



Pedestrian Detection with Improved LBP and Hog Algorithm

Wei Zhou, Suyun Luo

Automotive Engineering College, Shanghai University of Engineering Science, Shanghai, China

Email: 1539465759@qq.com

How to cite this paper: Zhou, W. and Luo, S.Y. (2018) Pedestrian Detection with Improved LBP and Hog Algorithm. *Open Access Library Journal*, 5: e4573. <https://doi.org/10.4236/oalib.1104573>

Received: April 4, 2018

Accepted: April 25, 2018

Published: April 28, 2018

Copyright © 2018 by authors and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This article aims to improve the HOG + SVM pedestrian detection method proposed by previous researchers. The speed of HOG + SVM to detect pedestrians is relatively slow, and the detection accuracy is not very good. This paper proposes a PCA (principal component analysis) dimension reduction for HOG and also interpolates it. The article combines the dimensions of individual HOG features and improves their accuracy, and fuses them with improved LBP features. The features of the fusion of HOG features and LBP features can both express pedestrian profile information and obtain pedestrian texture information. This can improve the speed of pedestrian detection and improve the accuracy of detection, which is beneficial to reduce false detection and missed detection. Although some researchers have combined the two features of HOG and LBP, after simple fusion of these two features, the experimental results show that the detection effect is not much improved. This article is aimed at different formats of video detection material, an application program written on the MFC platform, making pedestrian detection of the material quickly verified, which is conducive to pedestrian detection results data analysis and recording.

Subject Areas

Computer Engineering

Keywords

HOG Feature, Improved LBP Feature, MFC

1. Pedestrian Detection of Specific Programs

As a future trend of smart driving, its complex structure and what can be explored is very much. The article mainly focuses on the pedestrian detection part.

The whole smart car pedestrian detection can be divided into the following several parts: information acquisition part, comprehensive feature extraction, classification training, implementation detection as shown in **Figure 1**.

2. Establish an Improved LBP Model

Figure 2 is a basic LBP feature schematic, which is based on a pixel-by-pixel review of the image. With this pixel as the center, and with this pixel's size value set as a threshold, then its surrounding 3*3 range of pixels is compared (binarized), and its binarization result is specified Arrange regularly to get a set of binary values and use this binary value as the output point for this point. Its definition is shown in (2-1):

$$LBP_{p,R} = \sum_{p=0}^{p-1} S(g_p - g_c) 2^p \quad (2-1)$$

In the formula, $S(x) = \begin{cases} 1, & \text{if } x \geq T \\ 0, & \text{otherwise} \end{cases}$, g_c is the gray value of the center pixel,

g_p ($p = 0, 1, \dots, p-1$) is the grayscale value of p surrounding pixels, and T is the threshold value. For example, for the center in the range of 3*3 in **Figure 2**, its gray value is 68, with 68 as the threshold. He binarizes his eight fields and sets the value of the binarization into a new value, *i.e.*, 10001011, in a clockwise direction from the top left (the order of the specific directions can be self-determined, as long as the laws follow a certain rule). That is, decimal 139 and 139 as output points. After the overall scan is over, there will be an LBP output image. The histogram of this output image is the LBP histogram, which is often used as the recognition feature of the later work and is therefore also called the LBP feature.

Its code implementation in open cv and test results are shown in **Figure 3**.

LBP Rotation Invariant Mode

The basic LBP has good robustness to illumination (*i.e.*, grayscale invariance), but it does not have rotation invariance. Therefore, researchers have extended the above basis and proposed LBP features with rotation invariance. The idea is to make the LBP feature in the circular neighborhood continue to rotate, and then get different LBP eigenvalues, find the smallest LBP value from the rotated LBP eigenvalues, and use this value as the characteristic value of the last center pixel. The specific process is shown in **Figure 4**.

