International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.10 (2016), pp.175-182
http://dx.doi.org/10.14257/ijmue.2016.11.10.17

Pedestrian Detection Algorithm Combining HOG and SLBP

Aili Wang[1], Mingxiao Wang[1], Jitao Zhang[1], Yuji Iwahori[2], and Bo Wang[1]

[1]Higher Education Key Lab for Measuring & Control Technology and Instrumentations of Heilongjiang, Harbin University of Science and Technology, Harbin, China
[2]Dept. of Computer Science, Chubu University, Japan
aili925@hrbust.edu.cn

Abstract

In order to solve the problem of pedestrian detection performance, the described operator was improved. In this paper, semantic local binary pattern (SLBP) and histogram of oriented gradient (HOG) are combined as new feature operator. This feature method would enrich the information and enhance the detection performance. And then histogram intersection kernel support vector machine (HIKSVM) classifier is trained by the augment feature. Because the time cost is too large by the conventional SVM. HIKSVM could make up this drawback, and significantly reduce the training time. The experiments on the INRIA pedestrian dataset show that the method obtained significant improvement in accuracy comparing to HOG descriptors.

***Keywords**: Pedestrian detection; HOG; SLBP; HIKSVM*

1. Introduction

Pedestrian detection has very important applications in video surveillance, content-based image retrieval, video annotation and so on. However, detecting humans in images is a challenging task owing to their variable appearance and the wide range of poses that they can adopt.

Pedestrian detection is to segment the pedestrian outline from the background and locate accurately in each frame of the video sequence. It is a typical challenging task in object detection field. However, owing to high variations of pose, clothing, cluttered backgrounds and partial occlusion handling that people can adopt, the detection task is rather difficult. So it is crucial to extract feature and choose classifier.

Several prevalent features are widely used for pedestrian detection, such as Haar-like feature, Local Binary Pattern(LBP) , Histogram of Oriented Gradient(HOG) and Scale Invariant Feature Transform(SIFT), *etc.* Among these features, HOG is better than other features in pedestrian detection, and usually used to capture the edge or local shape information. The most representative work can be found in [1-5], where overlapped and dense local descriptors based on HOG are extracted trained via Support Vector Machine (SVM), and detected by classifying the images window. It gives significantly higher accuracy on INRIA human database. Moreover, they find that HOG combined with SVM is a better method in a compromise between the runtime and accuracy through great experiments. T. Ojala *et al.* [6] developed Local Binary Pattern(LBP) operator to extract local texture features, that is highly discriminative and its key advantages is namely invariance to rotation [9] and monotonic gray level changes [7]. Wang *et al.* [8] combined HOG and LBP to solve the partial occlusion problem. For the histogram is built according to binary-to-decimal conversion codes, there is no guarantee that semantically similar features must fall into spatially nearby histogram bins, so we adopt semantic LBP(SLBP)to replace basic LBP and combine with HOG feature as the method in this paper.

ISSN: 1975-0080 IJMUE

Support Vector Machine (SVM) and variants of boosted decision trees are two of the leading techniques used in object detection in images, like face recognition [12-13], human detection [1, 15] and Vehicle Identification [14-15]. Although boosted decision trees have faster classification speed, they are significantly slower to train and the complexity of training can grow exponentially with the number of classes. However, linear SVM is widely used in real-time detection due to its fast speed in training and classification. Linear SVM has some special structures that have been exploited to provide super-fast linear SVM solvers, *e.g.*, dual coordinate descent [17-18]. These linear SVM solvers can solve large problems (*e.g.*, with millions of examples) very efficiently (in the order of seconds). Linear SVM is frequently used in text classification, a domain that can easily have millions of feature dimensions. Classification of images need to deal with a large (but not too large) number of feature dimensions (*e.g.*, from thousands to tens of thousands) [19]. Linear SVM usually has lower accuracies than non-linear kernels in this case. So HIKSVM is used in this paper to train and test pedestrian images [20].

The remainder of this paper is organized as follows. In Section 2, an introduction of the proposed algorithm of human detection is provided. Detailed description of HOG, SLBP and HIKSVM is given in Section 3. Section 4 shows the experimental results and Section 5 gives the conclusion.

