ITM Web of Conferences **9**, 03002 (2017) DOI: 10.1051/itmconf/20170903002
AMCSE 2016

Static and Dynamic Pedestrian Detection Algorithm for Visual Based Driver Assistive System

Idoko John Bush[1,*], *and Kamil Dimililer*[2]

[1]Computer Engineering Department, Near East University, Nicosia, Cyprus.
[2]Electrical & Electronic Engineering Department, Near East University, Nicosia, Cyprus.

Abstract. This paper presents a new pedestrian detection algorithm used in Advanced Driver-Assistance System with only one camera aiming to improving traffic safety. The new pedestrian detection algorithm differs from traditional pedestrian detection algorithm, which only focuses on pedestrian detection rate or pedestrian detection accuracy. Conversely, the proposed algorithm focuses on both the accuracy and the rate. Some new features are proposed to improve pedestrian detection rate of the system. Also color difference was used to decrease the false detecting rate. The experimental results show that the pedestrian detection rate can be around 90% and the false detecting rate is 3%.

Keywords. Pedestrian detection, detection rate, accuracy and false detection.

1. Introduction

Increasing concern for pedestrian safety in the last years has resulted in the flourishing of pedestrian detection algorithms. These are essential in Advanced Driving Assistance Systems for preventing accidents involving pedestrians. Car companies are considering incorporating such systems into their models. A number of research work have been done in car automatic protection systems, which is referred to as advanced driver assistance system (ADAS), to reduce the probability of the accidents.

Even though the problem was analyzed and tackled by many researchers it remains largely unsolved due to several difficulties:

i. The appearance of pedestrians in the image is always with high variability as the pose, the clothing and the articulations of the body parts can be various.

ii. Accidents always occur in urban environments where the background is complicated.

iii. When pedestrians are far from the moving vehicle, they may look similar to many background objects such as trees, telegraph pole and so on.

iv. As the speed of the vehicle is high, the pedestrian detection and the forewarning should be real time.

Several methods have been developed to improve the pedestrian detection rate and reduce the processing time. F. Garcia et al. [1] used a laser scanner and a far infrared camera to construct an ADAS. Alberto Broggi et al. [2] used four laser scanners and a camera to detect pedestrians. Junfeng Ge et al. [3] proposed a vision system for pedestrian detection and tracking during nighttime at the cost of a near-infrared camera. However, all the algorithms proposed are expensive from manufacturers' and economical angle. So this paper proposed a new algorithm to be used for ADASs with high pedestrian detection rate and less processing time. Also the new algorithm only needs one cheap camera.

2. Review of existing works

For the purpose of this paper, detail of few related works will be presented. These are papers which are strongly correlated with the approach implored in this paper and their description is needed for comparison. For a comprehensive overview of pedestrian detection algorithms the reader should consult the technical literature surveys [4], [5], [6], [7]. Despite the fact that there exist a multitude of approaches there is a tendency towards a general system architecture that is employed by most of them. This architecture is used to present and to emphasize different parts of the existing methods and the approach suggested in this paper. This paper mainly follows the description in [7] and state that the general pedestrian detection system has the following modules: pre-processing, feature extraction, region of interest selection (or foreground segmentation), object classification, post-processing (verification and refinement) and tracking.

To start off, traditional approach is described based on the method developed by Dalal [8] and extended by several other researchers [9], [10]. It is based on a fixed-size sliding window detection algorithm. To enable detection of pedestrians of different heights, the algorithm needs to resize the image and to recalculate the features for each scale. This is necessary because of two reasons: the detection window is of fixed size, and the features are not scale invariant. To obtain good results resizing must be done 4–8 times per octave and typically on 4–5 octaves. This leads to 16–40 feature recalculation steps. This is considered the weak point in similar approaches and the aim of this paper is to circumvent this situation.

* Corresponding author: john.bush@neu.edu.tr

Using two innovative ideas a recent publication by Benenson et al. [11] claim to achieve pedestrian detection at more than 100 frames per second. One of the ideas is to resize the features not images. The other is to use depth information for a successful region of interest selection. In this paper, a new pedestrian detection algorithm is proposed based on Adaboost, which is not only simple, but also with high pedestrian detection rate and high real time adjustment. Moreover, it is very cheap as only one camera is used making the system economical.

