DOI: 10.15849/IJASCA.230720.14

Int. J. Advance Soft Compu. Appl, Vol. 15, No. 2, July 2023
Print ISSN: 2710-1274, Online ISSN: 2074-8523
Copyright © Al-Zaytoonah University of Jordan (ZUJ)

Recognizing Traffic Lights of Interest for Vehicle Driver without Prior Information

Reem Shadid, Waseem Shadid, and Yara Khawaja

Department of Electrical Engineering, Applied Science Private University, Amman 11931, Jordan
MEU Research Unit, Middle East University, Amman, Jordan
e-mail: re_shadid@asu.edu.jo
Department of Software and Information Systems (SIS), The University of North Carolina at Charlotte, Charlotte, NC 28223, USA
Email: wshadid@uncc.edu
Department of Electrical Engineering, Applied Science Private University, Amman 11931, Jordan
MEU Research Unit, Middle East University, Amman, Jordan
e-mail: y_khawaja@asu.edu.jo

Abstract

This paper presents a novel automatic system to estimate the traffic light signal of interest for a vehicle driver in colored images of multiple lanes street from a camera mounted on car. The system analyzes the image to determine the candidate regions for traffic light signals. These regions are categorized based on their color and shape. Then spatial constraints are applied to generate an estimate of the traffic location. Existing solutions attempt to detect all traffic light signals in an image. Further solutions require prior information to specify the traffic light of interest with GPS information. But there is no work that estimate the traffic light signal of interest for the driver when multiple traffic light signals present in an image without prior information. There are two key contributions for this work: Building the first prototypical system that generates an estimate for the traffic light signal, and using a unique combination of the normalized RGB and the HSV color spaces. The proposed system was evaluated on LISA Traffic Light Dataset with maximum accuracy 97.6%. Moreover, the proposed system has been tested on private data sets from real-world environments. This work helps to improve the performance of advanced driver assistance systems.

Keywords: *Color thresholding; Image processing; Normalized RGB and HSV; Traffic light.*

Received 22 March 2023; Accepted 6 May 2023

Nomenclature

a1,a2	Multiplication factors
B_{bottom}	Range band bottom values
Bn	Normalize blue
B_{top}	Range band top values
D(.)	Distance between the traffic light rectangle center and the center of the image
Gn	Normalized green
H	Hue of the HSV color space in the range from 0 to 360
(Sj(i))	Height of the bounding rectangle
Ia	Binary image for amber color image
Ig	Binary image for green color image
Ipa	Processed amber image
Ipg	Processed green image
Ipr	Processed red image
Ir	Binary image for red color image
J	Color identifier
L	Label of traffic light
LocX	Location along x-axis
Nj	Number of regions
Rn	Normalized red
Sca	Candidate regions for amber color
Scg	Candidate regions for green color
Scr	Candidate regions for red color
Sj_{Center}	Top of traffic light candidate region on center side image
Sj_{Left}	Top of traffic light candidate region on left side image
Sj_{Right}	Top of traffic light candidate region on right side image
T1	Set the first third of the image width
T2	Set the second third of the image width
TL	Traffic light
TL_{top}	Traffic light signal that is closest to the image top
TL_{driver}	Traffic light signal of interest
V	Value of the HSV color space
W(Sj(i))	Width of the bounding rectangle
Y(Sj(i))	Y-location of the top side of bounding rectangle

1 Introduction

In recent years, developing an Advanced Driver Assistance System (ADAS) has gained a lot of attention due to its importance [1], [2]. This system is used to assist drivers and to build autonomous vehicle navigation system. ADAS is built with on-board sensors to provide various environmental data to the used processing algorithms. These data include: traffic signal, signs, and lane markings. The algorithms used in ADAS include a Traffic

Light (TL) detection technology to estimate TL state, i.e., red, amber, and green, in images acquired from a camera mounted on the vehicle. This technology is important for road safety and accident avoidance.

There are two main methods for TL recognition and detection: model-based and learning-based. In model based, the information of color and shape of TL are used to build a model that is capable to detect it. Whereas in learning-based method, recognition and detection processes were changed from rule-based to learning-based. In this paper we are focusing on model-based method.

TL detection is a challenging problem. Challenges include illumination variation, occlusions, and background noises, e.g., billboards and vehicle tail-lights. TLs have a well-defined structure in terms of color and shape information In conventional TL detection algorithms, a set of candidate regions for TL is generated according to color and shape information [3]–[6]. This set is then shorten using video tracking algorithms before identifying the TL state. More advanced algorithms integrate GPS and prior maps containing TL locations [7], [8] and [2]. These algorithms use GPS information and TL location database to generate a set of candidate regions for TL in the image. Then a machine learning algorithm is applied to recognize the TL state. The main shortcoming of all these algorithms is the lack of ability to identify the TL signal of interest for the driver in images that contain multiple TL signals for multiple lanes street without prior information. This problem is essential to build an autonomous vehicle navigation system.

In this work, a novel automatic system to generate an estimate for the TL signal of interest for the vehicle driver in static images. This system is designed to work on the result of existing systems that detect all TLs in images. Therefore, detecting all TLs in an image is outside the scope of this work, and a simple approach is used to extract all TLs in images to find the performance of the introduced system. This system consists of five stages: (1) color thresholding, (2) morphological image processing, (3) image segmentation, (4) generating sets of candidate regions, (5) Estimating the TL signal of interest. Color thresholding is performed using both the normalized RGB color space and the HSV color space to generate three images for red color, amber color, and green color. Morphological image processing is applied on the three images to fill the holes and gaps between different regions belong to the same TL signal and clean up noise regions. Image segmentation is performed to extract groups of connected pixels that share the same color feature, i.e., red, amber, or green from the three images. Then these groups are filtered using shape constraints to generate sets of candidate regions for red, amber, and green. Finally, a set of spatial constrains is applied to generate an estimate of the TL signal of interest for the driver. These stages are needed to produce a reliable automatic system that helps in building an autonomous vehicle navigation system.