2. The Overview of the Proposed Human Detection Algorithm

The general pedestrian detection process mainly includes four aspects as follows: selecting the training samples, feature extraction process, classifier training and using the trained classifier to detect. The procedures of our proposed human detection algorithm based on the HOG-SLBP feature are shown in Figure 1.

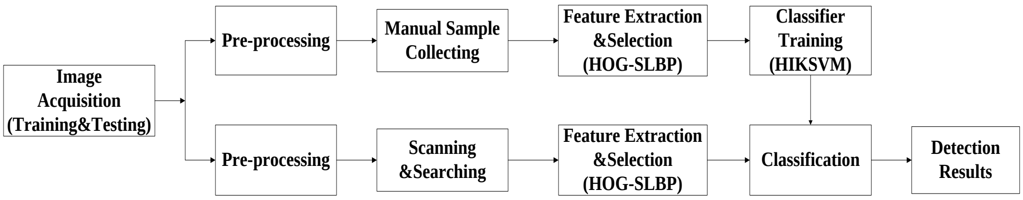


Image Acquisition (Training&Testing)
Pre-processing
Manual Sample Collecting
Feature Extraction &Selection (HOG-SLBP)
Classifier Training (HIKSVM)
Pre-processing
Scanning &Searching
Feature Extraction &Selection (HOG-SLBP)
Classification
Detection Results

Figure 1. The Framework of Pedestrian Detection Algorithm

From the Figure 1, we could see two horizontal flows of these blocks, representing the two phases of any object detection framework, *i.e.* the training phase and the detecting phase. The functions of these blocks are described in detail as follows.

Firstly, according to Figure 1, the image acquisition and pre-processing blocks of the framework collect enough image samples for training, including the positive sample set and the negative sample set. Pre-processing of these images is an important part of the detection algorithm and has great influence for the next procedures. The more information collected from the pre-processing stage, the better performance would be achieved for the whole scheme.

Secondly, this paper adopts the improved HOG-SLBP algorithm to extract features from the training sample set. Although the HOG feature could solve most of the detection problems, there are still some factors that could degrade the performance, such as the noisy background of the humans in the image. We use the mixed features which combines HOG and SLBP to improve the performance, because the latter could make up for some of the disadvantages of the former.

Thirdly, the HIKSVM classifier is adopted in this framework. During the training phase, one classifier is obtained by training these samples. Usually, to get a better classifier, the training phase would cost a lot of time due to the large amount of images. The conventional SVM could achieve high accuracy than other classifiers. But the training time is sometimes difficult to accept. So this paper adopts the HIKSVM, which could reduce the training time and guarantee the same accuracy at the same time.

Then in the detecting phase, this paper uses multi-scale sliding window method [7] to detect pedestrian by this classifier. This method scales the image in multiple scale, then uses the fixed size sliding window to slide on the whole image by an isometric step in the whole image sliding, and detects each sliding window.

3. Detailed Description of HOG Feature, SLBP Feature and HIKSVM Classifier

This section gives a detailed description of HOG feature, SLBP feature and HIKSVM classifier.

3.1 HOG Feature

HOG descriptor characterizes the local gradient amplitude and the feature of the direction. It is not sensitive to the illumination changes and small offset and can effectively depict edge features of the human. Detailed steps for the extraction of HOG features are shown as follows:

The first step is to normalize the whole image. As usual, transform image to grey-scale map, the compression formula as follows(γ=1/2):

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

Then calculate the gradient of the image horizontal and vertical coordinate direction, and gradient direction value of each pixel. Derivative operation can not only capture the contour, figure and texture information, but also further weaken the effect of illumination. The gradient of image pixel (x, y) is calculated via the following formula:

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

Where $G_x(x, y)$, $G_y(x, y)$, H(x, y) separately impress the horizontal direction gradient, the vertical direction gradient and the pixel value of the input image pixel(x, y). Use the following formula to count the pixel(x, y) gradient amplitude and gradient direction:

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2}$$
$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x(x, y)}{G_y(x, y)}\right) \tag{3}$$

Where use [-1,0,1] and $[-1,0,1]^T$ gradient operator to proceed convolution operation on the original image and obtain gradient component of X and Y direction, then the gradient magnitude and direction of the pixel are calculated by the formula above.

Next, construct the gradient direction histogram for each cell unit, and the image is divided into cells which contains 6×6 pixels. Adopt 9-bin histogram to count the gradient information for each cell, that means the gradient orientation of the cell is divided into 9 directions from 0° to 180°. In this way, we can obtain gradient orientations which are independent of the light-dark relationship of the luminance of the target and the background.

