

Street Light Intensity Controller Using Density Mapping Mechanism

Yogesh Yadav¹, Rajas Walavalkar², Abhishek Yedurkar³, Sagar Suchak⁴, Swapnil Gharat⁵

¹Student, Dept of Computer Engineering, RGIT, Maharashtra, India.

²Student, Dept of Computer Engineering, RGIT, Maharashtra, India.

³Student, Dept of Computer Engineering, RGIT, Maharashtra, India.

⁴Student, Dept of Computer Engineering, RGIT, Maharashtra, India.

⁵Lecturer, Dept of Computer Engineering, RGIT, Maharashtra, India.

Abstract - Roadway Illumination is one of the vital component of safety while driving during the night. Rules and regulations have been placed to ensure all publicly maintained roads are properly lit, but the management and maintenance process is too time consuming, costly and there is no central control over street lamps. In addition, the required information for analytics process is also not maintained for further research. The aim of this project is to construct an intelligent street lighting system capable of ensuring efficient electricity consumption of street lamps as well as having central control over it. The proposed system introduces real time remote light intensity monitoring system using Raspberry Pi. Raspberry Pi is a low cost ARM powered linux based computer which acts as a server, it communicates with client with LAN or wifi module. The proposed system consists of a prototype model that includes camera sensor, raspberry pi model 3, relay switch model, designed PCB circuit and LED module. The proposed system will map the traffic density on the road at any instance and in turn will adjust the intensity value of the street lamps, the street lamps will lit with the variable intensity depending on the traffic of the road. The data recorded will be stored in the database for future use, thus helping to repair poorly lit roads and possible safety concerns.

Key Words: Haar-cascade, OpenCV, PCB (Printed Circuit Board), Relay switch module, Wamp Server, MOG(measure of Gaussian), GPIO (general purpose input output) pins.

1.INTRODUCTION

Fast transportation systems are important factors for the economic growth and development of any nation. . Street lighting system plays an important role in maintaining the transport system of a nation. Present Lighting systems are designed according to the old standards of reliability and they often do not take advantage of the latest technological developments. It is estimated that drivers are 2.5 times more likely to be involved in the accident during night hours due to poor illumination on the roadways. Street lighting system consists of street lamps being installed on the roads. A poorly lit roadway does not necessarily imply lack of lighting, but the illumination on the roads that are

inconsistent throughout the roads, thus leaving a high contrast between bright and dark areas. There is no central control on the street lamps i.e. if any of the street lamp gets damaged due to external or internal factors, it remains in the same state as information is not conveyed to the centre[1]. This can lead to road accidents which are a major concerns. In addition, street lamps remains continuously on even if there is no traffic or vehicle on the road. This leads a huge amount of energy consumption with very low efficiency[3]. The intensity of the street lamps gets affected due to adverse weather conditions, thus proper intensity of lamps is not present on the required patch of road which can lead to accidents. Considering the current situation where population of humans is growing exponentially and limitation in the availability of natural resources, it is a vital concern on how we use the available resources. It requires a constructive effort that resources should be used in an optimum manner to make sure it remains available for the future generation. Thus we need to develop ways and techniques to use the resources smartly and intelligently. In case of street lighting system, present day energy consumption is huge with very low efficiency, thus we need to develop an intelligent street lighting system where energy consumption is optimum[2]. The proposed street lighting system will control the intensity of the street lamps on the basis of the traffic on the road, if no traffic it will lit with threshold intensity[3].

The conventional method for measuring roadway illumination involves spot checking along a grid set up in the selected road. Measurements are taken using a light meter placed six inches above the ground. The operator must set up the light meter, stand back, and trigger a measurement. The light meter is then moved to the next point on the grid and the process is repeated. This method is both time consuming and dangerous[2]. It is a lengthy process for the operator to set up the collection device in the roadway, step back, and record a reading. This leaves the operator and the testing equipment vulnerable to the dangers of the road for prolonged periods of time. This danger can be avoided with lane closures and police presence, but doing so adds additional cost to the measurement process. Due to these factors, extended roadway illumination studies are rarely implemented, leaving questions on whether the roads meet the standards for sufficient lighting.

In order to properly collect large amounts of illumination data in a safe and time effective manner we must be able to take measurements[4]from a moving vehicle. This would eliminate the dangers of pedestrians in the roadway without road closures and expensive police presence. It is required that the proposed system be accurate and automated, allowing simple data collection while the vehicle is traveling.

