

# Pedestrian Traffic Light Recognition

## For The Visually Impaired

Department of Electrical Engineering, Ben-Gurion University

Amir Shalev ; Ben Lauterbach

### Abstract

Our project main focus is the to give a better and much more efficient solution for a well known problem. We choose to tackle a problem, which concern the visually impaired community – Road crossing; And to be more precise: Pedestrian Traffic Light Recognition. In this work we employ the technology of computer vision for solving this issue. Our approach is mainly based on color segmentation. The system is able to detect and to track green and red traffic lights used by the pedestrians. We notice a high rate of correctly recognized traffic lights and very few false alarms.

**Keywords – Traffic light recognition; computer vision; color detection; classification**

### 1. INTRODUCTION AND PROBLEM DESCRIPTION

According to the statistics of World Health Organization, there are approximately 40 million people suffering from visual impairment all over the world. Usually, blind people use a first aid, such as the white cane. It allows them to detect obstacles in a close range, but cannot help them with detecting the condition of a traffic light. The visually impaired have difficulty seeing the traffic signs, however they can hear. Although there are some audio beepers employed to inform a visually impaired person what color the traffic light is by means of different frequency. Such equipment is not available at every road crossing and also makes unbearable noise, especially at populated areas. With the fast growing of smartphone users around the world and the advanced technological development of wearable electronic products such as: eyewear cameras (Google glass) – a computer vision solution is inevitable. Our goal was to design a real time recognition system based on video stream input. The system will recognize the traffic light and will inform the user of its state (red/green light). The system should work in both day and night time and should have a high reliability. Such a system would increase the mobility of the blind people, freeing them from the special facilities

provided by different cities or districts and will increase their sense of security regarding road crossing.

In recent years, several different approaches have been taken to resolve the problem [1-4]: color detector, a shaped based detector, spot light detection etc. Knowing each picture (video frame) is a mathematical combination of three basic colors also known as the RGB color model (where red, green, and blue light are added together in various ways to reproduce a broad array of colors). And the fact that we wish to find red/green colors, led us to look for a color model solution. However, we found that using the RGB model to detect strain colors is not an easy task in a real time complex scene, such as road crossing. Moreover, even when we do succeed detecting all the red and green colors in a frame, who can we be sure which one is a red / green light of a Pedestrian Traffic light? Many road-crossing junctions include several traffic lights, not necessarily in the same direction. Respectively, another problem is to determine which traffic light is relevant for the pedestrian in a certain point of time. In general, our approach to reach our goals was based on color segmentation by HSI color model. Similar to the HSV (fig.1) color model the HIS model is also a cylindrical-representation of points in an RGB color model, where H stands for Hue, S stands for Saturation and I for Intensity.

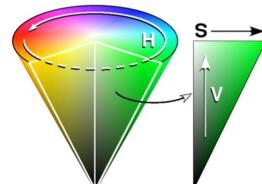


Fig.1: HSV color model

In order to successfully recognize the red and green light of the traffic light, after extracting all the

suspicious areas we perform a series of tests to distinguish between foals and true results.

## 2. OUR SOLUTION

The input of the system is a continues traffic video sequence, which first will be separated into individual image frames for further processing.

**Step 1:** The image frames composed of the RGB tri-color model are converted into the HIS (fig.2) color model ( $I$ )

$$I = \frac{1}{3}(R + G + B)$$

$$S = 1 - \frac{3}{(R + G + B)}[\min(R, G, B)] \quad (1)$$

$$H = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{1/2}} \right\}$$

[4]. First we tried to extract the red and green colors by the next defined values (as mentioned in: [4]): Red light is defined by the condition of hue value with  $0^\circ$  plus or minus  $15^\circ$  and intensity value of 0.8. Green light is defined by the condition of hue value with  $120^\circ$  plus or minus  $15^\circ$  and intensity value of 0.8 as well. Unfortunately we discover that those values are not so accurate. We decide to examine a large group of stills photos (frames) to get to best threshold for extracting only the red and green lights.

At this part we collected the results in EXCEL sheet, which made it easier for calculating the best thresholds. (See appendices for farther information).



Fig.2(a): HSI color space

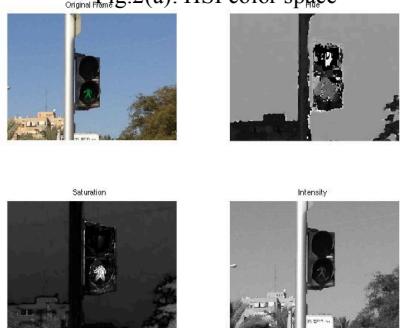


Fig.2(b): HSI color space traffic light zoom

**Step 2:** After we compute the most efficient threshold. We used this threshold to produce a binary image contain only the red and green areas. For better “clean” result we also used opening and closing functions.

(Fig.3)

**Step 3:** One can see that even after applying the most efficient threshold, plus minimize the false results with



Fig.3(a): Binary red areas



Fig.3(b): Binary green areas

opening and closing functions. We still cannot get a clear satisfying result. To over come a situation, where in every frame we can get multiply results about the location of the traffic light, we chose to process each frame through number of functions (filters). Each function shall give a grade for each suspicious pixels cluster. By the end of the procedure the one who got the highest score will be choosen as the traffic light coordinate.

