Gradient Local Binary Patterns for Human Detection

Ning Jiang, Jiu Xu, Wenxin Yu and Satoshi Goto

Graduate School of Information, Production and Systems, Waseda University

Waseda University, Fukuoka, Japan

jiangning@ruri.waseda.jp

***Abstract*— In recent years, local pattern based features have attracted increasing interest in object detection and recognition systems. Local Binary Pattern (LBP) feature is widely used in texture classification and face detection. But the original definition of LBP is not suitable for human detection. In this paper, we propose a novel feature set named gradient local binary patterns (GLBP), Original GLBP and Improved GLBP, for human detection. Experiments are performed on INRIA dataset, which shows the proposal GLBP feature is more discriminative than histogram of orientated gradient (HOG), histogram of template (HOT) and Semantic Local Binary Patterns (S-LBP), under the same training method. In our experiments, the window size is fixed. That means the performance can be improved by boosting and cascade methods. And the computation of GLBP feature is parallel, which make it easy for hardware acceleration. These factors make GLBP feature possible for real-time human detection.**

I. INTRODUCTION

After face detection techniques have become practical, human detection in images and still videos is becoming a focus research topic in computer vision. Human detection is widely used in image analysis, intelligent vehicle and visual surveillance. Recently, there are two types of popular approaches for human detection: part based methods [10] and sub-window based methods [1, 3, 5, 6]. Our method belongs to the second type. In this sub-window based method, each image is scanned from the top left to the bottom right with a invariable size sliding rectangle windows in different scales. For each sliding window, features are extracted and fed to the classifier which is trained using the training data. The classifier will give a value by the computation between the features from the sliding window and the data inside the classifier. By comparing this value with a threshold, the classifier divides each sliding window into two types depending on whether it contains a pedestrian or not. Compared with the part based detectors, the computation and time consumption of sub-windows based methods is less, but part based methods can get high efficient detection rate in occlusion case. Because the feature with the occlusion part will cause the classification result deteriorated.

There are two directions for the sub-window based method: building powerful features [1, 3, 5, 6] and developing powerful learning algorithms [7, 8, 9]. The gradient information is efficient in feature extraction for human detection. Some gradient based features are proposed in [1, 2, 3, 4]. The texture information is also efficient in human detection. Original Local Binary Pattern (LBP) is not suitable for human detection. Some improved LBP based features are proposed in [5, 6] for human detection. SVM method [8] and boosting methods [9] are two popular training methods in human detection. In SVM method, the training is simple. And SVM based detection can be easily accelerated by GPU and Multi-core in feature extraction part. Boosting methods [7] are often used in cascade detector. It has a good performance in combining a strongly classifier with lots of weak classifier. And these detectors can save lots of detecting time by discard the background windows rapidly.

II. FEATURE EXTRACTION

A. Limitation of Other Previous Features

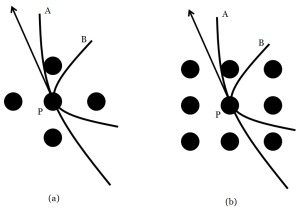


Figure 1. The disadvantages of gradient-based methods

The gradient-based features (HOG and COV) only use the value of 4 neighbor pixels to calculate the edge direction and magnitude. Seen from Figure1, it can't get powerful feature values to classify the curve A and B with different radians. But sometimes these differences within A and B is important and efficient in feature extraction. LBP based methods can get more powerful feature in this case. This is the first disadvantage of HOG. The second disadvantage of gradient-based feature is disturbed from the noise. One pixel noise will cause the feature of nearly pixels change a lot. It may vote a big value to the histogram. But in S-LBP and other Uniform-LBP features (uniform patterns will be defined in section 2.3), this noise can be removed by uniform checking. When a noise appears near a pixel, the pattern will changes to a non-uniform pattern and will not vote to the histogram. In HOG feature, the gradient values of pixels are vote to the histogram as weight but in LBP the same weight are voted. And the pixels with big magnitude value always play important part in the classification. But LBP and HOT features can't get enough information from these important pixels with big gradient values.