The proposed system aims to construct an intelligent street lighting system that will overcome the disadvantages of conventional lighting system by making use of current technological developments[2]. Two major disadvantages i.e no central control over street lamps and huge electricity consumption. The proposed system includes camera module, Raspberry Pi[2] model 3, relay switch model, designed PCB circuit module and LED module . The workflow of the proposed system is divided into three stages as follows:

Stage 1: The Raspberry pi module[2] is installed on the street lamp along with the camera module for a given patch of road. The other street lamps on the road ahead of initial first street lamp where raspberry pi is installed is connected in series through a wired network. The camera module installed is feed with real time traffic on the roads. The camera module captures the traffic on road frame by frame and transmit frame after every 5 seconds or 20 frames to the raspberry pi module. The raspberry pi module performs real time video processing[4] with the help of haar cascade application. The application detects number of vehicles present in the frame for a given instance[3], then compares the density of the traffic on road to standards values[2], if it matches to any of the standard condition it outputs an intensity value for the street lamp. Haar cascade[7] application thus detects number of vehicles on the road for a given instance and outputs an intensity value for the same.

Stage 2: The intensity value obtained from the stage 1 is then transmitted to the street lamp on which the raspberry pi is installed[2]. The street lamp will lit with that intensity, also street lamps in series connected through a wired network will be provided with that intensity value. Thus all the street lamps will lit with the same intensity value for a given interval of time. If the camera captures no vehicles then it means there is no traffic on the roads, thus in such situations the system is designed to ensure street lamps lit with threshold intensity[3]. Thus, ensuring that there is central control over all the street lamps and also electricity consumption efficiency is increased manifolds.

Stage 3: It involves data that is captured and processed to be stored in the database for future use eg. Analytics purpose.

2. HARDWARE DEVELOPMENT

The hardware architecture as shown in fig.1 has following Components: Camera Sensor Module, Raspberry Pi Module[2], Relay Switch Model, PCB Circuit Module, LED Module.

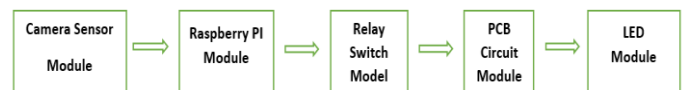


Figure 1

2.1 Camera Sensor Module

The task of the camera sensor is to identify the passage of a vehicle or pedestrian, giving an input to turn on a lamp or a group of lamps. This function depends on the pattern of the street[4]; in case of a street without crossroads, a single sensor is sufficient (or one at each end in case of a two-way street), while for a street requiring more precise control, a solution with multiple presence detectors is necessary. This feature enables switching on the lamps only when necessary, avoiding a waste of energy. The main challenge with such a sensor is its correct placement. The sensor should be placed at an optimal height[4], not too low (i.e., to avoid any erroneous detection of small animals) nor too high (for example, to avoid failure to detect children). A study of the sensor placement enables deciding the optimal height according to the user needs and considering the specific environment in which the system will work.

2.2 Raspberry Pi Module

Raspberry Pi[2](shown in Fig 2) is a credit-card-sized ARM powered Linux computer developed in the UK by Raspberry Pi foundation with the intention of stimulating the teaching of basic computer science in school .The proposed system use Raspberry PI 3- the third generation Raspberry Pi.[6] Compared to model 2 it has:

- 1.2GHz 64-bit quad-core ARMv8 CPU
- 802.11n Wireless LAN
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)
- 2.1 Raspberry Pi Module



Figure 2

The foundation provides following operating systems RASPBIAN, PIDORA, OPENELEC, RASPBM, RISC OS[6] and also Python as the main programming language, with the support for BBC BASIC, C and Perl. The GPU[5] is capable of Bluray quality playback, using H.264 at 40MBits/s. It has a fast 3D core accessed using the supplied OpenGL ES2.0 andOpen VG libraries.

2.3 Relay Switch Module

Relay models are switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. As relay diagrams show, when a relay contact is normally open (NO), current does not flow through that circuit. When a relay contact is Normally Closed (NC), current flows through the circuit. Thus relay models selects the appropriate circuit based on the density value passed to the model and adjust the intensity of the street lamp at that instance.

