Research of Cues Selection in Pedestrian Detector based on Adaboost

Zhao Chunhui, Liang Yan, Cheng Yongmei, Lu Zhaoxia
College of Automation
Northwestern Polytechnical University
Xi'an, China
zhaochunhui@nwpu.edu.cn

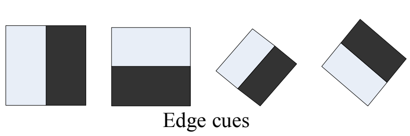
***Abstract*—Which cues to be used in describing pedestrian are the key to detect pedestrian with adaboost algorithm. In this paper we presented 4 triangle cues and 16 complex cues by researching on pedestrian's figures and proposed the way to calculate these cues' sum. To evaluate this calculating method, we proved the relation of iterative times with error and time spending. Compared with rectangle and asymmetry cues, triangle and complex cues are better in describing local characteristics of people's appearance and designing pedestrian detector. The experiment indicated that triangle and complex cues can be used in constructing robust classifier to detect pedestrian.**

***Key words*—Triangle cues, complex cues, adaboost, pedestrian detector, cue selection**

I. INTRODUCTION

The pedestrian detector based on adaboost[1,2,3,4] is very popular in video based object detection. The key of this detector are cue chosen and weak classifier design. It is very important to choose cue properly to improve the pedestrian detecting efficiency especially in small target and low SNR conditions. In order to gain better spacial identifiability, Papageorgiou[5,6] adopted 4 nonstandard Harr cues to detect obverse face and body of human, as figure 1 shown. This cues were availability for face detection, but they were not good for pedestrian detection because that they could not describe the human exactly. Rainer Lienhart et al[7,8] rotated the Papageorgiou's cues for 45 degree and proposed the Harr-like cues which be divided into edge, line, centre circle and diagonal cues, as figure 2 shown. Those cues had large amount and were very complex in calculation, so they could not be used widely. Zhu[9] proposed the asymmetry cues, as figure 3 shown. Those cues were constructed based on the figure of human and could describe the edge of human figure with rectangle cues efficiently. But they would not work effectively alone.

Figure 1. Rectangle cues



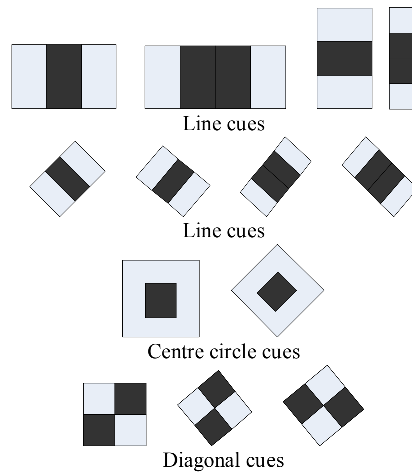


Figure 2. Extended rectangle cues

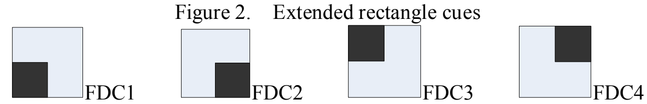


Figure 3. Asymmetry cues

The rectangle cues were proposed for face detection because those cues could describe the symmetrical face effectively. But they could not work when the pedestrian with different clothes and in small size in video. We analyzed the pose of pedestrian in video especially the edge cue of pedestrian and found that the triangle cues could describe human shape effectively. Moreover, we combined the triangle and rectangle cues into complex cues which had good performance in pedestrian detection. Finally, we compared those cues in adaboost algorithm to open out the merits of the proposed cues.

II. TRIANGLE CUES

We found that the acclivitous edges existed abundantly in the front, side and back of human aspect as figure 4 shown. Those acclivitous edges were found at the leg, arm and body of pedestrian and could not be described perfectly by rectangle and asymmetry cues. So we proposed the triangle cues to describe the acclivitous edges as figure 5 shown.

