REAL TIME TRAFFIC LIGHT CONTROL USING IMAGE PROCESSING

Ms.PALLAVI CHOUDEKAR

Ajay Kumar Garg Engineering College, Department of electrical and electronics
Ghaziabad,UP ,India

Ms.SAYANTI BANERJEE

Ajay Kumar Garg Engineering College, Department of electrical and electronics
Ghaziabad,UP ,India

Prof.M.K.MUJU

Ajay Kumar Garg Engineering College, Department of Mechanical
Ghaziabad,UP ,India

Abstract

As the problem of urban traffic congestion spreads, there is a pressing need for the introduction of advanced technology and equipment to improve the state-of-the-art of traffic control. Traffic problems nowadays are increasing because of the growing number of vehicles and the limited resources provided by current infrastructures. The simplest way for controlling a traffic light uses timer for each phase. Another way is to use electronic sensors in order to detect vehicles, and produce signal that cycles. We propose a system for controlling the traffic light by image processing. The system will detect vehicles through images instead of using electronic sensors embedded in the pavement. A camera will be installed alongside the traffic light. It will capture image sequences. The image sequence will then be analyzed using digital image processing for vehicle detection, and according to traffic conditions on the road traffic light can be controlled..

***Keywords*:** Intelligent Transportation System (ITS), Traffic light, Image Processing, edge detection.

1. Introduction

Automatic traffic monitoring and surveillance are important for road usage and management. Traffic parameter estimation has been an active research area for the development of intelligent Transportation systems (ITS). For ITS applications traffic- information needs to be collected and distributed. Various sensors have been employed to estimate traffic parameters for updating traffic information. Magnetic loop detectors have been the most used technologies, but their installation and maintenance are inconvenient and might become incompatible with future ITS infrastructure. It is well recognized that vision-based camera system are more versatile for traffic parameter estimation [l,4]. In addition to qualitative description of road congestion, image measurement can provide quantitative description of traffic status including speeds, vehicle counts, etc. Moreover, quantitative traffic parameters can give us complete traffic flow information, which fulfills the requirement of traffic management theory. Image tracking of moving vehicles can give us quantitative description of traffic flow[3].
In the present work the designed system aims to achieve the following.

- Distinguish the presence and absence of vehicles in road images;
- Signal the traffic light to go red if the road is empty;
- Signal the traffic light to go red if the maximum time for the green light has elapsed even if there are still vehicles present on the road.

Components of the current project

- Hardware module
- Software module
- Interfacing

Hardware Module
Image sensors: In this project a USB based web camera has been used.
Computer: A general purpose PC as a central unit for various image processing tasks has been used.
Platform: consisting of a few toy vehicles and LEDs (prototype of the real world traffic light control system).
Software Module
MATLAB version 7.8 as image processing software comprising of specialized modules that perform specific

tasks has been used.

Interfacing

The interfacing between the hardware prototype and software module is done using parallel port of the personal computer. Parallel port driver has been installed in the PC for this purpose.

2. Methodology

Following are the steps involved

- Image acquisition
- RGB to gray conversion
- Image enhancement
- Image matching using edge detection

Procedure

Phase1:

- Initially image acquisition is done with the help of web camera
- First image of the road is captured, when there is no traffic on the road
- This empty road's image is saved as reference image at a particular location specified in the program
- RGB to gray conversion is done on the reference image
- Now gamma correction is done on the reference gray image to achieve image enhancement
- Edge detection of this reference image is done thereafter with the help of Prewitt edge detection operator

Phase2:

- Images of the road are captured.
- RGB to gray conversion is done on the sequence of captured images
- Now gamma correction is done on each of the captured gray image to achieve image enhancement
- Edge detection of these real time images of the road is now done with the help of prewitt edge detection operator

Phase3:

- After edge detection procedure both reference and real time images are matched and traffic lights can be controlled based on percentage of matching.
- If the matching is between 0 to 10% - green light is on for 90 seconds. If the matching is between 10 to 50% - green light is on for 60 seconds. If the matching is between 50 to 70% - green light is on for 30 seconds. If the matching is between 70 to 90% - green light is on for 20 seconds. If the matching is between 90 to 100% - red light is on for 60 seconds.