978-1-4673-5762-3/13/$31.00

Beside the advantages and disadvantages in gradient features and texture feature, we proposed our Gradient Local Binary Pattern (GLBP) which is extracted by combining gradient information and texture information together.

B. *Feature Extraction of Original GLBP Feature*

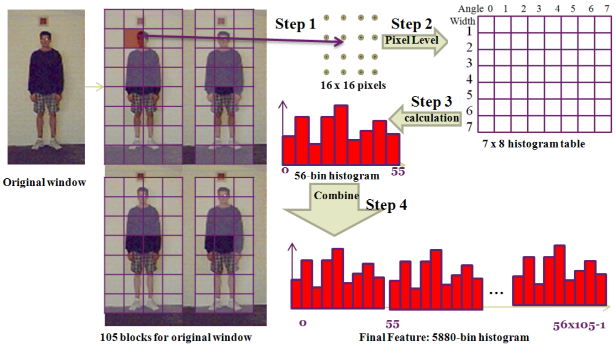


Figure 2. Feature extraction of Original GLBP feature

The flow of feature extraction for GLBP feature is shown in Figure 2. We get a 56x105 dimensional vector for each sample window. There are 4 steps in GLBP feature extraction method.

In **step 1**, the input sample window is cut into several blocks. Here the size of each input sample is fixed at size 66 by 130. One line width pixels around the middle 64x128 image are added for computing the LBP binary codes and gradient values as neighbor pixels for the verge pixels of the middle 64x128 image. We get 105 blocks in this step. The size of each block is 16 by 16. The stride between two neighbor blocks is 8.

In **step 2**, we calculate the gradient value and GLBP table position value (angle value and width value) for each pixel in one block. The flow of calculation is shown in Figure 3.

In **step 3**, we calculate the 56 bins histogram by adding the 56-dimensional vectors from 16 x 16 pixels by step 2 in each block. After we get this 56 bin histogram for each block, we do the L2 normalization on 56 values in the histogram in each block like HOG [1] and HOT [6].

In **step 4**, we get a 56x105=5880 bins histogram by put 105 blocks together. The final 5880 dimensional vector from this 5880 bin histogram is the final GLBP feature.

Figure 3 shows the pixel level of feature extraction of original GLBP feature which is the step 2 of feature extraction.

In step A, the values of original pixel and its eight neighbor pixels are read from the memory first. Then we calculate the eight bit binary code by comparing the value of middle pixel with the values of eight neighbor pixels one by one. The binary value is "0" when the value of middle pixel is bigger, "1" when the value of neighbor pixel is bigger. After the calculation of 8-bit binary code, we get 1 pattern from 2^8=256 different patterns for each pixel. Then we put the neighbor pixel with same binary value together to get several "1" area and "0" area. When the "1" area and "0" area appear only once, we call it Uniform Pattern. For example, binary pattern "00111000" is a uniform pattern. Patterns "00000000" and "01110101" are non-uniform patterns in our definition. The pixel with patterns "00000000" and "11111111" are treated as a noise pixel, so these two patterns are ignored in GLBP feature. Fifty six uniform patterns are selected from 256 patterns in this uniform check part. We do the next step if the binary pattern of this pixel is uniform. If not, we ignore this pixel and calculate the next pixel.