This paper is organized as follows. The related work will be described in this section. Section II presents the methodology and state-of-the-art in TL detection. In section III, the results are presented. Finally, future works and conclusion are provided in section IV.

2 Related Work

Various methods and numerous research for TL detection and recognition have been done and performed. A good literature review paper could be found in [9], [10]. It lists a wide variety of approaches as well as the conditions that these approaches are working on,

Table 1 summarize some references based on used model. Since this paper work is focusing on model-based, related research are discussed in this part.

In [11], authors presented an algorithm that uses fast radial symmetry transformation and color segmentation to detect referenced object based on its color, shape and position. However, it is not focused to detect TL specifically.

In [12], authors proposed an algorithm that can recognize TLs and arrow lights using the digital map and precise vehicle pose which is estimated by a localization module. Despite they achieved a good performance result but their system still needs a prior information to detect TL.

Table 1: Recent review research

ref	**Used technique**	**Method**
[13],[11]	Color segmentation, shape dimension and position	Model-based
[14][15],[5]	Color properties and shape features	
[16]	Canny operator	
[17]	Color properties, fuzzy clustering and Bayesian filters	
[18]	Color properties and hidden Markov model	
[19]	Convolutional neural networks	Learning-based
[7], [20]	Integrating deep learning and multi-sensor data fusion assist	
[21], [22]	Machine learning YOLOv4	
[23]	Convolutional neural network and module, integrated channel feature tracking	

In [15], the authors presents an application of digital image processing technique for TL detection, it involves automated TL detection and response system for an Unmanned Ground Vehicle. The images are provided by the CCTV camera installed on the vehicle and processed for multi levels of detection like color and shape detection. They did also the vehicle density estimation. Their conclusion was that their algorithm has a good detection rate, accuracy, and optimum execution time but it still needs some improvements.

In [14], authors proposed a multi-feature system which combines both color and shape structure by detecting black box region. Moreover, geographic information has been used. Their system, however, suffer from large number of hyper parameters common on model approaches, which need recalibration under certain circumstances.

In [17], the authors present a technique to detect TL based on colors and implemented a fuzzy clustering which provides a better division of the TL colors. Then, the distance between TLs and the autonomous vehicle is estimated by applying Bayesian filters to the TL represented on the frames. Their research still needs more test and improvement to concern other rules in order to be more efficient and more practical.

A method for detecting a TL from a scene image was proposed in [24], the author used first a color space is converted from RGB to the normalized RGB to reduce the effect of the change in the lighting condition. Then candidate regions of a TL are detected by using edges and color information. Finally, a Hough transform is used to detect TL. They state

that the experimental results show the effectiveness of their method. They suggest to consider the position of the TL to increase the accuracy as well as they propose to test their proposal method on large scale images in order to evaluate their proposed method more.

In ADAS system, it is more helpful to integrate an algorithm that is able to identify the TL signal of interest for the driver in images that contain multiple TL signals for multiple lanes street. This work addresses this problem and proposes a system that generates an estimate for the TL state of interest for the driver.

3 The Proposed Method

The process for the proposed system consists of five stages: (1) color thresholding, (2) morphological image processing, (3) image segmentation, (4) generating sets of candidate regions, (5) Estimating the TL signal of interest. By following these stages, the system generates an estimate for the TL signal of interest for the driver in static images. Fig. 1 shows a block diagram for the five stages.

In the first stage, color thresholding is applied to generate three images: red, amber, and green. Each of these images are processed to filter out noise and remove small gaps between regions using morphological image processing. That image is then segmented to generate groups of connected non-white pixels that share the same color. These groups are filtered out using shape constraints to generate a set of candidate regions such that each unique group forms a unique region. Finally, spatial constraints are applied to estimate the TL signal of interest and determine its state. Fig. 5 shows an example for the output of the system. The image contains multiple TL signals. The system draws a rectangle around the estimated TL signal of interest for the driver.

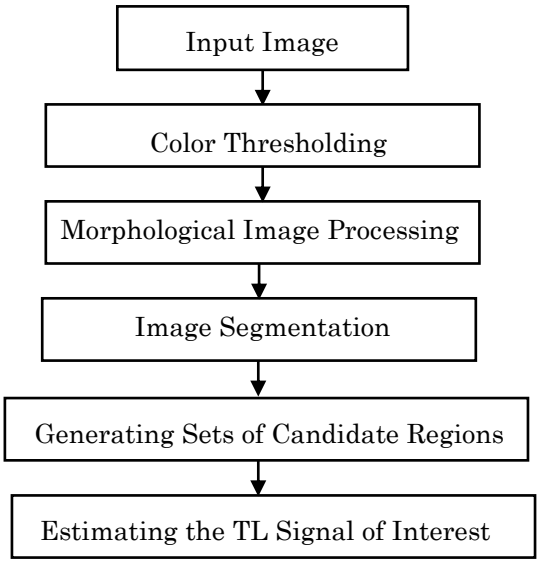


Fig. 1 Flow chart of TL detectior

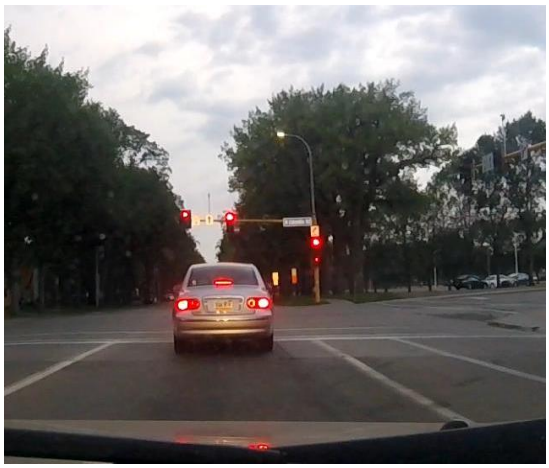


Fig. 2 Example of a scene image taken from vehicle

3.1 Color Thresholding

Color thresholding process is the first stage of this system. It generates labelled images that indicate red, amber, and green pixels in input image. It takes as input the image obtained from the camera as seen in Fig. 2 and generates as output three binary images: red image (Ir), amber image (Ia), and green image (Ig). This process labels each pixel in the input image as: red, amber, green, or white.