3. Proposed algorithm

The algorithm proposed in this paper contains two parts; the first one is to use feature based on Adaboost as a base algorithm, and the second one is to use color information to refining detection results.

3.1. Adaboost based algorithm

Paul Viola et al. have proposed a wavelet-based Adaboost method to integrate appearance information with motion information to detect pedestrian with great speed. The detector of this pedestrian detection system is based on the rectangle filters presented by Viola and Jones [12]. Even the wavelet-based Adaboost system contains both the static features and the motion features, the features are somewhat limited without considering the difference between the legs or arms of walking pedestrians and the background. The success rate is only 60%. So in this paper new feature shown in Fig.1 is added to express the difference between the pedestrian and background in one frame, to improve the detection rate to meet the demand of ADAS.

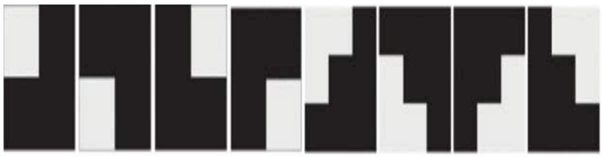

Fig. 1. The new features stand for the difference between the legs or arms of a walking pedestrian and the background.

The sum of pixels of the white rectangles, which is multiplied by 2, is subtracted from the sum of pixels of the dark rectangles.
The information about the difference between the pedestrian and the background can be extracted as follows:

$$S2 = 3 \times S_{white} - S_{black} \quad (1)$$

$$S3 = 9 \times S_{white} - S_{black} \quad (2)$$

Where S2 is the value of the 2×2 rectangle feature as the left part in Fig.1 and S3 is the value of 3×3 rectangle feature as the right part in Fig.1. S_{white} is the sum of white pix in the rectangle and S_{black} is the sum of black pix in the rectangle.

Adaboost was used to select a set of features to build the best classifier as in [12]. The layer of the Adaboost is 15 as Markus Enzweiler has pointed out that the performance of the system runs in saturation when the number of layers of Adaboost is more than 15 [13]. For each layer the detection rate is 99.5% and the false positive is 50%.

3.2. Refining the results of pedestrian detection maintaining the integrity of the specifications

Markus Enzweiler et al. [13] have pointed out all the pedestrian detection systems make rather similar mistakes that typical false detections between pedestrian and the things which are dominated by strong vertical structure occur in local regions. For example, there are always false positive between the pedestrian and the trees. Fig.2 (a) shows some false positives and the statistical result of different H values of the trees and Fig.2 (b) shows some true positives and the statistical result of different H values of the pedestrians. Several pedestrian detection algorithms contain one step to refining the results of pedestrian detection [14] [15] [16]. Most refinement methods use the multi-frame information to reduce the false positive, which means the target must be tracked before refining. However, these methods, which are at cost of time, can't meet the real time request of the system. So, a new method was proposed using the color information to refine the detecting results in order to reduce the false positives between the trees and the pedestrians.

Fig. 2 (a). Some false positives and the statistical result of different H values of the trees

Fig. 2 (b). Some true positives and the statistical result of different H values of the pedestrians

 DOI: 10.1051/itmconf/20170903002

The x axis stands for the h value from 0° to 360°, while the y axis represents for the total number for the corresponding H value. The left two figures are typical results of all the experiments.
Considering the fact that the color of the trees and the pedestrian are usually different and the H value of the objects doesn't change with the light, H value of the trees and the pedestrian was used to distinguish them.
The first step of the refining process is to calculate the H value for each pixel in the rectangle which is detected by classifier.

$$H = \begin{cases} 0, \text{ if } max = min \\ 60^0 \times \frac{G-B}{max-min} + 0^0, \text{ if } max = R, G > B \\ 60^0 \times \frac{G-B}{max-min} + 360^0, \text{ if } max = R, G < B \\ 60^0 \times \frac{B-R}{max-min} + 120^0, \text{ if } max = G \\ 60^0 \times \frac{R-G}{max-min} + 240^0, \text{ if } max = B \end{cases} \quad (3)$$

Where R,G,B is the original color information of an image, max is the maximum of the R value, G value and B value while the min is the minimum of the R value, G value and B value.

