Computer Vision: Pedestrian Detection image processing algorithm for Traffic light system

[1] Shraddha Kallappa Walikar, [2] Dr. Aswatha Kumar M
[1] PG Student, [2] Principal, Sapthagiri College of Engineering Bangalore
[1] shraddha.ekgd@gmail.com

Abstract*— With the technology today, detection of pedestrian was introduced to decrease the accidents in the roads. It was implemented on the traffic lights and cars. In this paper, integration of face detection and pedestrian takes place on the device called Raspberry Pi. OpenCV was used for the detections and programmed in Python. The webcam was used to capture images in real-time. The image was processed at the microprocessor. When pedestrians were detected, the microprocessor played the recorded voice and outputs it to the connecting speaker to notify those pedestrians or the pedestrian to wait before crossing until such time. The microprocessor sends a signal to the Gizduino to process the time of the red lights on state. The microcontroller was programmed with the traffic signals. When it receives a signal from the microprocessor, it added an extra seconds to the red lights on state. Afterwards when it returned, it changed back to its original on state. The success rate was 80% this shows the two detection was integrated in the system successfully worked.

Index Terms— ***Open CV; Raspberry Pi; Python; Gizduino; Pi-camera***

I. INTRODUCTION

The Pedestrian push button traffic control or the Ped call button [5] that exist in most crossing areas are used to change traffic light transitions by the pedestrians in order to have their turn to cross. Another device that is found in some areas is the Closed-circuit television cameras. They are also used to monitor areas or things that could to help security personnel's investigate acts harmful to human and infrastructure. Hence cameras are now used for pedestrian detection feature to detect the presence of people. This type is used in the streets to know if the pedestrian wants to cross the street, which is similar with a Ped call system (Tom de Castella, 2013). This makes a cameras on pedestrian lane have additional functionality to help the pedestrians consider their turn to cross.

The application used is real-time which does not record the amount of frames of images to analyze. For cross-line roadways, multi-camera surveillance is implemented to utilize a mathematical approach for switching emergency events. On evening rush hours, surveillance processing for monitoring vehicle movements is essential (S. Chandana, 2010). This study of Haar Cascaded Algorithm for object and digital image recognition (Ms. Jaya M et al., 2013) leads to Haar features have been studied intensely for the detection of objects, in particular for face detection (P. Viola and Jones, 2004). One another major feature used for object detection is provided by HOG as evaluated in (P. Dollar et al., 2009). Pedestrians, faces and bicycles are successfully detected when represented by HOG (N. Dalal et al., 2005; A. Adam et al., 2006) algorithm also used for pedestrian [4].

The absence of initiating Ped call button while waiting by uninformed people was observed. Existing face detection and pedestrian detection algorithms [4], [11] were not previously used in the pedestrian detections. The proposed system, should integrate an image processing system into a camera system thus eliminating the push button system. Cameras of today are more advanced than before. Pedestrian crossings are one of the places where cameras are installed. In some pedestrian crossings, push button devices are used instead of cameras. The pedestrian crossings are used by where the traffic light control system time changes once the pedestrian pushes the button [5].

The main objective of this study was to design a pedestrian detection [11] system using image processing by integrating it into an existing camera commonly installed on street poles. Specifically, amid;

- To develop and test a face detection algorithm for human facial expressions.
- To develop and test a pedestrian detection algorithm for pedestrian proper location in the opposite side of the pedestrian crossing and
- To integrate and test the algorithms for automatic traffic light control system

Face detection [8], [9] for pedestrians was helpful to the community providing a method of traffic control, without the presence of traffic enforcers for either gestural manipulation or button manipulation. This was to eliminate the Ped call button traffic control system [7]. It would be feasible when the system was integrated with cameras and with this system.

II. Hardware Used

A. Raspberry pi

The **Raspberry Pi** is a series of credit card sized single-board computers developed in Wales, United Kingdom by the Raspberry Pi with the intention of promoting the teaching of basic computer science in schools and developing countries. The original Raspberry Pi and Raspberry Pi 2 are manufactured in several board configurations through manufacturing agreement. The hardware is the same across all manufacturers.