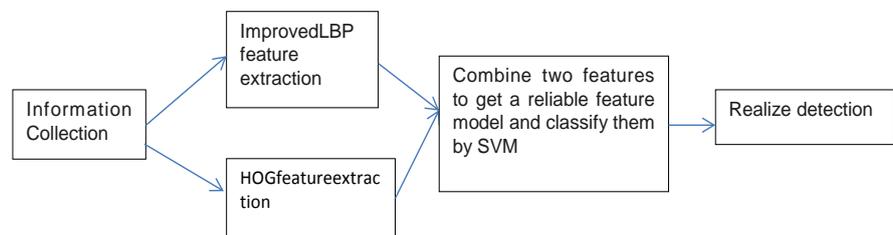


Figure 1. Pedestrian inspection system.

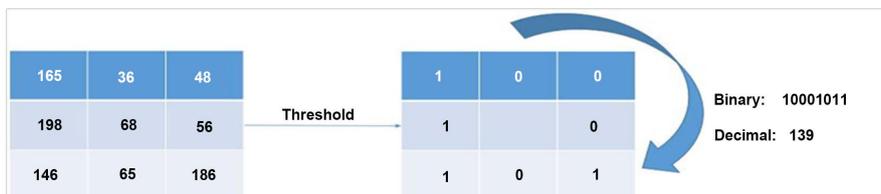


Figure 2. Schematic diagram of hog feature extraction.

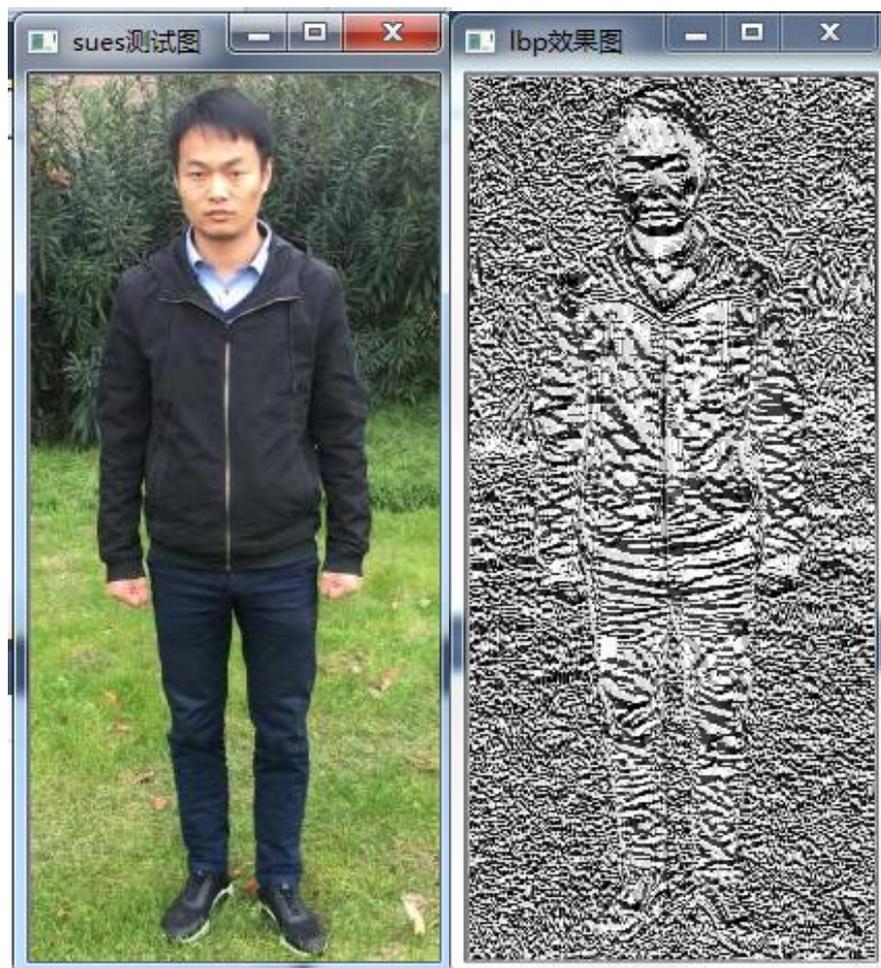


Figure 3. Schematic diagram of the extraction of hog blocks.