Then the cells are combined into blocks, normalize the gradient histogram in blocks, and the normalized block descriptor (vector) is called HOG descriptor. The best setting parameter is 3×3 cells / interval, 6×6 pixels/cell, 9 histogram channels, so the feature dimension is 3×3×9.In the end, collect the HOG features of all overlapping blocks in the detection window, and they are combined into the final feature vector for classification.

3.2. SLBP Feature

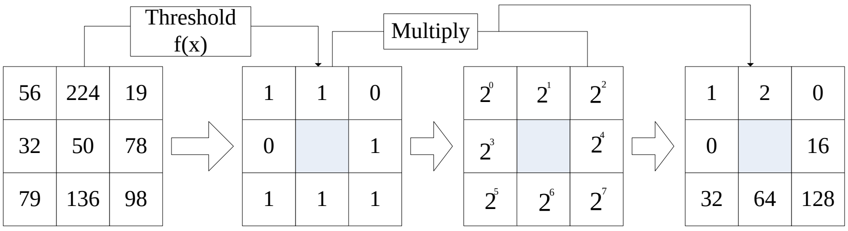
The original LBP operator (OLBP) is defined as: in the 3×3 window, take the center pixel as the threshold and compare the center pixel with the gray value of the adjacent 8 pixels. If the surrounding pixel value is greater than the central pixel value, the position of the pixel is marked as 1, otherwise as 0. Thus, the 8 points in the 3×3 region can generate 8 binary number by comparison (usually converted to a decimal number that is LBP code, in total 256 species), that is the LBP value of the window center pixel, which is used to reflect the texture information of the region. As shown in Figure 2, LBP (x) =1+2+16+32+64+128 =243.

The LBP code value of the center point is calculated using the formula:

$$LBP(x_c, y_c) = \sum_{i=0}^{k-1} f(p_i - p_c)2^i \tag{4}$$

As shown in the formula, p_c is the brightness value of center pixel (x_c, y_c), p_i is the brightness value of the i point in the K adjacent domain. Function f(x) is defined as:

$$f(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{5}$$



Feature 2. Calculation of LBP value

The shortcoming of OLBP texture is that 3×3 neighborhood cannot describe the texture of the large scale structure, so it is extended to different scales of the neighborhood for computation, including using the circular neighborhood and interpolation to calculate the pixel values. ELBP (Extended Local Binary Pattern) can be used for the texture features of different pixel number and different radius circular neighborhood, expressed as $LBP_{K,R}$. K points are selected through uniforming circular neighborhood, R is neighborhood radius, the representation form is related to the rotation angle of the texture. The bigger the K is, the smaller the R is, the more powerful the texture is, the higher the computational complexity is, and the higher the noise effects.

The ELBP operator can generate 2^k different output values, corresponds to 2^k different binary mode patterns. As the image rotates, LBP value also changes. An operator $LBP^{ri}_{K,R}$ with rotation invariance is defined via the following formula:

$$LBP^{ri}_{K,R} = \min\{ROR(LBP^{ri}_{K,R}, i) | i = 0,1, \cdots K-1\} \tag{6}$$

Where ROR(x, i) expresses that x shifts right i bit. Thus we can know that the unified model 00000001 rotates to 000000001, 00000010, 00000100, 00001000, 00010000, 00100000, 01000000, 100000000.

If the local binary pattern meets the following conditions:

a. $LBP_{K,R}$ operator is the rotation invariant $LBP^{ri}_{K,R}$ operator;

b. $U(LBP_{K,R}) \leq 2$;

So call $ULBP^{ri}_{K,R}$ as the texture unified model of $LBP_{K,R}$ (abbreviated as ULBP), which is the basic texture feature. Where function U(x) said the number of 0/1 jumps of the binary code ring through connecting beginning and end of x, such as U(01100000)=2. Replace LBP operator with ULBP operator, the formula is shown as following:

$$ULBP^{ri}_{K,R} = f(x) = \begin{cases} \sum_{i=0}^{K-1} f(p_i - p_c) & U(LBP_{K,R} \leq 2) \\ K+1 & \text{others} \end{cases} \tag{7}$$

By definition, the point with K neighborhood contains K+2 ULBP patterns. These patterns are distributed to their unique label (0-K) according to the number of 1 in each model, other non-uniform patterns of texture can be identified as K+1. The 256 binary patterns are reduced to 10 uniform patterns which can represent the vast majority of texture information, so that the histogram is more compact and less susceptible to noise interference. And the Figure 3 shows the uniform patterns in the (8, R) neighborhood.