Labelling process is performed using the normalized RGB color space [5] and the HSV color space [13]. For each pixel at (x,y), the label L(x,y) is determined red as specified in equation (1).

$$\left\{\begin{array}{c} \mathbf{L(x,y) = RED} \\ \\ Rn(x,y) - Gn(x,y) > 0.35 \quad \& \\ Gn(x,y) - Bn(x,y) < 0.08 \quad \& \\ Gn(x,y) < 0.25 \quad \& \\ (H(x,y) > 354 \;||\; H(x,y) < 8) \;\& \\ V(x,y) > 0.5 \end{array}\right\} \tag{1}$$

Where Rn is the normalized red, Gn is the normalized green, Bn is the normalized blue, H is the hue of the HSV color space in the range from 0 to 360, V is the value of the HSV color space. The label L(x,y) is determined amber and green as specified in equation (2) and (3) respectively:

$$\left\{\begin{array}{c} \mathbf{L(x,y) = AMBER} \\ \\ Rn(x,y) - Gn(x,y) > 0.5 \quad \& \\ Gn(x,y) - Bn(x,y) > 0.15 \quad \& \\ Gn(x,y) > 0.14 \quad \& \\ (H(x,y) > 10 \;\&\; H(x,y) < 39) \;\& \\ V(x,y) > 0.5 \end{array}\right\} \tag{2}$$

$$\left\{\begin{array}{c} \mathbf{L(x,y) = GREEN} \\ \\ Rn(x,y) - Gn(x,y) < -0.15 \quad \& \\ Gn(x,y) - Bn(x,y) > 0 \quad \& \\ Gn(x,y) > 0.25 \quad \& \\ (H(x,y) > 141 \;\&\; H(x,y) < 214) \;\& \\ V(x,y) > 0.5 \end{array}\right\} \tag{3}$$

The label L(x,y) is determined white if conditions in equations (1,2, &3) are not satisfied. The three binary images Ir, Ia, and Ig are generated as specified in equation (4, 5, & 6).

$$Ir(x,y) = \left\{\begin{array}{ll} 128, & L(x,y) = RED \\ 255, & Otherwise \end{array}\right\} \tag{4}$$

$$Ia(x,y) = \left\{\begin{array}{ll} 128, & L(x,y) = AMBER \\ 255, & Otherwise \end{array}\right\} \tag{5}$$

$$Ig(x,y) = \left\{\begin{array}{ll} 128, & L(x,y) = GREEN \\ 255, & Otherwise \end{array}\right\} \tag{6}$$

In Fig. 3a, one generated image for red is an example for this stage. These images are

used to determine the region of interest for TLs.

3.2 Morphological Image Processing

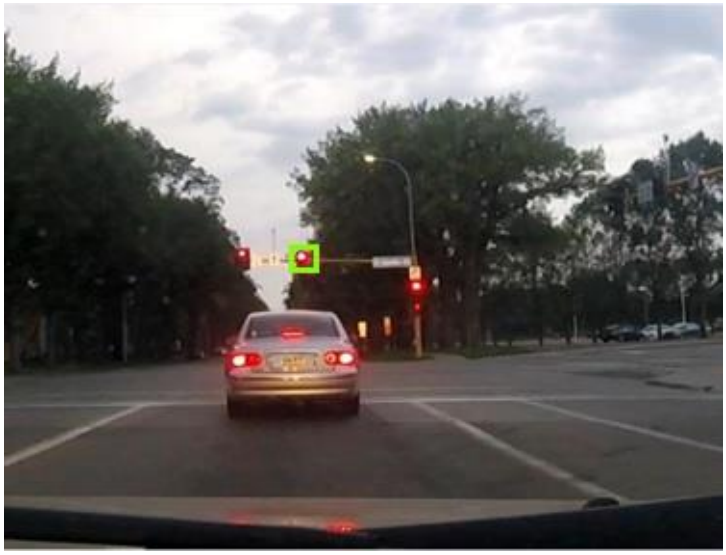
Morphological image processing is the second stage of this system. It applies a nonlinear filter on Ir, Ia, and Ig to reduce noise and remove small gaps between regions. It takes Ir, Ia, Ig generated in the first stage as input. It generates as output three binary images: processed red image (Ipr), processed amber image (Ipa), and processed green image (Ipg). This process performs an erosion filter with window size of n= 5x5 pixels on each input image as specified in equation (7).

$$\begin{Bmatrix} I_{pr} = Erode(I_r,\ n) \\ I_{pa} = Erode(I_a,\ n) \\ I_{pg} = Erode(I_g,\ n) \end{Bmatrix} \quad (7)$$

This filter is chosen to grow the regions that correspond to red, amber, and green colors. The growing of these regions helps in closing small gaps between the regions that occurred due to illumination variation, occlusions, and noises, Fig. 3b shows the result of this stage.