The second step is to make a count on the amount of each H value. We have done experiments on 500 pictures which contain 280 pedestrian images and 220 non pedestrian images. Fig.2 (a) and (b) shows some results of the H value of the pedestrians and the trees.

Then the last step is to remove the false positives of the trees according to the percentage of the H value within the range from 90° to 200°:

$$P_H = n_{range}/N_{all} \quad (4)$$

Where P_H is the percentage of which the H value is in the range 900 to 2000, n_{range} is the total number of the pixels, and the N_{all} is the sum of all the number with different H value. If P_H is more than a threshold, there should be a false positive.

4. Experimental results

In this section, some experimental results were presented to evaluate the pedestrian detection algorithm. The tests contain two parts, the first one is to use images captured in our university campus (Near East University, Cyprus) to evaluate detection rate and false positive rate. The second one is to test the robustness of the algorithm in real environment in the campus with a camera on a vehicle.

4.1. Evaluation of the new features

In order to test the usefulness of the new features, we give a comprehensive comparison between the different pedestrian detection system using different features. In the implementation, the number of positive examples is 280 while the number of negative examples is 220. All the positive examples are normalized to 32×80. Adaboost layer 15 was selected, the detection rate 99.5% and the false positive 50%. Fig.3 shows the comparing results. As expected, the detection rate of the pedestrian system with the new features is much higher than that without the new features at the same false positive.

4.2. Evaluation of; refining the detecting Results module

Even though the new features increase the detection rate when the false positive is small, there are still the false positives between the trees and the pedestrians. So the aim of this test is to evaluate detecting result refining module. The test lasted for 22minutes in our university campus where there are trees at both sides of the road. Then statistics was carried out on the set of 500 images. See Fig.4 for the result. We can conclude that, refining the detecting results module reduce the false positives at the same detection rate.

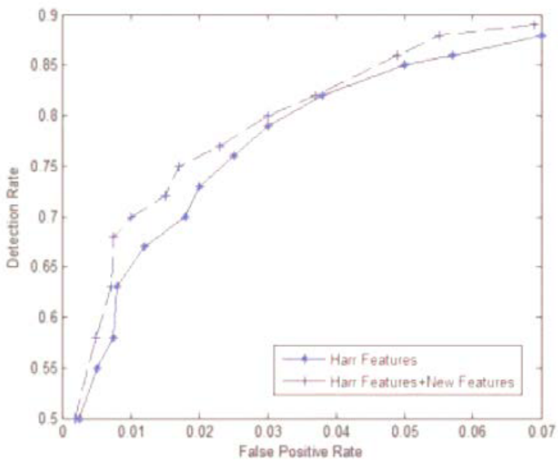

Fig. 3. Performance comparison of the pedestrian detection system using different features.

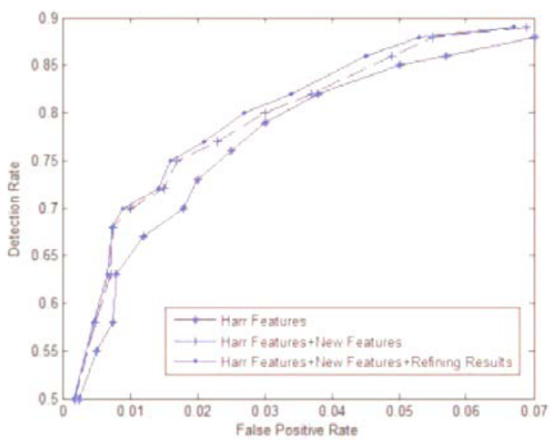

Fig. 4. Performance comparison between the system with refining detecting results module and that without refining detecting results module.

 DOI: 10.1051/itmconf/20170903002

4.3. Evaluation of the robustness

Also the pedestrian detection algorithm was tested on a real car with different vehicle speed in Near East University campus. The vehicle speed of the first experiment is around 30km/h, and the second one is 35km/h. The system was built on Intel Pentium 2 processor with 2.0GHz and 4GByte Memory with Window10 system and the program was written in C++.