First we shell count and locate all the suspicious candidates (using the ‘bwlabel’ function). Next we’ll calculate their center of mass coordinate. (From this point each center of mass (CM) coordinate will represent each candidate).

Traffic light pole detection – considering the fact that each pedestrian traffic light is mounted on a pole, we deiced to detect all the relevant poles in each frame. By doing so we gave a high score for those center of mass coordinate that found closest to the pole. We preform 3 major steps for poles detection (Fig.4). Edge detection using ‘Canny’, ‘Hough’ transform for straight-line detection and finally ‘ACE’ algorithm (applying Inner product for the angle of the line and it’s radius) to extract the orthogonal lines – meaning the poles (most

of the straight lines in a frame are horizontal).



Fig.4(a): Edge detection using 'Canny'



Fig.4(b): 'Hough' transform for straight-line detection



Fig.4(c): 'ACE' algorithm, extracting poles

As we already monitor, cross road scenes are very complex, and sometimes include several traffic lights. After we detect all the poles in a frame and score each center of mass coordinates respectively, we shall know deal with the issue of determining which traffic light is relevant for the pedestrian in a certain point of time. Due to the fact we are using just one camera and can't analyze a 3D scene, we had to come up with a simple but efficient solution. After examining significant numbers of cross roads we came to the conclusion that the designated traffic light (Center of mass) is always higher than the rest of the traffic lights. A simple function is scoring the CM coordinates from the highest to the lowest.

In order to improve and accelerate the process even more, we wrote a function that considers the last location the traffic light was found. Due to the fact that people (and especially visually impaired persons) tend to cross intersections in slow walking pace. The location of the traffic light (relative to the observer) from frame

to frame is also slowly changing. This function gives a higher score for suspicious CM coordinates which are relatively close to the old CM found in the previous frame.

By now we got consistently satisfying results, except in extreme cases such as green poles (Fig.5). Accordingly the final frame process refers to the traffic light as a source of light. We used 'imtophat' and 'imadjust' functions for spot light detection. [1]



Fig.5: False identification

Fig.6 summarizes the Traffic Light Recognition algorithm used in this paper.

Fig.6: Algorithm flowing chart



### 3. EXPERIMENTS AND RESULTS

Fig.6(a-d) showing the results we accomplish using the above algorithm. We exam our algorithm on steals photo and video, both in day and night time.



Fig.6(a): Green traffic light day time



Fig.6(b): Red traffic light day time



Fig.6(c): Green traffic light, night time



Fig.6(d): Red traffic light, night time.

One can see, we were able to get good and reliable results. However we straggle to get the same quality results just before night time, at twilight (Fig.7). Most false results were related to red light detection.



Fig.7(a): False result, wrong traffic light



Fig.7(b): False result, wrong red recognition

Another weakness is our program running time. We didn't manage to get a clean smooth video.

### 4. CONCLUSIONS AND FUTURE IDEAS

In this work we examined and executed a real time traffic light recognition system, based on computer vision technology. To achieve our goal we used a variety of techniques. We think that the most significant thing that we learned from this project is the fact, that to accomplish a specific task (red/green color recognition) a whole set of tools is needed. Good chance that we would never reach such results if we tried to solve the recognition dilemma only by color segmentation methods. As we shown in this paper, traffic light recognition is possible by means of computer vision. Without a doubt implement such thing can benefit the life of the visually impairment in the near future.

Future ideas – to make this algorithm effective and truly useful for every day use, a better running time is required, (Less time consuming functions). In addition a solution for twilight time is required. This project can be expanded for other road cross themes, such as Zebra Crossing detection. We did deal with this issue a little

bit. Like to pole detection we assumed that find the zebra crossing pattern will help us recognize more effectively the traffic light (maybe using vanishing point?). Unfortunately our function didn't work on video stream and only gave some satisfactory results on steals photos (Fig.8).



Fig.8: Zebra crossing detection

## 6. APPENDICES

- In the google drive share folder of the course we put some excel sheets to illustrate the work we did about finding the best green / res threshold. (Project No 9)

## 5. REFERENCES

1. R. de Charette and F. Nashashibi, "Real time visual traffic lights recognition based on Spot Light Detection and adaptive traffic lights templates," *2009 IEEE Intelligent Vehicles Symposium*, Xian: IEEE, 2009, pp. 358-363.
2. Jianwei Gong, Yanhua Jiang, Guangming Xiong, Chaohua Guan, Gang Tao and Huiyan Chen. The Recognition and Tracking of Traffic Lights Based on Color Segmentation and CAMSHIFT for Intelligent Vehicles. 2010 IEEE Intelligent Vehicles Symposium University of California, San Diego, CA, USA June 21-24, 2010.
3. Bruno A, Electronic Sensory Systems for the Visually Impaired, *IEEE Inst & Mea Mag*, 2003.
4. Shun-Hsien Taso, Pedestrian Traffic light recognition For The Visually Impaired, Biomedical Engineering: Applications, Basis and Communications, Vol 19, No. 5 (2007), 289-294.