(a) Color Thresholding of Red color

(b) Morphological Image



(c) Detected Red TL

Fig. 3 Applying Images segmentation method

3.3 Image Segmentation

Image segmentation is the third stage of this system. It collects the groups of connected pixels that share the same label from the three processed images Ipr, Ipa, and Ipg. It takes Ipr, Ipa, Ipg generated in the second stage as input. It generates as output three sets: red regions set (Sr), amber regions set (Sa), and green regions set (Sg). Image segmentation is implemented using the region growing method specified in [25]. Using this method, the region is iteratively grown by comparing all unallocated neighbouring pixels to the selected region. The similarity is measured by comparing the label of the unallocated pixel to the label of the region. If they have the same label the pixel is allocated to that region. This process stops when the label of the region is different than new pixels. The group of pixels that are forming a region is then add to its corresponding set. Using these sets, for each group of pixels, it becomes possible to build a bounding box which contains the whole connected area. This method is applied on Ipr, Ipa, and Ipg to generate Sr, Sa, and Sg, respectively. The region growing method requires a lot of iterations that may affect the speed for real time applications. But these regions are relatively small that do not require a lot of computations. Besides, it is possible to resolve this issue by utilizing parallel processing.

3.4 Sets of Candidate Regions

Generating sets of candidate regions is the fourth stage of this system. It filters out Sr, Sa, and Sg from the pixel groups that are less likely to be corresponding to a TL signal. It takes as input Sr, Sa, and Sg that are generated in the third stage. It generates as output three sets: set of candidate regions for red color (Scr), set of candidate regions for amber color (Sca), and set of candidate regions for green color (Scg). This process applies shape constraints to decide whether a group is a candidate region or not. For set Scj as determined in equation (8)

$$\left\{\begin{array}{ll} Scj = Scj \cup Sj(i), \text{where} & \\ W(Sj(i)) > 5 \;\&\; H(Sj(i)) > 5 & \& \\ Ratio(Sj(i)) > 0.67 & \& \\ Ratio(Sj(i)) < 1.5 & \& \\ Area(Sj(i)) \geq [0.7 \times H(Sj(i)) \times W(Sj(i))] & \end{array}\right\} \quad (8)$$

Where W(Sj(i)) and H(Sj(i)) are the width and height of the bounding rectangle for group Sj(i). Ratio(Sj(i)) is the ratio between the height and width of the bounding rectangle for Sj(i), i.e. $Hj(i)/(Wj(i))$, Area(Sj(i)) is the number of pixels in Sj(i).

The factors specified in this equation have been obtained either by analyzing the shapes of the lights of the TL or empirically. The width and height of Sj(i) has to be greater than 5 to filter out noise objects and too small traffic regions, e.g., car brake lights and billboards with similar colors to TLs, since it is using the shape constraint and area constraint only. Resolving this issue requires the usage of complex TL detection algorithms that integrate deep learning with GPS information as described in [2], [26]. In this paper we are focusing in detecting the TL of interest to the driver from the set of detected TLs in the image.

3.5 Estimating the TL Signal of Interest

Estimating the TL signal of interest is the fifth stage of this system. It takes as input Scr, Sca, and Scg that are generated in the fourth stage. It generates as output a bounding rectangle for the TL signal of interest with its state. This process applies spatial constraints to recognize the TL of interest. This process consists of eight steps:

Step 1: Classify the candidate regions in each set in to three categories according to their spatial position: left, center, and right. The classification process is defined in equation (9).

$$\text{Label}(Sj(i)) = \begin{cases} LEFT\ , & LocX(Sj(i)) < T1 \\ RIGHT\ , & LocX(Sj(i)) > T2 \\ CENTER\ , & Otherwise \end{cases} \quad (9)$$

Where Label(Sj(i)) is the spatial label for candidate region Sj(i). j is the color identifier, red, amber, and green, i.e., $j \in \{red, amber, green\}$. LocX(Sj(i)) is the spatial location along the x-axis for the center of the bounding rectangle for candidate region Sj(i). T1 and T2 are the left side and the right side x-location thresholds, respectively. The values of T1 and T2 depend on where the camera is mounted on the car and its parameters. In this work, the camera is mounted at the center of the car, so T1 is set the first third of the image width, and T2 is set to the second third of the image width.

Step 2: From each category select the TL that is closest to the top of the image as defined in equation (10), (11) & (12).

$$\begin{cases} \text{SjLeft} = \text{Sj(i)}, where \\ \text{Y(Sj(i))} < \text{Y(Sj(k))} \quad \& \\ \text{Label(Sj(k))} = \text{Left} \quad \& \\ \text{k} \in [1, \ldots, \text{Nj}] \end{cases} \quad (10)$$

$$\begin{cases} \text{SjRight} = \text{Sj(i)}, where \\ \text{Y(Sj(i))} < \text{Y(Sj(k))} \quad \& \\ \text{Label(Sj(k))} = \text{Right} \quad \& \\ \text{k} \in [1, \ldots, \text{Nj}] \end{cases} \quad (11)$$

$$\begin{cases} \text{Sjcenter} = \text{Sj(i)}, where \\ \text{Y(Sj(i))} < \text{Y(Sj(k))} \quad \& \\ \text{Label(Sj(k))} = \text{Center} \quad \& \\ \text{k} \in [1, \ldots, \text{Nj}] \end{cases} \quad (12)$$

Where Sj_{Left}, Sj_{Right} , Sj_{Center} are the top TL candidate region in Sj on left, right, and center sides of the image. j is the color identifier, i.e., $j \in \{red, amber, green\}$. Y(Sj(i)) is the y-location of the top side of the bounding rectangle for Sj(i). Nj is the number of regions in Sj.