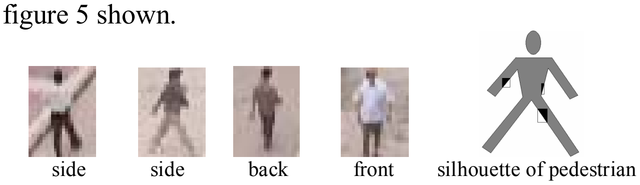


Figure 4. The poses of pedestrian

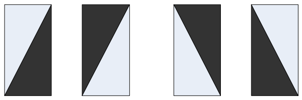


Figure 5. The four types of triangle cues

The integral image[1] was used to calculate the sums of rectangle cues. Rainer Lienhart [7] proposed the equation is:

$$feature = \sum_{i=1}^{N} \omega_i \, \mathrm{Re}cSum(r_i) \tag{1}$$

In equation (1), N is the number of the rectangle blocks in center of rectangle cue. ω_i is the weight of the corresponding rectangle block. $\mathrm{Re}cSum(r_i)$ is the sum of pixels in the corresponding rectangle block. Generally, the rectangle cue as figure 1 shown has two regions which are black and white. So N is 2 and the weight is:

$$\omega_2 / \omega_1 = -Area(r_1) / Area(r_2) \tag{2}$$

The sum would be deduced from equation (1) and shown by equation (3).

$$feature = \omega_1 \, \mathrm{Re}cSum(r_1) - \omega_2 \, \mathrm{Re}cSum(r_2) \tag{3}$$

Because of the acclivitous edge of triangle cues, we could not calculate the sum of them easily. In this paper, we proposed the limited approach to gain sum of triangle cues. This method was shown by figure 6.

In figure 6, the $\tilde{S}_i^j$ is the sum of inner pixels of the j*th* triangle in the i*th* level and the $\bar{S}_i^j$ is the sum of inner pixels of the j*th* rectangle in the i*th* level ($i, j \in N$). $\tilde{S}_0$ is the sum of inner pixels of white region of this triangle and $\tilde{S}_0^{'}$ is the sum of inner pixels of black region of this triangle. $\bar{S}_0$ is the sum of inner pixels of rectangle. Because of the same proportion of white and black regions, the ratio of ω_1 and ω_2 would be -1 as $\omega_2 / \omega_1 = -1$ shown. We can use $\omega_1 = -1$ and $\omega_2 = 1$ to calculate the sum of triangle cue as equation (3) shown and the result of it would be S as equation (4) shown.

$$S \approx \bar{S}_0 - 2 \cdot \sum_{i=1}^{N} \sum_{j=1}^{2^{i-1}} \bar{S}_i^j \tag{4}$$

Proof:

$$S \approx \tilde{S}_0^{'} - \tilde{S}_0 = \left(\bar{S}_0 - \tilde{S}_0 \right) - \tilde{S}_0 = \bar{S}_0 - 2 \cdot \tilde{S}_0 \tag{5}$$

and $\because \tilde{S}_0 = \bar{S}_1 + \sum_{j=1}^{2} \tilde{S}_1^j$

$$\sum_{j=1}^{2} \tilde{S}_1^j = \sum_{j=1}^{2} \bar{S}_2^j + \sum_{j=1}^{4} \tilde{S}_2^j$$

$$\sum_{j=1}^{4} \tilde{S}_2^j = \sum_{j=1}^{4} \bar{S}_3^j + \sum_{j=1}^{8} \tilde{S}_3^j$$

……

$$\tilde{S}_0 = \sum_{j=1}^{1} \bar{S}_1^j + \sum_{j=1}^{2} \bar{S}_2^j + \sum_{j=1}^{4} \bar{S}_3^j + \cdots \approx \sum_{i=1}^{N} \sum_{j=1}^{2^{i-1}} \bar{S}_i^j \tag{6}$$

So the S is

$$S = \bar{S}_0 - 2 \tilde{S}_0 \approx \bar{S}_0 - 2 \cdot \sum_{i=1}^{N} \sum_{j=1}^{2^{i-1}} \bar{S}_i^j \tag{7}$$

The calculation of triangle cues' sum was shifted by equation (7) into the calculation the sum of inner pixels in several rectangles which were gotten by integral image. So the relation between the error S_{Error} of the triangle cue and the level ***N*** of recursion calculated by equation (7) is:

$$S_{Error} = \sum_{j=1}^{2^N} \tilde{S}_N^j \tag{8}$$

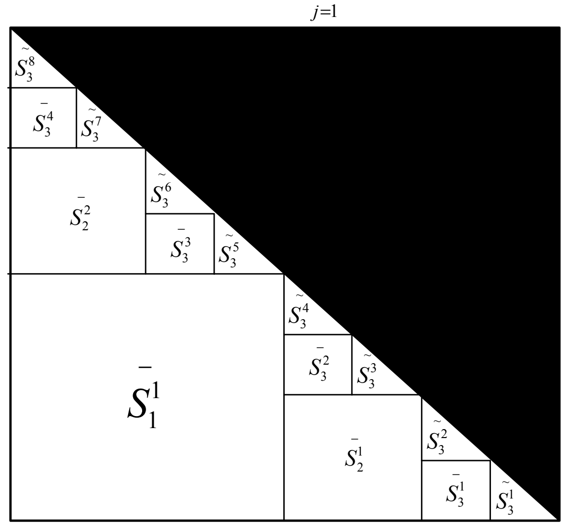


Figure 6. The sketch map of triangle cue calculation

The load of calculation is related to the number of rectangles and the level ***N*** of recursion. The load of calculating rectangle regions with integral image is independent of the area but only the number of the rectangle regions. So we can calculate the load $T_{consume}$ is:

$$T_{consume} = T + \sum_{i=1}^{N} \sum_{j=1}^{2^{i-1}} T = T \cdot \left(1 + 2^0 + 2^1 + \cdots + 2^{N-1} \right) = 2^N \cdot T \tag{9}$$

We can deduce it from equation (8) and (9) that the error and the calculation time of $T_{consume}$ are both related to the level N of recursion. The more of N, the smaller of the error and the more of time spend. So we should take balance of the error and the time.

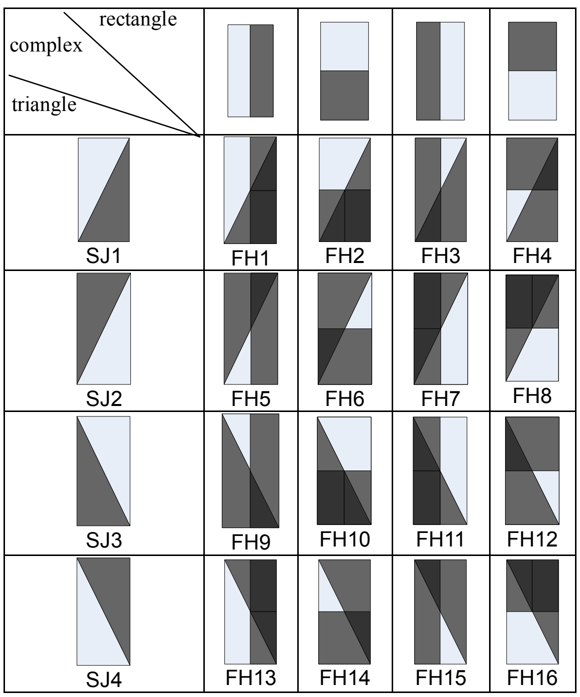
III. Complex Cues

The triangle cues could describe the edge but not whole of human figure because of the different clothes, color and illumination of human images. So the region of human image had no rule to stand by and the complicated background of the image could depress the effect of triangle cues. In order to describe the human figure totally, we proposed the complex cues which were combined by triangle and rectangle cues to pursue better effect in complex way.

There was a question about the complex cues that how to combine this two types of cue. The different area, size, number of those two cues would make different type of the combined cues. The modes would be boundless. So we should restrict those parameters of cues. In this paper, we ensured that those two types of cues had the same area, size and number. So we got 16 complex cues as table 1 shown. The sum of complex cues S_c would be calculated by sums of triangle and rectangle cues. We used the addition of those two sums which is shown in equation (10). The sum of triangle and rectangle cues are S and S' respectively.

$$S_c = S + S' \tag{10}$$

TABLE 1. COMPLEX CUES



IV. EXPERIMENT

A. Sample database

We set up a database to test the pedestrian detection algorithm. This database included over 200 videos recorded by still, moving and multiple cameras in manifold weather, background, shadow, illumination, indoor and outdoor conditions. Each video had 1000-20000 frames variably with 15 fps or 25 fps. The size of each frame was 320×240. We chose 15 videos to test the proposed algorithm, 10 for training data and 5 for test data.

The positive pedestrian samples were segmented from the training data by hand and were unified to the same size 24×32. All the positive samples were 4600, 4000 for training and 600 for test. The negative samples were the images without pedestrian and segmented automatically by program from the background of video. Figure 7and 8 showed some positive and negative samples respectively.