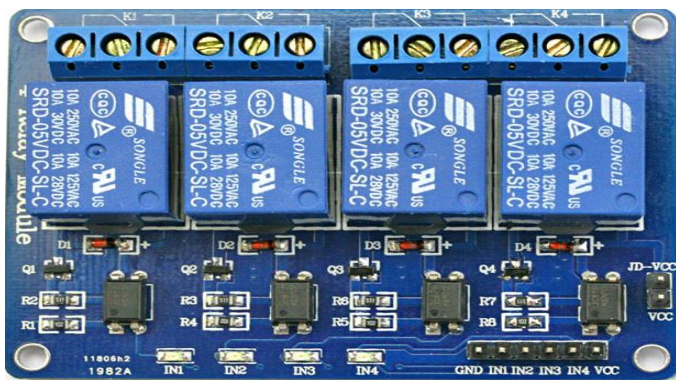


Figure 3

2.4 PCB Circuit Module

In the PCB Circuit diagram, the 3 LEDs are connected in series with a voltage of 9 volt applied to it. The relay switch model selects a given circuit depending on the density value and mapping condition. By default circuit 1 remains open as soon as power supply is connected. Jumping wires are used

for connection between raspberry pins[8] and relay models as well as relay model and pcb circuit. Three registers are used to control the flow of current through the LEDs to make sure the LEDs lit with an appropriate intensity value.

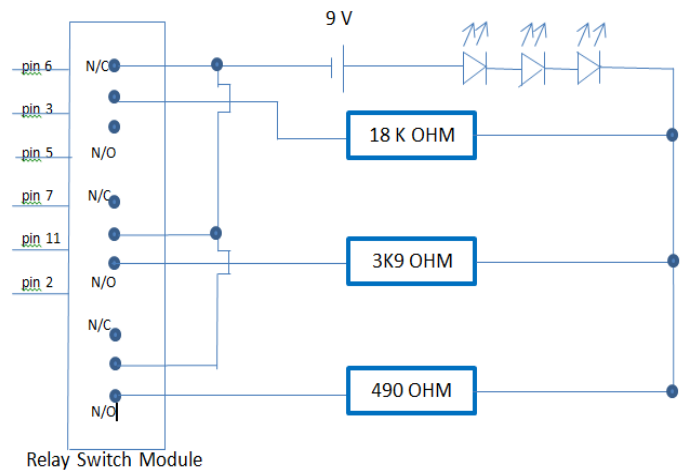


Figure 4

2.5 LED Module

The three LEDs are used to display the results of the implementation i.e to ensure the circuit[8] is properly connected and the working is precise is understood by observing the intensity with which the LEDs are lit for a particular instance. If everything is proper, the LEDs will lit with intensities : threshold, medium, high.

3. SOFTWARE DEVELOPMENT

3.1 OpenCV and Haar Cascade Classifiers[7]:

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. Object detection is one of its application which can be implemented using haar cascade[7]. Haar based cascade classifier is an effective object detection method which uses machine learning approach where a cascade function is trained with a lot of positive and negative images.it is then used to detect objects in other images.

After a classifier is trained, it can be applied to a region of interest (of the same size as used during the training) in an input image. The classifier outputs a "1" if the region is likely to show the object (i.e., face/car)[4], and "0" otherwise. To search for the object in the whole image one can move the search window across the image and check every location

using the classifier[8]. The classifier is designed so that it can be easily “resized” in order to be able to find the objects of interest at different sizes, which is more efficient than resizing the image itself. So, to find an object of an unknown size in the image[5] the scan procedure should be done several times at different scales.

The word “cascade” in the classifier name means that the resultant classifier consists of several simpler classifiers (stages) that are applied subsequently to a region of interest until at some stage the candidate is rejected or all the stages are passed[8]. The word “boosted” means that the classifiers at every stage of the cascade are complex themselves and they are built out of basic classifiers i.e a decision tree with at least 2 leaves.

3.2 Python Socket programming

Sockets are the endpoints of a bidirectional communications channel. Sockets may communicate within a process, between processes on the same machine, or between processes on different continents. Python provides two levels of access to network services. At a low level, you can access the basic socket support in the underlying operating system, which allows you to implement clients and servers for both connection-oriented and connectionless protocols. Python also has libraries that provide higher-level access to specific application-level network protocols, such as FTP, HTTP, and so on.

A Simple Server:

To write Internet servers, we use the socket function available in socket module to create a socket object. A socket object is then used to call other functions to setup a socket server. Now call bind(hostname, port) function to specify a port for your service on the given host. Next, call the accept method of the returned object. This method waits until a client connects to the port you specified, and then returns a connection object that represents the connection to that client.