Table1 shows the system performance under different vehicle speeds. As expected, the detection rate of the system is around 90% in both cases, and the false alarm is under 2.

Table 1. System Performance

Vehicle Speeds	System Performance		
	Running Speeds(ms)	Detection Rate	False Alarm
30km/h	42	94.3%	1
35km/h	44	89.8%	2

5. Conclusion

This paper presents a new pedestrian detection algorithm which not only improves pedestrian detection rate but also decreases false positive rate. 500 images have been tested to evaluate the performance of the algorithm. Also we have tested the algorithm in real environment to evaluate whether the algorithm is available in real ADASs. The results indicate that the pedestrian detection rate can be improved to 90% and the false positive rate can be decreased to 3%. The real experiments in Near East University campus proved the robustness of algorithm with different speeds of cars, which showed the algorithm can meet the requirement of ADASs.

Future work in the field and environment (Near East University) will include integration of camera calibration to pedestrian detection in order to get the real distance between pedestrians and vehicles so that we can give an advanced alarm to avoid some traffic accidents. Also, comparison between the future work and other related works will be done to validate the authenticity of the pedestrian detector.

References

[1] F. Garcia, D. Olmeda, J. M. Armingol and A. de la Escalera, “Hybrid fusion scheme for pedestrian detection based on laser scanner and far infrared camera,” Intelligent Vehicles Symposium, pp.1134-1139, Jun. 2010.

[2] Alberto Broggi, PietroCerri, Stefano Ghidoni, Paolo Grisleri and Ho Gi Jung, “A new approach to urban pedestrian detection for automatic braking,” IEEE Trans. Intell. Transp. Syst., vol. 10, no. 4, pp.594-605, Dec. 2009.

[3] D. Gavrila, J. Giebel and S. Munder, “Vision-Based Pedestrian Detection: The PROTECTOR System,” Proc. IEEE Intelligent Vehicles Symp.,pp: 13-18, 2004.

[4] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: A benchmark,” in CVPR, 2009, pp. 304–311.

[5] P. Doll´ar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,” IEEE Trans. Pattern Anal. Mach. Intell, vol. 34, no. 4, pp. 743–761, 2012.

[6] M. Enzweiler and D. M. Gavrila, “Monocular pedestrian detection: Survey and experiments,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.

[7] D. Ger´onimo, A. M. L´opez, A. D. Sappa, and T. Graf, “Survey of pedestrian detection for advanced driver assistance systems,” IEEE Trans. Pattern Anal. Mach. Intell, vol. 32, no. 7, pp. 1239–1258, 2010.

[8] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in CVPR, 2005, pp. I: 886–893.

[9] Q. A. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, “Fast human detection using a cascade of histograms of oriented gradients,” in CVPR, 2006, pp. II: 1491–1498.

[10] T. Watanabe, S. Ito, and K. Yokoi, “Co-occurrence histograms of oriented gradients for pedestrian detection,” in PSIVT, 2009, pp. 37–47.

[11] R. Benenson, M. Mathias, R. Timofte, and L. J. V. Gool, “Pedestrian detection at 100 frames per second,” in CVPR. IEEE, 2012, pp. 2903– 2910.

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

[13] Markus Enzweiler, “Monocular Pedestrian Detection: Survey and Experiments,” IEEE Transactions on pattern analysis and machine intelligence, vol. 31, no. 12, pp: 2179-2195, Dec. 2009.

[14] D.M. Gavrila and S. Munder, “Multi-Cue Pedestrian Detection and Tracking from a Moving Vehicle,” Int’l J. Computer Vision, vol. 73, no. 1, pp: 41-59, 2007.

[15] Roland Schweiger, Heiko Neumann and Werner Ritter, “Multiple-cue data fusion with particle filters for vehicle detection in night view automotive applications,” IEEE Intelligent Vehicle Symposium, pp:753758, 2005.

[16] D. Gavrila, J. Giebel and S. Munder, “Vision-Based Pedestrian Detection: The PROTECTOR System,” Proc. IEEE Intelligent Vehicles Symp.,pp: 13-18, 2004.