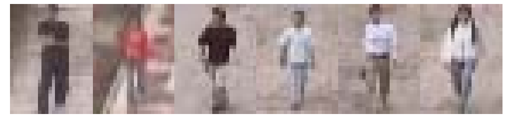


Figure 7. Some positive samples

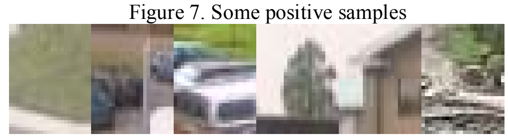


Figure 8. Some negative samples

B. Weak classifier test

In order to test the performance of those cues included 4 rectangle, 4 asymmetry, 4 triangle and 16 complex cues, we trained them by weak classifier shown by Paul Viola[1]. In this test, there were 28 types of cues which were YS, FDC, SJ, FH denoted 4 rectangle, 4 asymmetry, 4 triangle and 16 complex cues respectively. Every type has 31680 different size cues and all the cues were 887040. Trained by weak classifier, the cues were chosen with different number shown in and table 2.

TABLE 2. THE NUMBER OF CUES CHOSEN BY WEAK CLASSIFIER

Cues / Max error	YS1	YS2	YS3	YS4	FDC1	FDC2
	1	**2**	**3**	**4**	**5**	**6**
0.2	389	457	1812	66	6085	5137
0.1	0	0	42	0	682	559
0.05	0	0	5	0	38	62
0.01	0	0	0	0	0	0
Cues / Max error	**FDC3**	**FDC4**	**SJ1**	**SJ2**	**SJ3**	**SJ4**
	7	**8**	**9**	**10**	**11**	**12**
0.2	6071	5124	9724	9724	9724	9724
0.1	678	554	2699	2699	2699	2699
0.05	34	59	1251	1251	1251	1251
0.01	0	0	18	18	18	18
Cues / Max error	**FH1**	**FH2**	**FH3**	**FH4**	**FH5**	**FH6**
	13	**14**	**15**	**16**	**17**	**18**
0.2	11015	9552	11015	9552	11015	13358
0.1	2231	2837	2231	2837	2231	5015
0.05	999	1007	999	1007	999	2087
0.01	8	6	8	6	8	38
Cues / Max error	**FH7**	**FH8**	**FH9**	**FH10**	**FH11**	**FH12**
	19	**20**	**21**	**22**	**23**	**24**
0.2	11015	13358	9738	13358	9738	13358
0.1	2231	5015	2281	5015	2281	5015
0.05	999	2087	1019	2087	1019	2087
0.01	8	38	8	38	8	38
Cues / Max error	**FH13**	**FH14**	**FH15**	**FH16**		
	25	**26**	**27**	**28**		
0.2	9738	9552	9738	9552		
0.1	2281	2837	2281	2837		
0.05	1019	1007	1019	1007		
0.01	8	6	8	6		

Analyzed the result, we got this conclusions about rectangle, asymmetry, triangle and complex cues.

1) Under every level of the maximal error, the number of weak classifier trained by triangle cues was far more than

that by other cues. It indicated that the triangle cues could describe the human efficiently and robustly.

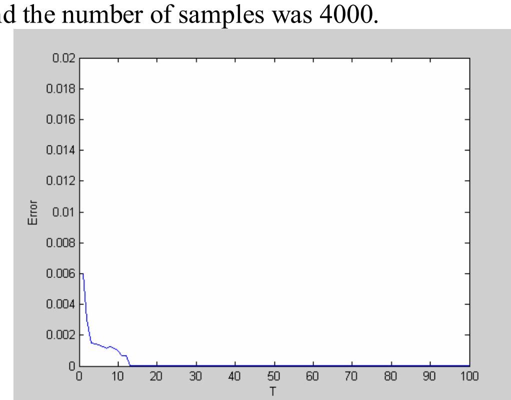
2) When the maximal error was **0.1**, there was no weak classifier trained by rectangle cues and few weak classifiers trained by asymmetry cues could be up to the mustard. Nevertheless, a great deal of weak classifier trained by triangle cues was chosen. When the maximal error was reduced to **0.01**, there was no weak classifier trained by rectangle and asymmetry cues could be used. But there were over **10** of which trained by triangle cues could be used. It indicated that the triangle cues could improve the performance of pedestrian detector.

3) At the same maximal error, those **4** types of triangle cues have the same number of weak classifier trained by each cue. It was proved that they were same effective and reliable.