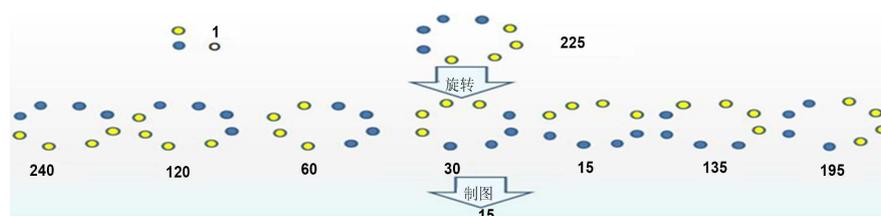


Figure 4. Rotate invariant lbp describe subflow diagram.

Its definition is as follows (2-2).

$$LBP_{Q,R}^{ri} = \min \left\{ ROR(LBP_{P,R}, i) \mid i = 0, 1, \dots, Q-1 \right\} \quad (2-2)$$

Among them, $ROR(x,i)$ means to cycle x to the right by moving the i bit. Here, there is no provision for which point to start from. It can be seen from **Figure 5** that it is clockwise rotated. The rotation-invariant LBP descriptor not only has the robustness of the illumination of the basic LBP descriptor, but also has the advantages of rotation invariance and fewer model types, making the LBP texture more simplified.

3. SVM Classifier

The support vector machine (svm) was proposed by Vapnik and Core [1] based on statistical VC dimension theory and structural risk minimization in 1995. Its advantage lies in its ability to solve small sample, nonlinear and high-dimensional pattern recognition and get good results. The SVM can find a good balance between the learning accuracy of a given training sample (complexity of the model) and the ability to identify its sample (learning ability) as accurately as possible based on a limited sample content. Get the best practicality.

1) Linear separable SVM

The initial development of SVM begins with a linearly separable optimal classification surface. Divide the positive and negative samples in the sample into two parts accurately, and also maximize the separation interval. The SVM strives to obtain a hyperplane that keeps the points in the sample as far away from the face as possible, that is, the area where the largest margin formed by the faces where the positive and negative samples are far apart from each other. Points H_1 , H_2 on the separation plane parallel to the hyperplane and passing the positive and negative samples, such a point (training sample). We call him the support vector. **Figure 5** shows the classification line in the case of linear separability:

2) Linear Inseparable SVM

For linearly inseparable problems we analyze and deal with the following examples. As shown in **Figure 6** below: Define the points in the blue part

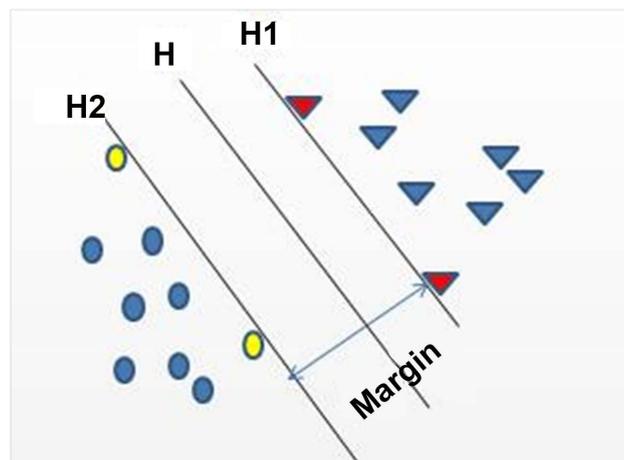


Figure 5. Classification of linear separable cases.



Figure 6. Positive and negative sample set.

between points A and B on the number axis as positive samples, and the points in the yellow parts of both sides as negative samples. A linear function (straight line) in two-dimensional space cannot find a straight line to separate positive and negative samples.

But we can find a curve $g(x) = a_0 + a_1x + a_2x^2$ to separate positive and negative samples, as shown in **Figure 7** below.