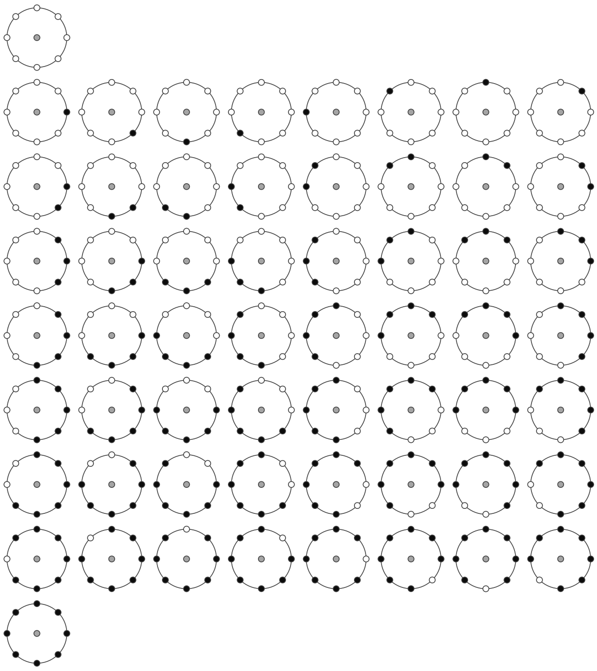
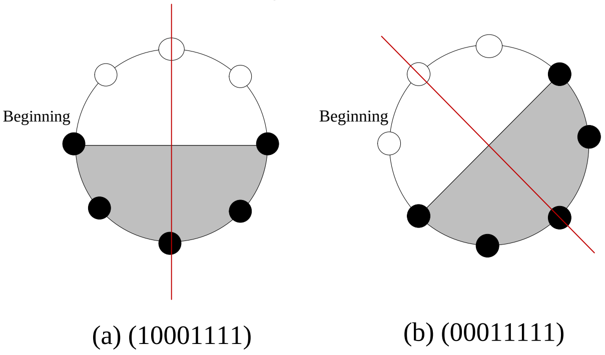


Figure 3. Fifty-Eight Different Uniform Patterns in the (8, R) Neighborhood

These are the introduction of several LBP operators, we will use another LBP operator from a geometric point of view to explain and redefine, that is, the semantic local binary pattern model (SLBP). For the histogram built according to binary-to-decimal conversion codes, there is no guarantee that semantically similar features must fall into spatially nearby histogram bins, and it has huge storage requirement on the other hand. For example, (10001111) anticlockwise shifts left a bit to (0001111), there is a high similarity between them, but the difference between them is far behind after decimal coding.

The semantic local binary pattern is based on the uniform LBP operator; the binary sequence is clockwise represented as circular arc in geometrical sense. Two features of the arc length and the angle of the main shaft replace the decimal encoding number of the basic LBP operator.

As shown in Figure 4, the two binary are consistent, and the axes differ by 45 degrees, which are according with the characteristics of semantic similarity, and the uniform LBP operator has two transitions between 0 and 1. The uniform LBP operator has at least two transitions between 0 and 1, and then it may be a circular arc.



(a) (10001111)   (b) (00011111)

Figure 4. Circular Arc of Uniform Local Binary Pattern Model

Firstly, we will operate binaryzation on the color image space. Then calculate the arc length and the angle of the main shaft, and give up the non-uniform pattern. Finally, carry out the transformation between the matrix and the vector, and get the one dimensional vector through connecting each column of the matrix. Figure 5 describes the semantic implications of SLBP.