3. Image Enhancement

The acquired image in RGB is first converted into gray. Now we want to bring our image in contrast to background so that a proper threshold level may be selected while binary conversion is carried out. This calls for image enhancement techniques. The objective of enhancement is to process an image so that result is more suitable than the original image for the specific application. There are many techniques that may be used to play with the features in an image but may not be used in every case. Listed below are a few fundamental functions used frequently for image enhancement.

- Linear (negative and identity transformations)
- Logarithmic (log and inverse log transformations)
- Power law transformations(gamma correction)
- Piecewise linear transformation functions

The third method i.e., power law transformation has been used in this work. The power law transformations have the basic form

$$s = cr^{\gamma}$$

Where S is output gray level, r is input gray level, c and γ are positive constants. For various values of gamma applied on an acquired image we obtained the following graph shown in figure1.

From this figure it is evident that the power law curves with fractional values of γ map a narrow range of dark input values into a wide range of output values with the opposite being true for higher values of input levels. It depicts the effect of increasing values of γ >1. The images are shown with γ =1,2,3,4,5 as may be seen, the figure with γ =1 gives the best results in terms of making fine details identifiable. As is evident the fractional

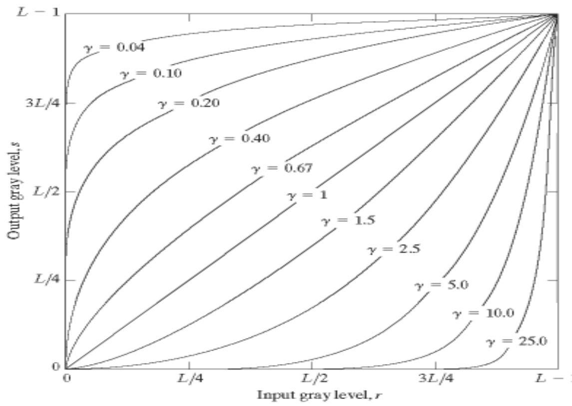


Fig .1

values of γ can not be used since these attempts show a reverse effect of brightening the image still further which we may find as undesirable for the present case.

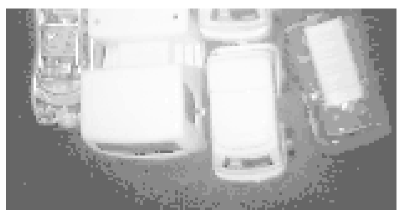


Fig 2 Gamma = 0.2

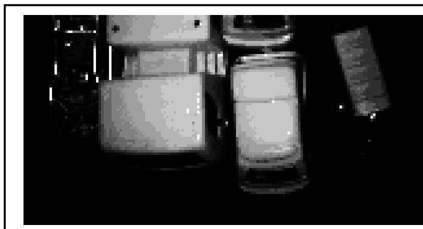


Fig 3 Gamma = 4

4. Edge Detection and Image Matching

Step 1: Edge detection: Among the key features of an image i.e. edges, lines, and points, we have used edge in our present work which can be detected from the abrupt change in the gray level. An edge essentially demarcates between two distinctly different regions, which means that an edge is the border between two different regions.

Here we are using edge detection method for image matching:

- Edge detection methods locate the pixels in the image that correspond to the edges of the objects seen in the image.
- The result is a binary image with the detected edge pixels.
- Common algorithms used are Sobel, Prewitt and Laplacian operators.

We have used gradient based Edge Detection that detects the edges by looking for the maximum and minimum in the first derivative of the image.

- First derivative is used to detect the presence of an edge at a point in an image.
- Sign of the second derivative is used to determine whether an edge pixel lies on the dark or light side of an edge.

The change in intensity level is measured by the gradient of the image. Since an image f (z, y) is a two dimensional function, its gradient is a vector

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{df}{dx} \\ \frac{df}{dy} \end{bmatrix} \quad (1)$$

The magnitude of the gradient is given by

$$G[f(x,y)] = \sqrt{Gx^2 + Gy^2} \quad (2)$$

The direction of the gradient is

$$B(z, y) = \tan^{-1}(G_y/G) \quad (3)$$

where the angle *B* is measured with respect to the X-axis. Gradient operators compute the change in gray level intensities and also the direction in which the change occurs. This is calculated by the difference in values of the neighboring pixels, i.e., the derivatives along the X-axis and Y-axis. In a two-dimensional image the gradients are approximated by

$$Gx = f(i+1,j) - f(i,j) \quad (4)$$

$$Gy = f(I,j+1) - f(i,j) \quad (5)$$

Gradient operators require two masks, one to obtain the X-direction gradient and the other to obtain the Y-direction gradient. These two gradients are combined to obtain a vector quantity whose magnitude represents the strength of the edge gradient at a point in the image and whose angle represents the gradient angle.