Obviously this curve can separate positive and negative samples, but he is not a linear function and is a general quadratic function. In order to make it a linear function, it is rebuilt to define a variable y and b equivalence as (2-3):

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix} b = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} \quad (2-3)$$

Then $g(x)$ can be equivalent to $f(y) = \langle b, y \rangle a$, i.e., $g(x) = f(y) = c_1 y_1 + c_2 y_2 + c_3 y_3$, it can be seen that $g(x)$ becomes The linear function, its difference with the quadratic function is that the dimension becomes higher, and here we get a method that encounters a linearly inseparable sample, trying to increase the dimension of the function, so that it becomes linearly separable. The above is the principle knowledge used in this article.

4. Pedestrian Detection with Fusion IHOGP-LBP Feature Multiple Training

The previous section mainly studied the improved method of HOG algorithm. Through the simplified three-line interpolation and PCA dimension reduction [2] of HOG, the calculation speed of HOG is improved and the accuracy of its detection is also improved. Its effect can be reflected in the following experiments. HOG can describe the edges and gradients of objects very well during feature extraction but lacks description of texture information for some pedestrians. Here we will fuse LBP descriptors, combine pedestrian texture information, and better express pedestrian information through the integration of multi-feature integration graphs, making the detection effect more perfect [3]. Firstly prepare the positive and negative samples, then extract the IHOG features of the positive and negative samples and then reduce the dimension. After training, the IHOGP detector is obtained. Then the negative samples are detected by the detector and then the features of the hard example are extracted, and the IHOGP characteristics before the fusion are obtained. Continue training and eventually get the appropriate detector. The right frame shows the process of extracting LBP. Its process is the same as the model training process on the left [4]. The final result is the

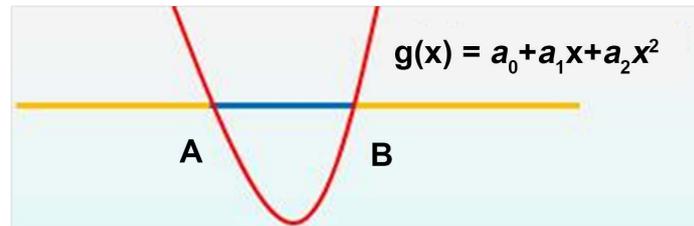


Figure 7. Positive and negative sample set classification.

LBP detection operator. Pedestrian detection of the main line of thought is the middle of the framework of the order, the left and right sides of the middle of the process.

In the specific algorithm, the detection scheme for the fusion IHOGP-LBP feature multiple training is shown in **Figure 8**.

Pedestrian detection of the main line of thought is the middle of the framework of the order, the left and right sides of the middle of the process. It can be seen from the above figure that after the picture is input, the IHOGP feature is extracted first, and then input into the SVM classifier to train to obtain a suspicious pedestrian area, but it is not sure whether it is a pedestrian. The LBP descriptors are then extracted and classified to obtain suspicious positive samples. Finally, the two characteristics are combined to train, and a more reliable pedestrian detector is obtained. Through the last pedestrian detector, the pedestrian in the image is detected, which can accurately detect the location of a person.

Figure 9 is a schematic diagram of IHOGP-LBP feature fusion.

The above figure is the process of feature fusion and can be expressed by Equation (3-1).

$$F_{-}(IHOGP-LBP)(I) = F_{-}IHOGP(I) + F_{-}LBP(I) \quad (3-1)$$

Where I is represented as a sample, $F_{-}IHOGP(I)$ is represented as the $IHOGP$ feature of the sample, and $F_{-}LBP(I)$ is the lbp feature of the sample. The samples are first extracted from the $IHOGP$ features, then the LBP features are extracted, and finally they are combined in parallel to form a fusion feature. From the above figure, we can see that the feature histogram of fusion has become more prominent, which shows that the features of the pedestrian after fusion are more obvious, making the probability of detecting pedestrians even higher.

5. Realization of Video Pedestrian Detection System in Driving Environment

In the above method for pedestrian detection in video, the source code and video file format need to be modified for each scene detection. In practice, it seems to be tedious. This article will develop a simple application program that will make video in various formats quickly available and detect pedestrians.