The Foundation provides Debian and Arch Linux ARM distributions for the some download, and promotes Python as the main programming language, with support for BBC BASIC, C,C++, Java, Perl, Ruby, Squeak Smalltalk and more also available. All Raspberry Pi-2 includes the same Video Core IV GPU, and either a single-core ARMv6-compatible CPU, compatible quad-core one; and 1 GB of RAM (in Pi 2), 512 MB (in Pi 1 models B and B+) or 256 MB (in models A and A+, and in the older model B). They have a Secure Digital (SDHC) slot (models A and B) or a Micro SDHC one for boot media and persistent storage.

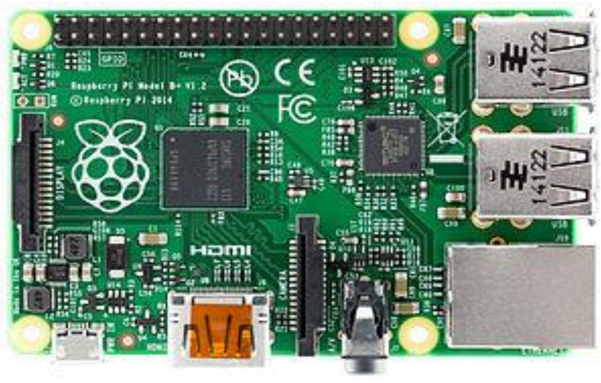

Fig.2. Raspberry Pi-2 Model [12]

In 2014, the Raspberry Pi-2 Foundation launched the Compute Module, for use as a part of embedded systems for the same compute power as the original Pi.

B. Raspberry Pi-camera

The Fig 3 shows Raspberry Pi-camera. There are three applications provided, raspistill, raspivid and raspistillyuv. raspistill and raspistillyuv are very similar and are intended for capturing images, raspivid is for capturing video. Raspberry Pi-camera of all the applications is command line driven, written to take advantage of the MMAL API which runs over Open MAX. The MMAL API provides an easier to use system than that presented by Open MAX. Note that MMAL is a Broadcom specific API used only on Video core 4 systems. The applications use up to three Open MAX (MMAL) components of the Pi-camera, preview and encoder [12].

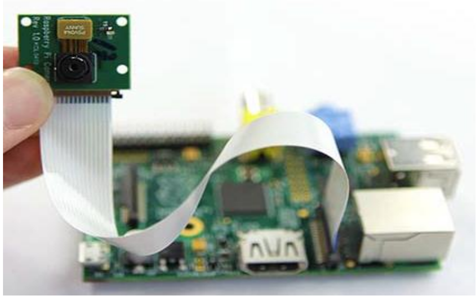

Fig.3 Raspberry Pi Camera [12]

All applications use the camera component, raspistill uses the Image Encode component, raspivid uses the Video Encode component and raspistillyuv does not use an encoder and sends its YUV output direct from camera component to file. The preview display is optional, but can be used full screen or some directed to a specific rectangular area on the display. The Raspberry pi ribbon cable has 15pins with 5mp rear camera; and has full HD with 1080p [12].

III. Methodology

The system was composed of Pi-camera, Raspberry Pi-2, Arduino for traffic signal for controlling traffic light system and recorded voice notifies using a speaker to alert the nearby people. The Pi-camera was connected to the raspberry pi-2 model, where the input of the images was captured. The raspberry pi-2 analyzed the captured image. The program in the Raspberry Pi would detect the face of the pedestrian. If detection of face was found, a recorded voice would play through the speaker and the traffic lights of red light [7] would be hold some time shows in Fig 4.

A. System Flow

Images captured were analyzed using the pedestrian detection followed by the pedestrian of the face detection, [3] if there was a pedestrian found in the images. If there was a face found, it played the recorded voice to alert vehicles and sent a signal to the Arduino, which was programmed with the traffic light system. The green light of traffic signal is turned as red light and red light were hold for few seconds. If none was found, it would come into original traffic signal form.

B. Software Coding

The Raspberry Pi uses the face and pedestrian detections were integrated using Python language and the traffic light system was in C language, tool used as Open CV.