Step 3: For the left categories, i.e., Sr_{left}, Sa_{left} and Sg_{left} select the TL_{left} that is closest to the top of the image as defined in equation (13).

$$\left\{ \begin{matrix} TL_{left} = Sj_{left}, where \\ Y(Sj_{left}) < Y(Sk_{left}) \quad \forall \\ k \in [red, amber, green] \end{matrix} \right\} \tag{13}$$

Step 4: For the center categories, i.e Sr_{center}, Sa_{center} and Sg_{center} select the TL_{center} that is closest to the top of the image as defined in equation (14).

$$\left\{ \begin{matrix} TL_{center} = Sj_{center}, where \\ Y(Sj_{center}) < Y(Sk_{center}) \quad \forall \\ k \in [red, amber, green] \end{matrix} \right\} \tag{14}$$

Step 5: For the right categories, i.e., Sr_{right}, Sa_{right} and Sg_{right} select the TL_{right} that is closest to the top of the image as defined in equation (15).

$$\left\{ \begin{matrix} TL_{right} = Sj_{right}, where \\ Y(Sj_{right}) < Y(Sk_{right}) \quad \forall \\ k \in [red, amber, green] \end{matrix} \right\} \tag{15}$$

Step 6: Specify the estimated range band for valid TL signals in the image. This band has a top and a bottom. The top and bottom values are specified with relative to the spatial position of candidate TL signal that is closest to the image top. This TL is determined as in equation (16) The top and bottom values for the estimated range band are determined as in equations (17 & 18).

$$TL_{top} = \left\{ \begin{matrix} TL_{left}, Y(TL_{left}) < Y(TL_{center}) \\ Y(TL_{left}) < Y(TL_{right}) \\ TL_{center}, Y(TL_{center}) < Y(TL_{left}) \\ Y(TL_{center}) < Y(TL_{right}) \\ TL_{right}, Y(TL_{right}) < Y(TL_{left}) \\ Y(TL_{right}) < Y(TL_{center}) \end{matrix} \right\} \tag{16}$$

$$B_{top} = Y(TL_{top}) - a1 * H(TL_{top}) \tag{17}$$

$$B_{bottom} = Y(TL_{top}) + a2 * H(TL_{top}) \tag{18}$$

Where TL_{top} is the TL signal that is closest to the image top. B_{top} and B_{bottom} are the range band top and bottom values, respectively. $H(TL_{top})$ is the height of the bounding rectangle for TL_{top}. a1 and a2 are multiplication factors and their values specified according to equations (19 & 20), respectively.

$$a1 = \begin{Bmatrix} 1, Label(TL_{top}) = RED \\ 2, Label(TL_{top}) = AMBER \\ 4, Label(TL_{top}) = GREEN \end{Bmatrix} \quad (19)$$

$$a2 = \begin{Bmatrix} 4, Label(TL_{top}) = RED \\ 2, Label(TL_{top}) = AMBER \\ 1, Label(TL_{top}) = GREEN \end{Bmatrix} \quad (20)$$

Notice that the three distinct values, i.e., 1, 2, and 3, are set based on the assumption that the lights in TLs are arranged vertically from top to bottom as Red, Amber, and Green, respectively.

Step 7: Filter out TL candidates that do not lay within the estimated location range band for valid TLs as specified in equation (21).

$$TL_{pos} = \begin{Bmatrix} \{\}, & LocY(TL_{pos}) < B_{top} \\ \{\}, & LocY(TL_{pos}) > B_{bottom} \\ TL_{pos}, & Otherwise \end{Bmatrix} \quad (21)$$

Where pos is the spatial classification of TL, i.e., pos∈{Left, Center, Right).

Step 8: Estimate the TL of interest. The TL of interest is considered the one that is closest to the image center. If center TL_{center} is available then it is considered the one of interest, otherwise, the closest one to the center line of the image is chosen as the estimate as specified in equation (22).

$$TL_{driver} = \begin{Bmatrix} TL_{center}, TL_{center} \neq \{\} \\ TL_{left}, TL_{left} \neq \{\} \\ \& D(TL_{left}) < D(TL_{right}) \\ TL_{right}, TL_{right} \neq \{\} \\ \& D(TL_{right}) < D(TL_{left}) \\ \{\}, Otherwise \end{Bmatrix} \quad (22)$$

Where TL_{driver} is the TL signal of interest. D(.) is the distance between the TL bounding rectangle center and the center of the image.

By following these steps, the TL signal of interest is recognized. Fig. 3c shows an example for the result of this stage.

4 Results, Analysis and Discussions

This system is used to detect TL of interest for the driver of a vehicle from an image that contains zero, one or more TL. The system has been tested on a PC that has (widows7, Intel (R) Core (TM) i3-2330M Duo CPU @ 2.20GHz with 2GB RAM). The algorithm has been written on Matlab.

The system was tested on a public dataset, i.e., LISA TL Dataset [27], [28]. Because the detection of all TLs in the image is out of the scope of this work since we are focusing on TL of interest for the vehicle, we selected 12637 images as a unique image (one image from the same images set) from the overall data set, which the implemented all TLs method was able to detect them successfully with the accuracy as mentioned in Table 2 for each data set. The system works on mixed day time illumination condition as seen in Fig. 4. The maximum accuracy of the system in detecting the TL of interest reached 97.6% for Day Clip 6. Furthermore, the average accuracy for the overall data set images is around 80%. This result is very promising.