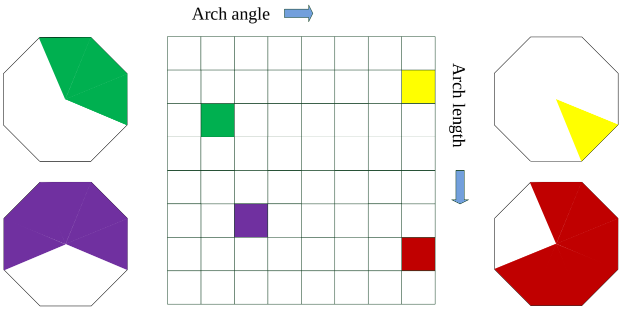


Figure 5. Implications of SLBP

From Figure 5, it can be seen that, in the form of geometry, four different binary sequences are represented, which are expressed as two dimensional matrix, respectively, the arc length and the angle of the main shaft are used as variables, which are expressed as two dimensional matrix.

3.3 The Description of HIKSVM

We define two n-dimensional histogram $A, B \in \mathbb{R}^n$, a_j and b_j are j dimensional vectors , so HIK is expressed as following:

$$K_{HIK}(A, B) = \sum_{j=1}^{n} \min(a_j, b_j) \tag{8}$$

The more $K_{HIK}(A, B)$ is, the more similar histogram A and histogram B are, otherwise the A and B are more different. HIKSVM utilizes the discrimination function as following to classify the sample $x \in \mathbb{R}^n$.

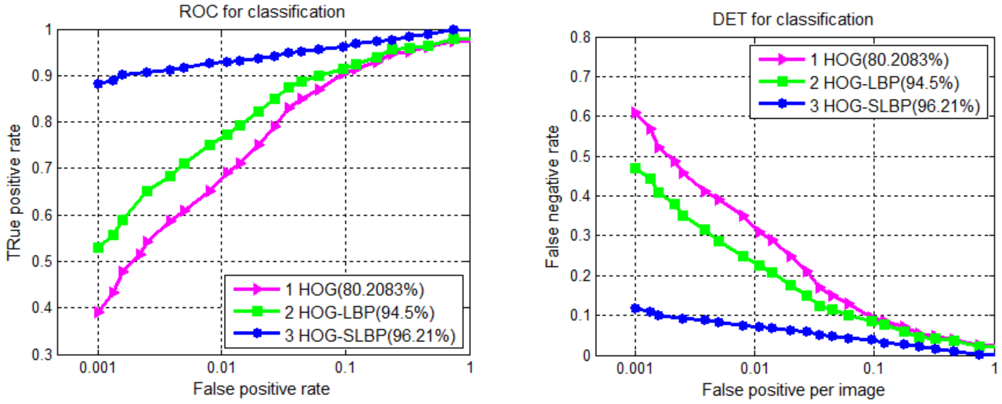
$$f(x) = \sum_{s=1}^{m} a_s y_s \left(\sum_{i=1}^{n} \min(x(i), x_s(i))\right) + b \tag{9}$$

It can be seen that the time and storage complexity are O(m×n) for HIKSVM to classify the samples. When either of the feature dimension n or the dimension m of the support vector is arbitrarily large, the amount of computation and the required storage space will become especially large.

4. Experimental Results and Analysis

This paper combines HOG feature with SLBP feature as the feature vector, and uses HIKSVM as classifier to detect pedestrian. Besides, it implements the proposed algorithm on the INRIA dataset, which provides the original images with high resolution and the corresponding label files, and each image size is 64×128 pixels. In this experiment, it contains 614 positive samples (including 2416 pedestrians) and 1218 negative samples for training, 288 positive samples (including 1126 pedestrians) and 741 negative samples for testing. In order to verify the effectiveness of the proposed method for human detection application, this paper uses the ROC (the Receive Operating Characteristic) and DET(the Detection Error Tradeoff) curves to evaluate. The detection results are shown in Figure 6, which compare the classification performance between HOG detector, HOG-LBP detector and HOG-SLBP detector.

Figure 6(a) gives the ROC curve using different methods, the detection performance of HOG-SLBP in this paper is better than others. It uses 0.01 false positive rate as a reference point, the true positive rate of HOG-SLBP is increased than HOG, HOG-LBP by 27.9%, 15.5%. As shown in Figure 6(b), obviously low false negative rate together with low FPPI is favorable in the DET curve. Moreover, HOG, HOG-LBP, HOG-SLBP respectively achieve the accuracy of classification 80.2083%, 94.5%, 96.21%. Therefore, the combination of HOG-SLBP with complementary characteristics significantly improves the performance of a single feature detector.