The edge detection operator we have used in the present work is Prewitt. Mathematically, the operator uses 3X3 kernels which are convolved with the original image to calculate approximations of the derivatives – one for horizontal changes, and one for vertical. If we define A as the source image, and G_x and G_y are two images which at each point contain the horizontal and vertical derivative approximations, the later are computed as

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} * A \ \& \ G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} * A \quad (6)$$

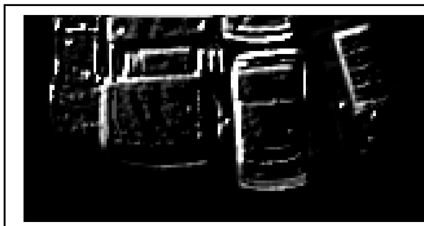
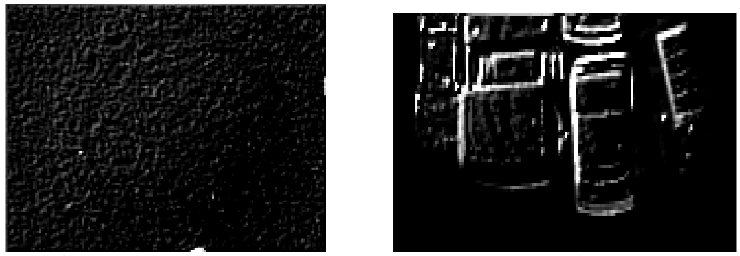


Fig 4
Edge detected image (Prewitt)

Step 2: Image matching: Edge based matching is the process in which two representatives (edge) of the same objects are pared together. Any edge or its representation on one image is compared and evaluated against all the edges on the other image.

Edge detection of reference and the real time images has been done using Prewitt operator. Then these edge detected images are matched and accordingly the traffic light durations can be set.



Reference image captured image

Fig.5

5. Experimental Results

Experiments are carried out and depending upon the intensity of the traffic on the road we get the following results regarding on time durations of various traffic lights.

Result 1: Matching between 10 to 50% - green light on for 60 seconds
Result 2: Matching between 50 to 70% - green light on for 30 seconds
Result 3: Matching between 70 to 90% - green light on for 20 seconds
Result 4: Matching between 90 to 100% - red light on for 60 seconds

6. Summary and Conclusions

The study showed that image processing is a better technique to control the state change of the traffic light. It shows that it can reduce the traffic congestion and avoids the time being wasted by a green light on an empty road. It is also more consistent in detecting vehicle presence because it uses actual traffic images. It visualizes the reality so it functions much better than those systems that rely on the detection of the vehicles' metal content. Overall, the system is good but it still needs improvement to achieve a hundred percent accuracy.

References

[1] David Beymer, Philip McLauchlan, Benn Coifman, and Jitendra Malik, "A real-time computer vision system for measuring traffic parameters," IEEE Conf. on Computer Vision and Pattern Recognition, pp495 -501,1997..

[2] M. Fathy and M. Y. Siyal, "An image detection technique based on morphological edge detection and background differencing for real-time traffic analysis," Pattern Recognition Letters, vol. 16, pp. 1321-1330, Dec 1995

[3] N. J. Ferrier, S. M. Rowe, A. Blake, "Real-time traffic monitoring," Proceedings of the Second IEEE Workshop on Applications of Computer Vision, pp.81 -88, 1994.

[4] Rita Cucchiara, Massimo Piccardi and Paola Mello, "Image analysis and rule-based reasoning for a traffic

[5] monitoring system," IEEE Trans. on Intelligent Transportation Systems, Vol. 1, Issue 2, pp 119-130, 2000.

[6] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications,"Image and Vision Computing, vol. 21, pp. 359-381, Apr 1 2003.