The application development environment for this article is windows 7, 64-bit operating system, memory 4G, and the processor is Intel(R) core(TM) i5. The

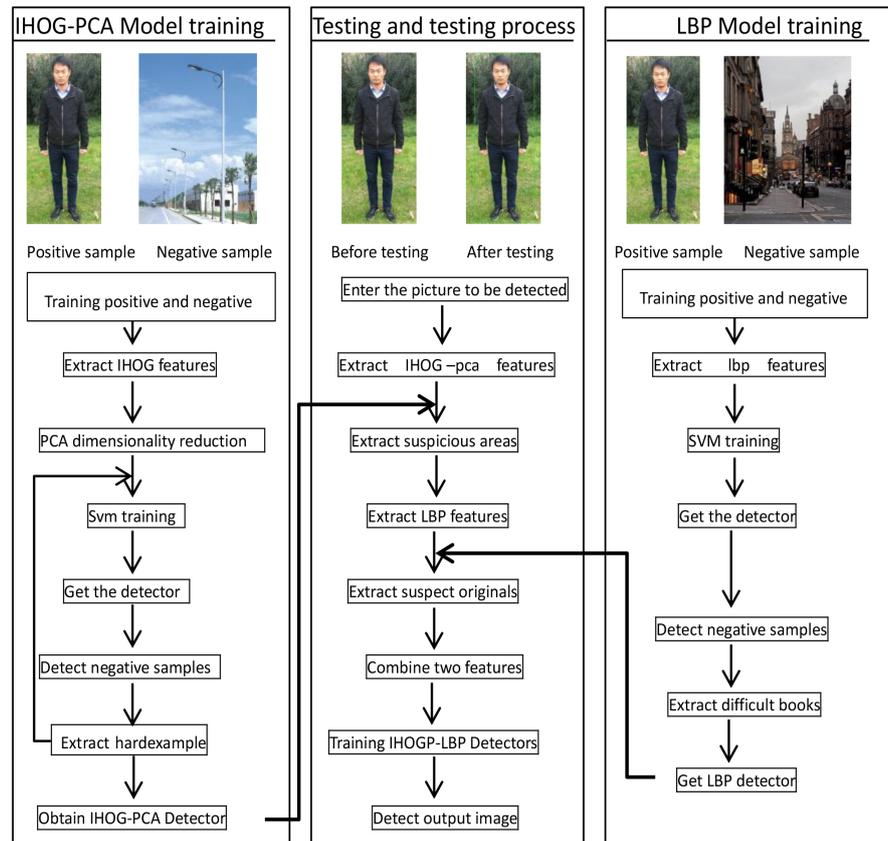


Figure 8. Improved algorithm for pedestrian detection.

developed software is MFC in visual studio 2010. MFC is a packaged windows API library provided by Microsoft Corporation [5]. Its biggest advantage is to provide the framework of the application program. This makes the program developers write their own programs in the existing framework, lost the tedious programming of the underlying program, but also makes programmers quickly familiar with the framework. MFC provides a large number of classes to facilitate different project development. Everything is double-sided, MFC because of the package of a large number of C++ classes, the existence of the package, making a lot of things disappeared. This makes it easier for people who are getting started to understand basic knowledge. This article completes a pedestrian detection application program under the MFC framework. Its main function is to open the video material in the file, and then detect the pedestrians in the material video and pass the window box to the pedestrian [6]. The interface designed in this article is easy to operate. The final rendering is shown in **Figure 10**.

The main class used in this article is CIVSDlg, which contains the video playback dialog [7]. The functions that define many video operations and the corresponding variables are shown in **Figure 11**.