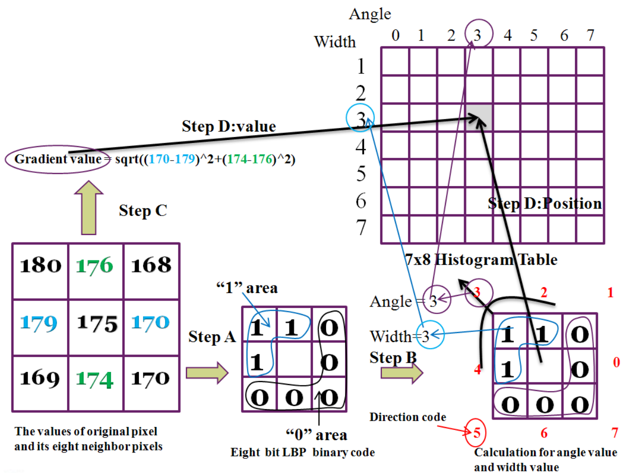


Figure 3. Feature extraction of Original GLBP feature for each pixel

In step B, we calculate the width value and angle value for the pixel with uniform pattern. Width value is the number of value "1" in the binary code of this pixel. Eight direction codes with 0 to 7 are defined in the direction of eight neighbor pixels. Angle value is the direction code of the middle pixel in "1" area of its binary code. When the width value is an even number, we set the angle value at the smaller value of these two direction values except the middle direction of "1" area is between direction "7" and direction "0". We set the angle value at "0" in this case.

In step C, we calculate the gradient value by the value of original pixel and the values of its 4 neighbor pixels as HOG in [1].

In step D, angle value and width value from step B are used for mapping the position of bin in GLBP Table. Then we write the gradient value from step C into this bin of the GLBP Table. At last, we get a 56-dimensional vector by this GLBP table. Only the value of one element in this 56-dimensional vector is non-zero. The values of other 55 elements are zero.

C. *Improved GLBP Feature*

We add two functions in feature extraction to improve the performance.

Patterns cutting: In the definition of original GLBP feature, only the feature of uniform pixels are calculated and extracted to the final feature, and the feature of non-uniform pixels are ignored, which will lose some information for two corners adjacent pixels. So we add this function to add more information in feature extraction. The non-uniform pattern which has two "1" areas and two "0" areas is cut into two uniform patterns as shown in Figure.4(a). The non-uniform pattern which has three "1" areas and three "0" areas is cut into three uniform patterns as shown in Figure.4(b). The non-uniform patterns "00000000", "11111111", "01010101" and "10101010" are ignored.

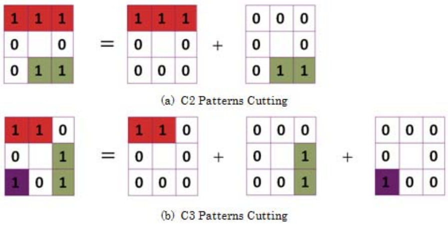
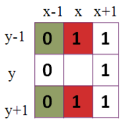


Figure 4. Patterns cutting for Improved GLBP Feature

Gradient value formula: In the definition of original GLBP feature, the gradient value formula is the same as HOG feature [1]. In GLBP feature, we can get the width and angle value for each uniform patter. And we can get a more powerful gradient value using this information.



$$Value(x,y)=\sqrt{(P(x,y-1)-P(x-1,y-1))^2+(P(x,y+1)-P(x-1,y+1))^2}$$

Figure 5. Patterns cutting for Improved GLBP Feature

So we proposed to use four pixel values in the "1-0" and "0-1" transitions as shown in Figure.5 to calculate the gradient value.

III. TRAINING METHOD

Training method is very import for the detection result. Boosting methods [9] are used for selecting some powerful value in the giving vector. SVM method [8] can make the training easily and achieve enough detection rate by using its different kernel types. SVM is effective for learning with small sampling in high-dimensional space. The decision rule is given by the following formula:

$$f(x)=\sum_{i=1}^{N_s}\beta_i K(x_i,x)+b \quad (1)$$

Where x_i are supported by support vector, N_s is the number of support vectors, $K(x,y)$ is the kernel function. The most popular functions are given by the following formulas:

Linear: $K(x,y)=x'\times y$ (2)

Radial basis function: $K(x,y)=\exp\left(-\gamma\times|x-y|^2\right)$ (3)

Figure 6. Some samples in INRIA dataset

In our experiments, the tools from LibSVM [13] are used for training. In our training, we set the training parameters as the default values in LibSVM [13].

IV. EXPERIMENT

A. Dataset

The training and testing dataset we used in the experiment is INRIA dataset [14]. It is widely used for human detection in still images. There are 1774 human annotation and 1671 person free image. 1208 human annotation and 1218 non-human images are used for training the detectors, the left are used for testing. For positive images, left-right reflections are used for getting 2416 human images from 1208 for training. This dataset include the samples with different clothing, pose, occlusion and illumination. So it is difficult for human detection and suitable for compare the different methods. At the same time, this dataset is suitable for pedestrian detection. Some samples in INRIA dataset are shown in Figure.6.