The system was tested too on a private database collected from an in-vehicle camera. The settings for the in-vehicle Camera are (Gopro Hero Session) 220×299 pixels (30 frame per second) colored image. The images were picked from the recorded video generated by camera. through a windshield of car while driving. TL could be detected even when its size

. **Table 2:** Summary for LISA data set results

Description		**Number of tested unique image**	**Percentage (%)**
Day Clip 6		411	97.6
	Day Clip 1	2155	71.83
	Day Clip 2	955	51.7
	Day Clip 3	548	64.23
	Day Clip 4	264	47.34
	Day Clip 5	2668	90.44
Day	**Day Clip 6**	412	97.33
Train	**Day Clip 7**	2707	95.79
	Day Clip 8	967	91.52
	Day Clip 9	951	67
	Day Clip 10	42	93
	Day Clip 11	435	95.4
	Day Clip 12	122	69

(a) (b)

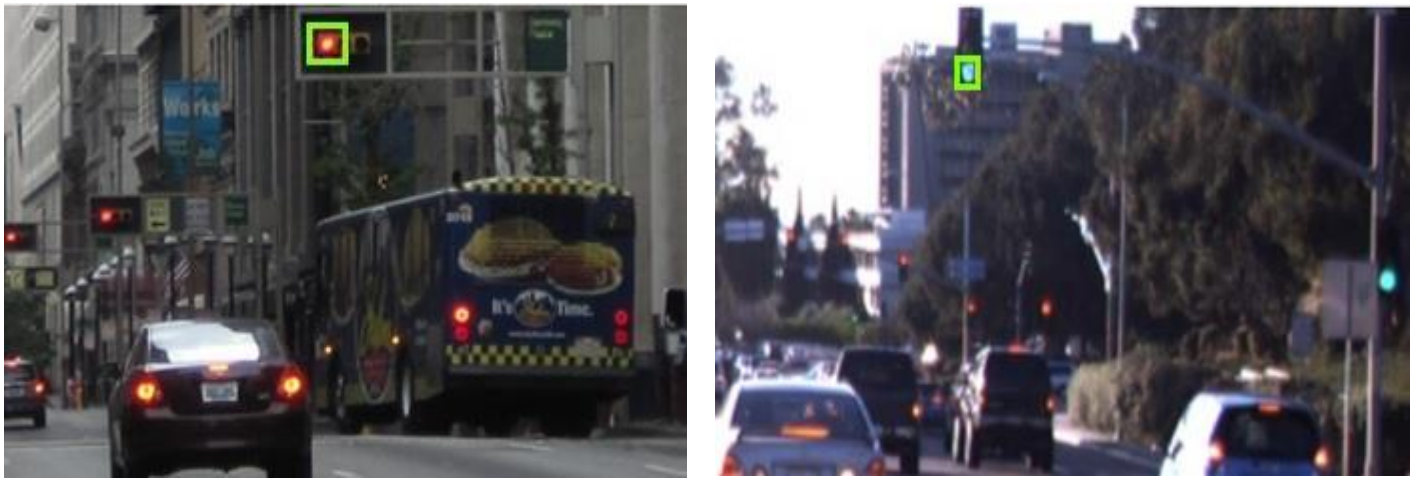
Fig. 4 Images taken from LISA data base [27], [28]

(a) (b)

Fig. 5 Images taken from in-vehicle video camera

is so small with 99%. Fig 5a & 5b are example of that and the purple rectangle shows the detected TL.

This system has the ability to predict which TL is related to the driver without any prior information. For example, when several TLs are simultaneously visible to car driver, each one could be in different light states, the assistance system must be able to select which TL is relevant to the driver, Fig. 6a & 6b shows three upcoming intersections TLs that all visible at the same time, the proposed technique determines correctly which TL is relevant to the driver.