In the “open” control, in addition to opening the video in the file, a message processing program needs to be inserted. The message processing is to enable it to run an image processing program, namely the above-mentioned detector in

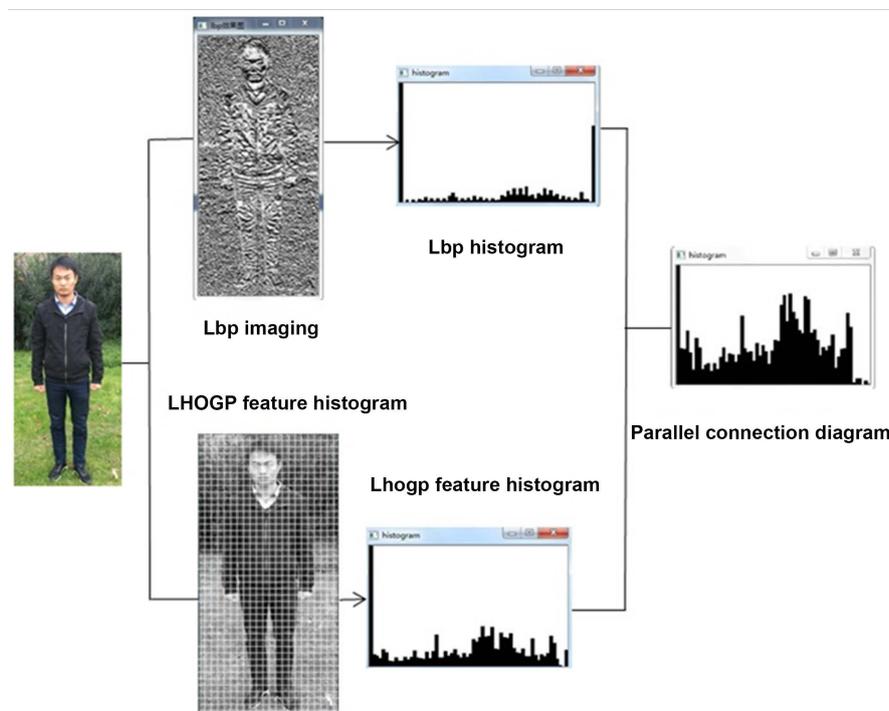


Figure 9. Fusion of IHOGP-IBP features.

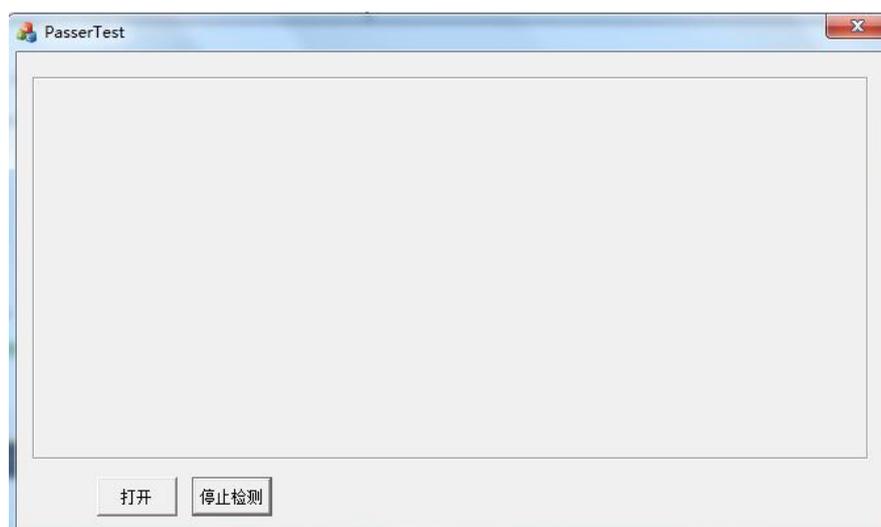


Figure 10. Pedestrian detection APP interface.

the text, so that it can detect pedestrians in the video. This article inserts a pedestrian detection handler and runs the test. Get the results shown in **Figure 12**.

6. Pedestrian Detection Results

The experiment in this paper compares the detection effectiveness of the two detection methods, hog + svm and ihogp-pca + svm, in different scenarios. This article selects various scenarios, and detects the effects of two detection methods in different scenarios. As shown in **Figure 13**.