B. Training

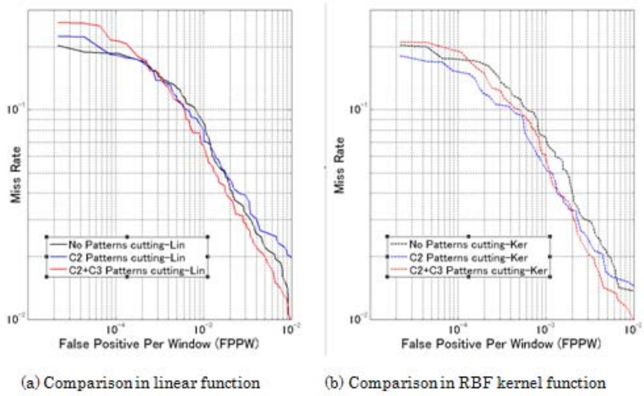


Figure 7. Different patterns cutting types for GLBP feature

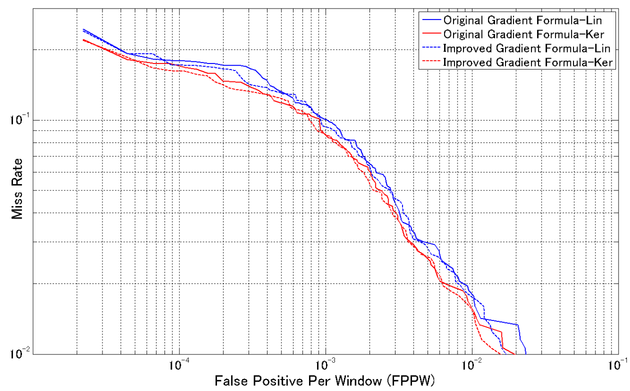


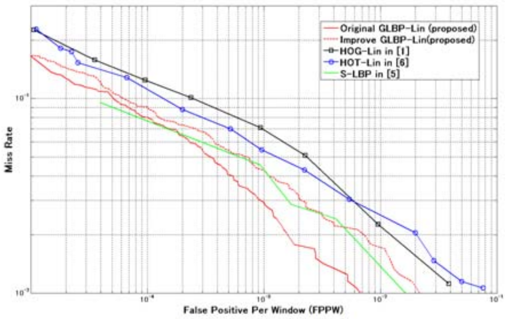
Figure 8. Different Gradient Formula for GLBP features

In the experiments, OpenCV [12] is used for image process, like image scale and matrix calculation. LibSVM [13] is used for training detectors. We use the same strategy as HOG in [1] in our experiments. The re-sample strategy and normalization strategy are also used. The block size is 16 by 16 and the stride between two blocks is 8 which are the same as HOG [1]. In the experiments of training parameter selection, we train the detectors without resample method. The detectors are trained by using different normalization types or different patterns cutting functions and same data. We find the best training parameters for training the final strong detectors.

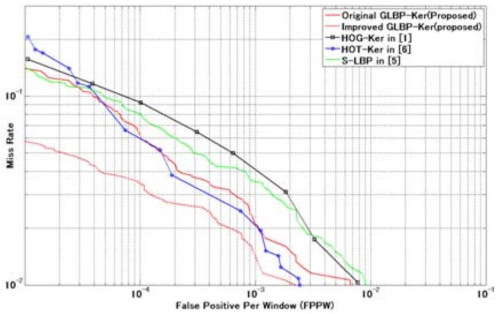
Patterns cutting: We trained the linear and RBF detectors using C2 patterns cutting and C2+C3 patterns cutting types. The result is shown in Figure.7. From Figure.7, we can see that improvement of patterns cutting for RBF kernel function is bigger than it for linear function. And C2 patterns cutting achieves better result than C2+C3 patterns cutting in low FPPW part, especially in the comparison of RBF kernel functions. So we choose C2 patterns cutting to train our final Improved GLBP detectors.