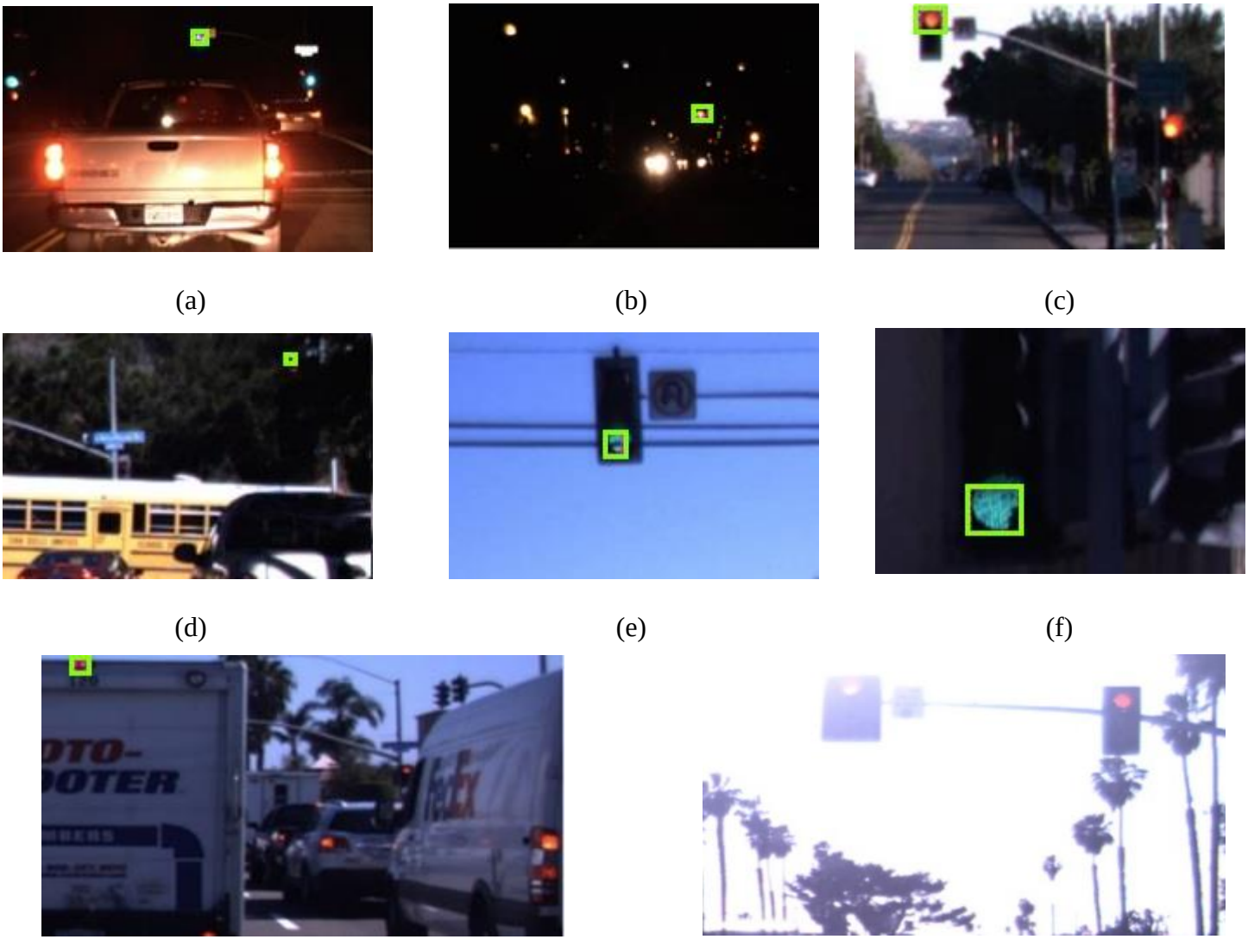


(a) (b)

Fig. 6 Multi TL due to intersection

In [10], the author presented some challenging images in recognizing TL, based on that, these images were tested in this system for the following challenges. First: Color tone shifting and halo disturbances which usually happens due to weather condition that influence other light sources, the system successfully detect it as shown in Fig. 7a, 7b & 7c. Second: Occlusion and partial occlusion due to oblique viewing angles and other objects, refer to Fig. 7d & 7e where the system succeeded in detecting the TL. Finally: incomplete shapes because of malfunctioning or dirty lights, it also happens because of other objects, refer to Fig. 7(f) where the system passes too.

There are two drawbacks for this system: (1) detecting TLs where the car brake light is located above the TL due to the orientation of camera or oblique viewing angles, in this case the brake light will be considered as TL instead of the right one as shown in Fig. 7g.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Fig. 7 Passed frames have difficulties in TL Recognition [10]

This issue can be resolved by having a better TL detection technique as in [26],[2]. (2) In case of the sun is located directly behind the TL as shown in Fig. 7h, where the sun light interferes with the colors of the TLs signals and affect the result.

5 Conclusion

This paper presents a novel system to estimate the TL signal of interest for a vehicle driver in color images obtained from a camera mounted on a car whether these images contain zero, one or multiple TL at intersections. The process for the proposed system consists of five stages: (1) color thresholding using normalized RGB and HSV color space, (2) morphological image processing, (3) image segmentation, (4) generating sets of candidates regions, (5) Estimating the TL signal of interest. The evaluation results show high recognition accuracy of the proposed system in varied illumination conditions. This work may help engineers in improving the performance of advanced driver assistance systems. The accuracy of the system in detecting the TL of interest reached 97.6% as maximum for tested images on LISA data sets with an average accuracy for the overall data set images is around 80%. Furthermore, 99% on the private database collected from an in-vehicle camera has been detected correctly, this result is very promising.

The drawbacks of this system lay in images where the sun is directed behind the TL, or the image has been taken in angle that makes the brake light of car locate above the TL. For future work, this system has the ability to be integrated to detect TL from video streams and apply video tracking algorithms to improve its accuracy. Moreover, its results should be compared with machine learning method and integrate it in its detecting algorithm.

References

[1] C. Badue *et al.*, "Self-driving cars: A survey," *Expert Systems with Applications*, vol. 165. 2021.

[2] V. John, K. Yoneda, B. Qi, Z. Liu, and S. Mita, "Traffic light recognition in varying illumination using deep learning and saliency map," *17th Int. IEEE Conf. Intell. Transp. Syst.*, pp. 2286–2291, 2014.

[3] H. K. Kim, K. Y. Yoo, J. H. Park, and H. Y. Jung, "Traffic light recognition based on binary semantic segmentation network," *Sensors (Switzerland)*, 2019.

[4] M. Diaz-Cabrera and P. Cerri, "Traffic light recognition during the night based on fuzzy logic clustering," *Comput. Aided Syst. Theory-EUROCAST 2013. Springer-Verlag Berlin Heidelb.*, no. PART 2, pp. 93–100, 2013.

[5] M. Diaz-Cabrera, P. Cerri, and J. Sanchez-Medina, "Suspended traffic lights detection and distance estimation using color features," *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, pp. 1315–1320, 2012.

[6] G. Trehard, E. Pollard, B. Benazouz, and N. Fawzi, "Tracking both pose and status of a traffic light via an Interacting Multiple Model filter," *Fusion 2014-17th Int. Conf.*, pp. 1–7, 2014.

[7] Z. Li, Q. Zeng, Y. Liu, J. Liu, and L. Li, "An improved traffic lights recognition algorithm for autonomous driving in complex scenarios," *Int. J. Distrib. Sens. Networks*, vol. 17, no. 5, 2021.