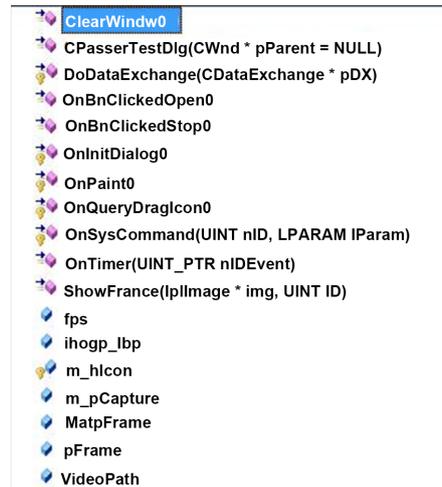


Figure 11. CIVSDlg class diagram.



Figure 12. Application use map.

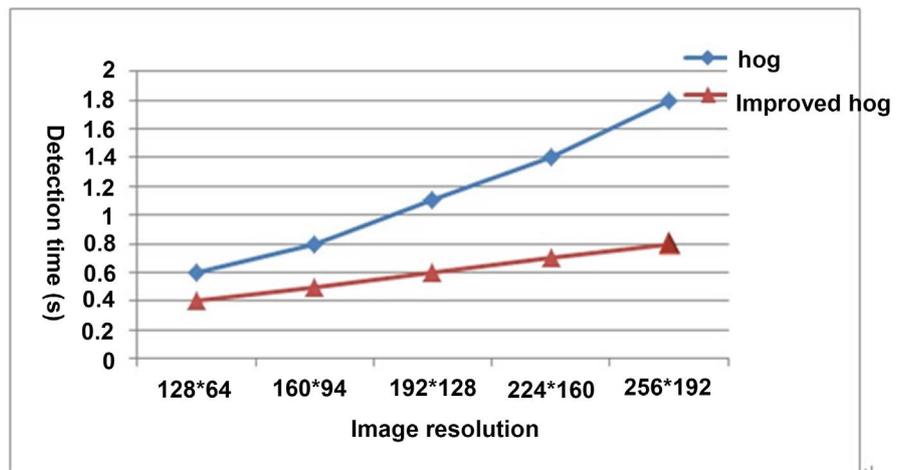


Figure 13. Hog and method detection time comparison.

From the figure above, we can see that in the feature extraction time, with the increase of resolution, the hog extraction time becomes longer and longer, and the improved feature extraction performs better in this aspect without much time extension.

References

- [1] Oren, M., Papageorgiou, C., Sinha, P., *et al.* (1997) Pedestrian Detection Using Wavelet Template. CVPR.
- [2] Xu, D., Li, X. and Liu, Z. (2005) Recognition Letteretal. Cast Shadow Detection in Video Segmentation. *Pattern*, **26**, 91-99.
- [3] Shashua, A., Gdalyahu, Y. and Hayun, G. (2004) Pedestrian Detection for Driving Assistance Systems: Single-Frame Classification and System Level Performance. *Proceedings of IEEE Intelligent Vehicles Symposium*, 1-6.
- [4] Sun, H., Hua, C.-Y. and Luo, Y.-P. (2004) A Multi-Stage Classifier Based Algorithm of Pedestrian Detection in Night with a Near Infrared Camera in a Moving Car. *Proceedings of 3rd IEEE International Conference on Image and Graphics*, USA, 120-123.
- [5] Mikolajczyk, K., Schmid, C., Zisserman, A., *et al.* (2004) Human Detection Based on a Probabilistic Assembly of Robust Part Detectors. ECCV, 69-82.
- [6] Lipton, A., Kanade, T., Fujiyoshi, H., *et al.* (2000) A System for Video Surveillance and Monitoring. Carnegie Mellon University, the Robotics Institute, Pittsburg.
- [7] Tons, M., Doerfler, R., Meinecke, M., *et al.* (2004) Radar Sensors and Sensor Platform Used for Pedestrian Protection in the EC-Funded Project SAVE-U. *Proceedings of IEEE Intelligent Vehicles Symposium*, USA, 813-818.