Gradient formula: We did the experiment to compare two gradient formulas. The result is shown in Figure.8. We can found that Improved Gradient Formula is better than Original Gradient Formula.

C. *Comparison with other features*



(a) Comparison using linear kernel



(b)Comparison using RBF kernel

Figure 9. Comparison with other features using linear and RBF kernel

Figure 10. Some detection results on natural images

In this part, we compare the detection rate of our proposed GLBP features with HOT, HOG and S-LBP features. In the comparison of linear detectors, we only compare our feature with HOG, S-LBP and HOT features. The results in Figure 9.(a) shown that our Original GLBP is better than HOG, S-LBP and HOT features. In the comparison of kernel (RBF) detector, the results in Figure 9.(b) shown that our Improved GLBP feature achieves the best detection rate in the comparison with HOG, HOT and S-LBP features. The result in section 4.2 shows that two additional functions for Improved GLBP, Patterns cutting and improved gradient formula, can improve the performance in RBF kernel function. These reasons make our Improved GLBP feature achieves the best detection rate in Figure 9.(b). Finally, some detection results on natural images by our Improved GLBP detector can be seen in Figure.10.

V. CONCLUSION

In this paper, a new set of features named Original Original-GLBP and Improved-GLBP are proposed for human detection. The results show that our Original-GLBP based linear detector achieves the best detection rate in the comparison with other linear detectors. And the computation complexity and time consumption of linear detector is the shortest in SVM based detector. So our Original GLBP feature is useful in some rapid detection cases. In the comparison of kernel detectors, our improved GLBP detector gets the best result. In our experiments, the window size is fixed. That means the performance can be improved by boosting methods. The computation of GLBP feature is parallel, which make it easy for hardware acceleration. These factors make it possible for real-time pedestrian detection.

ACKNOWLEDGMENT

This research is partly supported by Grants-in-Aid for Scientific Research <KAKENHI>.

REFERENCES

[1] N. Dalal and B. Triggs. "Histograms of oriented gradients for human detection", Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[2] Q.Zhu, S.Avidan, M.C.Yeh, and K.T.Cheng, “Fast human detection using a cascade of histogram of oriented gradients,” IEEE Conf. on Computer Vision and Pattern Recognition, pp.1491-1498, New York, 2006.

[3] D.Lowe, “Distinctive image features from scale-invariant key points,” IJCV, vol.60, pp.91-110, 2004.

[4] T.Oncel and P. Fatih, “Pedestrian detection via classification on Riemannian manifolds,” IEEE Trans. Pattern Anal. Mach. Intell., vol.30, no.10, pp.1713-1727, Oct. 2008.

[5] M.Yadong and Y.Shuicheng, “Discriminative local binary patterns fir human detection in personal album,” CVPR, 2008.

[6] S.Tang and S.Goto, “Histogram of Template for Human Detection” *In International Conference on Acoustics, Speech and Signal*, 2010.

[7] P.Viola and M.Jones, “Rapid object detection using a boosted cascade of simple features,” Conference on Computer Vision and Pattern Recognition, 2001.

[8] B.Scholkopf and A.Smola, Learning with Kernels Support Vector Machines, Regularization, Optimization and Beyond, 2002.

[9] J.Friedman, T.Hastie, and R.Tibshirani, “Additive logistic regression: A statistical view of boosting,” Ann.Stat, vol.28, pp.337-407, 2000.

[10] A. Hadid, M. Pietikäinen, and T. Ahonen, “A discriminative feature space for detecting and recognizing faces,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 797–804, 2004.

[11] T.Ojala, M. Pietikäinen and D.Harwood. “A comparative study of texture measures with classification based on featured distributions,” in Pattern Recognition, 29(1):51-59, 1996.

[12] Open Computer Vision Library(Opencv), by Intel Corporation.

[13] LibSVM [Online], http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[14] INRIA Dataset [Online], Avilable: http://lear.inrialpes.fr/data