, S. Divvala, R. Girshick, and A. Farhadi: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2016) 779. https://doi.org/10.1109/CVPR.2016.91

9 P. Dollar, Z. W. Tu, P. Perona, and S. Belongie: Proc. IEEE British Machine Vision Conf. (BMVC, 2009) 1. https://www.sogou.com/link?url=hedJjaC291Me4X18kIp9d36xegpwgIywnAtVxHdC6olJMj6wJ5hp3Vm8XV_Mf75j

10 W. Nam, P. Dollar, and J. H. Han: Proc. 27th Int. Conf. Neural Information Processing Systems (NIPS, 2014) 424. https://www.sogou.com/link?url=hedJjaC291OjP4LRzIQNfnSUkGmFncVVMmLfGjcG8qM.

11 Z. G. Ma and P. P. Gao: Proc. IEEE Int. Conf. Big Data and Smart Computing (BigComp, 2018) 314. https://doi.org/10.1109/BigComp.2018.00053

12 L. L. Zhang, L. Lin, X. D. Liang, and K. M. He: Proc. European Conf. Computer Vision (ECCV, 2016) 443. https://doi.org/10.1007/978-3-319-46475-6_28

13 J. Y. Mao, T. T. Xiao, Y. N. Jiang, and Z. M. Cao: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR, 2017) 6034. https://doi.org/10.1109/CVPR.2017.639

14 W. L. Ouyang and, X. G. Wang: Proc. IEEE Int. Conf. Computer Vision (ICCV, 2014) 2056. https://doi.org/10.1109/ICCV.2013.257

15 S. Q. Ren, K. M. He, R. Girshick, and J. Sun: IEEE Trans. Pattern Anal. Mach. Intell. **39** (2015) 6. https://doi.org/10.1109/TPAMI.2016.2577031

16 E. Ohn-Bar and M. M. Trivedi: Proc. IEEE Int. Conf. Pattern Recognition (ICPR, 2017) 3350. https://doi.org/10.1109/ICPR.2016.7900151

## About the Authors

**Guiyuan Li** received his B.S. and M.S. degrees from Liaoning University of Technology, China, in 2000 and 2006, respectively. Since 2006, he has been a lecturer at Liaoning University of Technology. Since 2013, he has been studying for a doctoral degree at Jilin University. His research interests are system control and simulation and image processing. (261134929@qq.com)

**Changfu Zong** received his M.S. and Ph.D. degrees from Jilin University, China, in 1994 and 1998, respectively. Since 2001, he has been a professor at Jilin University. His research interests are in vehicle system dynamics and vehicle chassis control. (2609582477@qq.com)

**Guangfeng Liu** received his B.S. degree from Shandong Agricultural University, China, in 2015 and his M.S. degree from Jilin University, China, in 2018. Since 2018, he has worked as a software engineer at SAIC Motor Technology Center. His research interests are image processing and computer vision. (614627708@qq.com)

**Tianjun Zhu** received his Ph.D. degree in vehicle engineering from Jilin University of Changchun, Changchun province, China, in 2010. Since 2017, he has been a professor at Zhaoqing University with the Mechanical and Automotive Engineering College in Guangdong province. His research interests include vehicle dynamic control, new energy vehicles, and self-driving vehicle applications. (happy.adam2012@hotmail.com)