[8] F. Lindner, U. Kressel, and S. Kaelberer, "Robust recognition of traffic signals," *IEEE Intell. Veh. Symp. 2004*, pp. 49–53, 2004.
[9] K. Yojitha, B. B. Priya, N. H. Krishna, D. Yashwanth, and G. Anuradha, "A Survey on Obstacle Avoidance and Traffic Light Detection Approaches for Autonomous Vehicle," in *2021 International Conference on Intelligent Technologies, CONIT 2021*, 2021.
[10] M. B. Jensen, M. P. Philipsen, A. Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Vision for Looking at Traffic Lights: Issues, Survey, and Perspectives," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 1800–1815, 2016.
[11] R. M. Shakirzyanov and A. A. Shakirzyanova, "Object Detection Using Color Segmentation, Radial Symmetry Detector, and Hough Method," in *Proceedings - 2021 International Russian Automation Conference, RusAutoCon 2021*, 2021.
[12] K. Yoneda, A. Kuramoto, N. Suganuma, T. Asaka, M. Aldibaja, and R. Yanase, "Robust traffic light and arrow detection using digital map with spatial prior information for automated driving," *Sensors (Switzerland)*, vol. 20, no. 4, 2020.
[13] W. G. Shadeed, D. I. Abu-Al-Nadi, and M. J. Mismar, "Road traffic sign detection in color images," in *Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems*, 2003.
[14] Y. Zhang, J. Xue, G. Zhang, Y. Zhang, and N. Zheng, "A multi-feature fusion based traffic light recognition algorithm for intelligent vehicles," in *Proceedings of the 33rd Chinese Control Conference, CCC 2014*, 2014.
[15] N. Bhattacharayya, A. K. Singh, S. Kapil, H. Jain, and A. Jain, "Traffic Light Solution for U . G . V . Using Digital Image Processing," *Int. J. Soft Comput. Eng.*, vol. 4, no. March, pp. 25–27, 2014.
[16] L. Yuming and G. Shuqing, "Traffic signal light detection and recognition based on canny operator," *J. Meas. Eng.*, vol. 9, no. 3, 2021.
[17] M. Diaz-Cabrera, P. Cerri, and P. Medici, "Robust real-time traffic light detection and distance estimation using a single camera," *Expert Syst. Appl.*, vol. 42, no. 8, 2015.
[18] A. E. Gomez, F. A. R. Alencar, P. V. Prado, F. S. Osorio, and D. F. Wolf, "Traffic lights detection and state estimation using Hidden Markov Models," in *IEEE Intelligent Vehicles Symposium, Proceedings*, 2014.
[19] T. W. Yeh, H. Y. Lin, and C. C. Chang, "Traffic light and arrow signal recognition based on a unified network," *Appl. Sci.*, vol. 11, no. 17, 2021.
[20] S. G. Sodjinou, V. Mohammadi, A. T. Sanda Mahama, and P. Gouton, "A deep semantic segmentation-based algorithm to segment crops and weeds in agronomic color images," *Inf. Process. Agric.*, vol. 9, no. 3, 2022.
[21] Q. Wang, Q. Zhang, X. Liang, Y. Wang, C. Zhou, and V. I. Mikulovich, "Traffic lights detection and recognition method based on the improved yolov4 algorithm," *Sensors*, vol. 22, no. 1, 2022.
[22] M. Shirpour, N. Khairdoost, M. A. Bauer, and S. S. Beauchemin, "Traffic Object Detection and Recognition Based on the Attentional Visual Field of Drivers," *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, 2023.
[23] K. Wang, X. Tang, S. Zhao, and Y. Zhou, "Simultaneous detection and tracking using deep learning and integrated channel feature for ambint traffic light recognition," *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 1, 2022.
[24] M. Omachi, "Traffic light detection with color and edge information," *2009 2nd IEEE Int. Conf. Comput. Sci. Inf. Technol.*, pp. 284–287, 2009.
[25] D.-J. Kroon, "region growing algorithm." [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/19084-region-growing/content/regiongrowing.m.
[26] M. Che, M. Che, Z. Chao, and X. Cao, "Traffic light recognition for real scenes based on image processing and deep learning," *Comput. Informatics*, 2020.

[27] M. B. Jensen, M. P. Philipsen, A. ; Møgelmose, T. B. Moeslund, and M. M. Trivedi, "Vision for Looking at Traffic Lights: Issues, Survey, and Perspectives. I E E E Transactions on Intelligent Transportation Systems," *Ieee Trans. Intell. Transp. Syst.*, 2016.

[28] M. P. Philipsen, M. B. Jensen, A. Mogelmose, T. B. Moeslund, and M. M. Trivedi, "Traffic Light Detection: A Learning Algorithm and Evaluations on Challenging Dataset," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2015.

Dr. Reem Shadid received the B.Sc. and M.Sc. degrees in electrical engineering from The University of Jordan, in 2003 and 2015, respectively, and the Ph.D. degree in electrical engineering from the University of North Dakota, USA, in 2018. She was working as head of transmission design section in National Electric Power Company (NEPCO). She is currently an Associate Professor at the Department of Electrical Engineering, Applied Science Private University. Her researches focus on image processing, power systems, and wireless power transfer

Dr. Waseem G. Shadid received the B.Sc. and M.Sc. degrees in electrical engineering from The University of Jordan, in 2001 and 2004, respectively, and the Ph.D. degree in electrical and computer engineering from The University of North Carolina at Charlotte, USA, in 2014. He is currently an Affiliated Professor at The University of North Carolina at Charlotte, where he supervises the research at the UNC Charlotte CyberDNA Research Center. His researches focus on electromagnetic theory, process modeling, machine learning/AI, data science, and 3D computer vision in a range of domains, including energy, cybersecurity, physics, and medical imaging. His research led to many real-world applications that are used worldwide, patents, and publications. He was involved in the projects sponsored by NSF, DARPA, and NASA.

Yara Khawaja is currently an assistant professor at Applied Science Private University, Jordan. She received her Ph.D. degree in Renewable Energy/Electrical Engineering from the School of Engineering, Newcastle University, Newcastle upon Tyne – UK in 2019, and her M.Sc. (with distinction) in Computer Science from Prince Abdullah bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University–Jordan in 2013. Yara received her B.Sc. degree in Power Engineering from the Hijjawi Faculty of Engineering Technology, Yarmouk University – Jordan, in 2007. Her current research interests include signal processing, sizing Microgrids containing distributed generation, energy management and control of Microgrids.