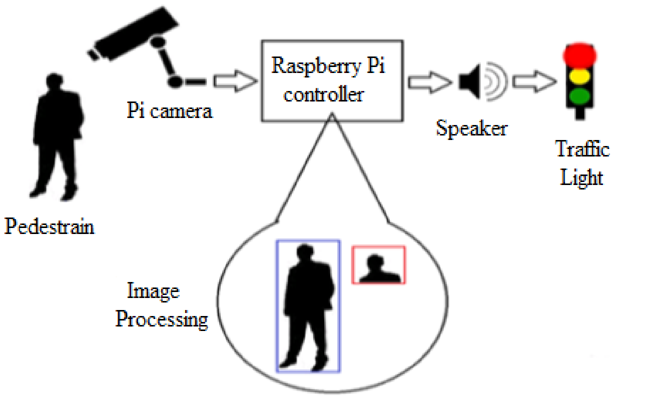

Fig.4 Block diagram of the System [7]

C. Set – Up

The Pi-camera was faced in the opposite side of the pedestrian lane with respect to the length of appropriate vision by the Pi-camera for the pedestrians. The component was inside a case which served as a protection from weather changes.

IV. Haar Features and HOG Algorithm

A. Haar Cascaded Classifiers

Haar features are based on Haar wavelets, which are functions that consist of a brief positive impulse, followed of a brief negative impulse. In image processing, a Haar feature is the difference between the sums of all pixels in two or more regions. Papa Georgiou et al. were the first to use Haar features for face detection. They used three types of Haar features of size 2×2 and 4×4 pixels, for a total of 1,734 different features in a 19×19 face image.

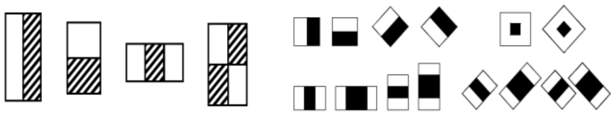

Fig.5 (a) Haar features introduced, (b) Extension to the basic set [6].

Viola and Jones proposed a basic set of four types of Haar features that are shown in Figure 5 a. The value of Haar feature is given by the sum of intensities of the pixels in the light region minus the sum of intensities in the dark region. Usingall possible sizes, they generate around 180,000 features for a 24×24 pixel image [6]. An extension to the basic set with rotated Haar features as shown in Fig 5 b.

B. HOG algorithm

The HOG features are extracted from all points on a dense grid. In this paper uses the grids of HOG features as the primitive features because they significantly outer perform existing feature sets for human detection. To improve the recognition performance and reduce the computation cost, it also applies PCA to the HOG features. In this paper, it calls them PCA-HOG features [1].

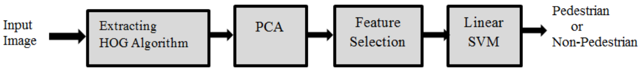

Fig.6. Overview of our pedestrian detection algorithm.

It is well known that feature selection is effective for pattern classification. It is expected that the recognition performance can be further improved by selecting a proper subset of the PCA-HOG features because some local regions are irrelevant to pedestrian detection. For example, textures in clothes are not relevant to person detection [2]. Then the selected PCA-HOG features are used as an input vector of linear SVM for person/non-person classification [10]. The overview of our pedestrian detection algorithm is shown in Fig 6.

A HOG feature vector represents local shape of an object, having edge information at plural cells. In flatter regions like a ground or a wall of a building, the histogram of the oriented gradients has flatter distribution [1]. On the other hand, in the border between an object and background, one of the elements in the histogram has a large value and it indicates the direction of the edge. Although the images are normalized to position and scale, the positions of important features will not be registered with same grid positions [10]. It is known that HOG features are robust to the local geometric and photometric transformations. If the translations or rotations of the object are much smaller than the local spatial bin size, their effect is small, a set of HOG feature vectors from all locations in an image grid and are used for classification.

V. Experimental Results

The face detection was firstly tested in this study. The pedestrian detection was also tested and it was found that it detected other objects other than the human body. The last was the integration of the both detection.