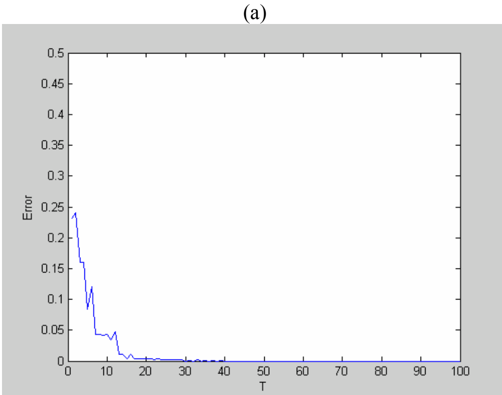
4) When the maximal error was different, the number of weak classifier trained by those 16 types complex cues were different. In the test, **FH6, FH8, FH10, FH12** were the best cues. It indicated that the complex cues were complicated because of the variety of clothes and illumination of human image. The performance of those complex cues was affected by the sample database.

C. *Strong classifier test*

We trained the strong classifier shown by Paul Viola[1] with those types of cues. The iterative times of training were 100 and the number of samples was 4000.



(a)



(b)

Figure 9. (a) The error curve of strong classifier trained by complex cues (b) The error curve of strong classifier trained by rectangle cues

The figure 9 showed that the error of the strong classifier trained by complex cues was 0.006 at the first training time but which of the strong classifier trained by rectangle cues was 0.2. After 10 times, the error caused by complex cues constringed to 0 but which caused by rectangle would spent 40 times to constringe to 0. It was said that the complex cues would improve the performance of strong classifier.

TABLE 3. NUMBER OF WEAK CLASSIFIER CHOSE EACH TYPE OF CUES

Cues	YS	FDC	FH	SJ
Number of every cues	4	4	16	4
Number of classifier	9	6	44	27
Ratio of chosen	2.25	1.5	2.75	6.75

The table 3 showed the number of weak classifier which chose each type cues and the ratio of weak classifiers with each cue which indicated that how many weak classifiers chose this cue. The ratio was more, the human description of the cue was better. In table 3, the triangle cues had the most capable to describe the pedestrian.

V. CONCLUSION

Started with studying the exterior cues of pedestrian, a new series of cues called triangle and complex cues were proposed to describe pedestrian. Using these cues, the adaboost pedestrian detector was built up with optimized structure and improved detecting efficiency. Under the same error ratio condition, experiments demonstrated that the adaboost detector with new cues used fewer cues than the one with symmetry cues.

REFERENCES

[1] Paul Viola, Michael Jones. Robust Real-time Object Detection[C]. In: Second International Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing, and Sampling, Canada. 2001.

[2] Dan Witzner Hansen, John Paulin Hansen, Mads Nielsen, Anders Sewerin, Johansen Mikkel B. Stegmann. Eye Typing using Markov and Active Appearance Models[C]. In: Proceedings of the Sixth IEEE Workshop on Applications on Computer Vision, 2002. 132~136.

[3] Saad Ali, Mubarak Shah. A supervised learning framework for generic object detection in images[C]. In: Proceedings of the Tenth IEEE International Conference on Computer Vision 2005. Vol.2, 1347~1354.

[4] Paclik P, Novovicova J, Duin R.P.W. A Trainable Similarity Measure for Image Classification[C]. In: Proceedings of the 18th International Conference on Pattern Recognition, 2006. Vol.3, 391~394.

[5] C. Papageorgiou, M. Oren and T. Poggio. A General Framework for Object Detection[C]. In: Proceedings of the sixth IEEE International Conference of Computer Vision, 1998. 555~562.

[6] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio. Pedestrian detection using wavelet templates[C]. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 1997. 193~199.

[7] Rainer Lienhart, Jochen Maydt. An Extended Set Of Harr-like Cues For Rapid Object Detection[C]. In: Proceedings of IEEE International Conference on Image Processing, New York, 2002. Vol.1, 900~ 903.

[8] Goncalo Monteiro, Paulo Peixoto, Urbano Nunes. Vision-based Pedestrian Detection using Haar-Like cues[C]. In: Robotica 2006 - Scientific meeting of the 6th Robotics Portuguese Festival, 2006.

[9] Zhu yiqiang. A pedestrian detecting system based on Adaboost algorithm:[Master dissertation] [D]. Xi'an: Northwestern Polytechnical University, 2006.