(a)ROC (b) DET

Figure 6. Detection Performance Using Three Different Features

5. Conclusion

In order to improve the pedestrian detection performance, this paper extracts HOG features and SLBP features, and uses HIKSVM classifier to train and classify pedestrians. Compared to other state of the art algorithm, our approach has stronger discriminative power and high detection rate in the INRIA pedestrian database. We will make further improvement to deal with complex background and occlusion situations.

References

[1] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection”, Computer Vision and Pattern Recognition, vol. 1, **(2005)**, pp. 886-893.

[2] L. Sun, G. Liu and Y. Liu, “Multiple pedestrians tracking algorithm by incorporating histogram of oriented gradient detections”, IET Image Processing, vol. 7, **(2013)**, pp. 653-659.

[3] J. Li, Y. Zhao and D. Quan, “The combination of CSLBP and LBP feature for pedestrian detection”, 2013 3rd International Conference on Computer Science and Network Technology (ICCSNT), **(2013)**, pp. 543-546.

[4] P. Dollar, R. Appel, S. Belongie and P. Perona, “Fast Feature Pyramids for Object Detection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, **(2014)**, pp. 1532-1545.

[5] A. Satpathy, X. Jiang and H. L. Eng, “LBP-Based Edge-Texture Features for Object Recognition”, IEEE Transactions on Image Processing, vol. 23, **(2014)**, pp. 1953-1964.

[6] T. Ojala, M. Pietikinen and D. Harwood, “A comparative study of texture measures with classification based on feature distributions”, Pattern Recognition, **(1998)**, pp. 51-59.

[7] X. Zhihua, “Infrared face recognition based on LBP co-occurrence matrix”, Control Conference (CCC), **(2014)**, pp. 4817-4820.

[8] X. Wang, T. X. Han and S. Yan, “An HOG-LBP Human Detector with Partial Occlusion Handling”, Computer Vision, **(2009)**, pp. 31-39.

[9] G. Zhao and T. Ahonen, “Rotation-Invariant Image and Video Description with Local Binary Pattern Features”, APRIL, vol. 21, **(2012)**, pp. 61-70.

[10] Z. Yin and J. Liu, “Introduction of SVM algorithms and recent applications about fault diagnosis and other aspects”, Industrial Informatics (INDIN), **(2015)**, pp. 550-555.

[11] C. H. Lampert, M. B. Blaschko and T. Hofmann, “Beyond sliding windows: Object localization by efficient sub-window search”, Computer Vision and Pattern Recognition, CVPR, **(2008)**, pp. 1-8.

[12] H. Tan, B. Yang and Z. Ma, “Face recognition based on the fusion of global and local HOG features of face images”, Computer Vision, IET, vol. 8, no. 3, **(2014)**, pp. 224-234.

[13] P. Viola and M. J. Jones, “Robust real-time face detection”, IJCV, vol. 57, no. 2, **(2004)**, pp. 137-154.

[14] A. E. Ghahnavieh and A. A. Shahraki, "Enhancing the license plates character recognition methods by means of SVM", Electrical Engineering (ICEE), **(2014)**, pp. 220-225.

[15] S. H. Lee, M. Bang, K. H. Jung and K. Yi, "An efficient selection of HOG feature for SVM classification of vehicle", Consumer Electronics (ISCE), IEEE International Symposium on, **(2015)**, pp. 1-2.

[16] A. Satpathy and X. Jiang, "Human detection using Discriminative and Robust Local Binary Pattern", IEEE International Conference, **(2013)**, pp. 2376-2380.

[17] C. J. Hsieh, K. W. Chang, C.-J. Lin, S. S. Keerthi and S. Sundararajan, "A dual coordinate descent method for large-scale linear SVM", Proceeding Int. Conf. Mach. Learn, **(2008)**, pp. 408–415.

[18] S. Zhang, X. Yu, Y. Sui, S. Zhao and L. Zhang, "Object Tracking with Multi-View Support Vector Machines", IEEE Transactions on Multimedia, vol. 17, **(2015)**, pp. 265-278.

[19] W. Zhong, L. Ma and X. He, "Rice Appearance quality recognition based on PCA and Improved BP Neural Network", Journal of Harbin University of Science and Technology, vol. 4, **(2015)**, pp. 76-81.

[20] J. Wu, "Efficient HIK SVM Learning for Image Classification", IEEE Transactions on Image Processing, vol. 21, **(2012)**, pp. 4442-4453.