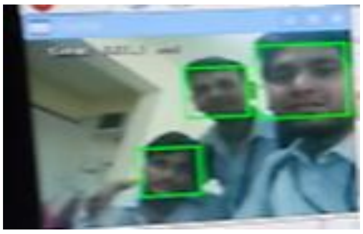

Fig 7 Real-time face detection

The real-time face detection and pedestrian detection was firstly tested in this study on the Raspberry Pi. The real-time Face and pedestrian detection was also tested. Fig.8 shows the result of real-time face detection and Pedestrian detection implemented using python language and Open CV tool.

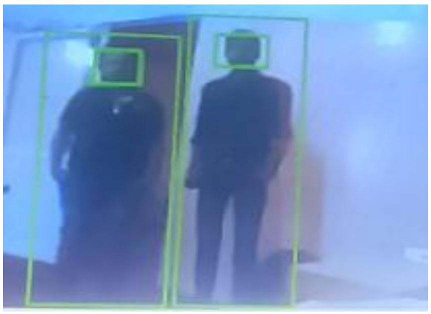

Fig 8 Real-time pedestrian detection

Table.1 Summary of Data for True Positive

	Face	Pedestrian	Face and Pedestrian
True Positive	2	5	5
Expected	5	5	5

The tables in this paper used the classification method where in the performances of the detections were tested. True positive were used to test for presence of the target.

$$\text{TPR} = \frac{\text{True Positive}}{\text{Expected}} \text{ x } 100$$

$$\text{TPR} = \frac{12}{15} = 0.8 = 80\%$$

The success rate of the experiment was 0.8. TPR or true positive rate was the rate of correctly detecting the Pedestrian and their face.

VI. Acknowledgment

It is great privilege to place on record my deep sensor gratitude to my guide Dr. Aswatha Kumar M and the management of Sapthagiri College of engineering, who patronized throughout career for the facility provided to carry out this work successfully and thanks to my family.

Conclusion

The objectives of this study were successfully achieved. The integration of the real-time face and pedestrian detection was successful implemented and implemented the system to the traffic light control system to change the signal of the traffic light. The face and pedestrian detection were found using python language and open CV tool. Based on the computed value using the True Positive Rate, it concluded that the system can detect a pedestrian’s body and face and the success rate of 80%.

References

[1] N. Dalal and B. Triggs, Histograms of oriented oradients for human detection. In Computer Vision and Pattern Recognition, pp.886–893, 2005.

[2] A. Adam, E. Rivlin and I. Shimshoni, Robustfragment-based tracking using integral histogram. In Computer Vision and Pattern Recognition – CVPR, 2006.

[3] P. Viola and M. Jones, Robust real-time face detection.In International Journal of Computer Vision, 2004.

[4] P. Dollar, C. Wojek, B. Schiele and P. Perona, Pedestrian detection: a benchmark. In CVPR, 2009.

[5] Tom de Castella, Does pressing the pedestrian crossing button actually do anything, BBC News Magazine September, 2013.

[6] Ms. Jaya M. Jadhav; Ms. Deipali V. Gore, Introducing celebrities in an images using HAAR cascaded algorithm, International Journal of Advanced Research in Computer Science and Software Engineering Vol. 3, pp 740-742, August 2013.

[7] Chandana S, Real Time Video Surveillance System Using Motion Detection, Dept of Electronics and Communication Engineering, DayanandaSagar College of Engineering Bangalore, 2010.

[8] Yohan Dupuis, Xavier Savatier, Jean-Yves Ertaud and PascualVasseur, A new approach for face detection with omnidirectional sensors, December 2011.

[9] Sheng Yang, Jiefa Wu and Lingling Zhang, A fast pedestrian detection method based on simplified hog descriptor, International Journal of Digital Content Technology and its Applications (JDCTA), Vol. 6, no. 4, March 2012.

[10] K. Mikolajczyk, C. Schmid and A. Zisserman, Human detection based on a probabilistic assembly of robust part detectors. In ECCV 2004.

[11] J. Gall, A. Yao, N. Razavi, L. V. Gool and V. Lempitsky, Hough forests for object detection, tracking and action recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 33, no. 11, pp. 2188–2202, 2011.

[12] Bridgwater, Adrian community strength blossoms for rasberry Pi 2, raspberry Pi camera and raspberry pi 2 forums raspberry pi 2 foundation. Retrieved Feb-2014.