A Simple Client:

Let us write a very simple client program which opens a connection to a given port 12345 and given host. This is very simple to create a socket client using Python’s socket module function. The socket.connect(hostname, port) opens a TCP connection to hostname on the port. Once you have a socket open, you can read from it like any IO object. When done, remember to close it, as you would close a file.

3.3 My SQL Database and Wamp Server

Wamp server Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing, but may also be used to serve live websites. WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server. The files/web pages that are hosted on a WAMP server can be accessed by typing http://localhost or http://127.0.0.1 in the web browser’s address bar. WAMP must be running in order to access either of the above addresses.

4. DESIGN AND DEVELOPMENT DETAILS

4.1 System Workflow

The proposed system workflow is divided into 2 stages as shown in below fig. starting from image frame send from camera module to the Raspberry Pi[2] , processing of image frame with the help of haar cascade[8] program for object detection in OpenCV of Raspberry Pi , transmission of density mapped intensity data to the relay model as well as server database and finally transmission of resultant intensity to the LEDs.

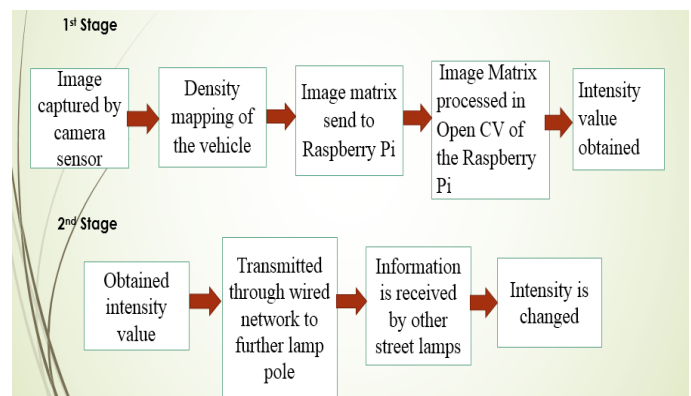


Figure 5

5. IMPLEMENTATION DETAILS

On base OS, Run the Wamp server and restart all services in the Wamp server. In addition, run the server side program on the command prompt.

```
C:\Python27>python.exe ser.py
16/04/2017
101
SU ROAD
MEDIUM
HIGH
HIGH
HIGH
HIGH
MEDIUM
MEDIUM
LOW
MEDIUM
HIGH
MEDIUM
('Total Row(s):', 11)
('101', 'SU ROAD', 'MEDIUM', 'HIGH', 'HIGH', 'HIGH', 'HIGH', 'MEDIUM', 'MEDIUM', 'LOW', 'MEDIUM', 'HIGH', 'MEDIUM', None)
```

Figure 6

On Raspberry pi, enter the following commands on the terminal to and then run the haar cascade program.

```
pi@raspberrypi:~ $ source ~/.profile
pi@raspberrypi:~ $ workon cv
(cv) pi@raspberrypi:~ $ cd harcascade/
(cv) pi@raspberrypi:~/harcascade $ cd haarc/
(cv) pi@raspberrypi:~/harcascade/haarc $ python test.py
^CTraceback (most recent call last):
  File "test.py", line 12, in <module>
    ss.connect(("192.168.137.3", port))
  File "/usr/lib/python2.7/socket.py", line 224, in meth
    return getaddrinfo(self._sock.name)(*args)
KeyboardInterrupt
(cv) pi@raspberrypi:~/harcascade/haarc $ python test.py
```

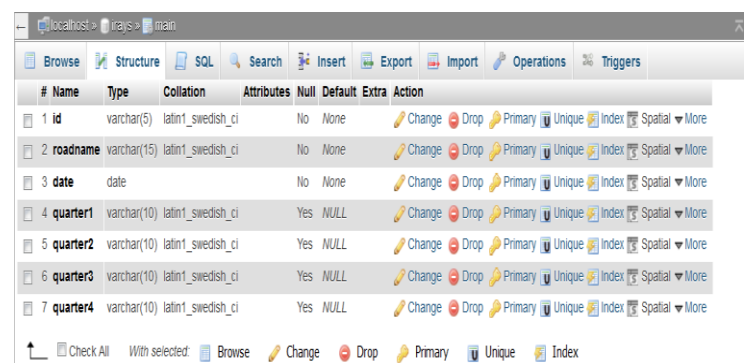
Figure 7

The Output displays two windows, in the first window the real time video feed of the road is displayed [4] and in the second window, detection of number of car vehicles is displayed.



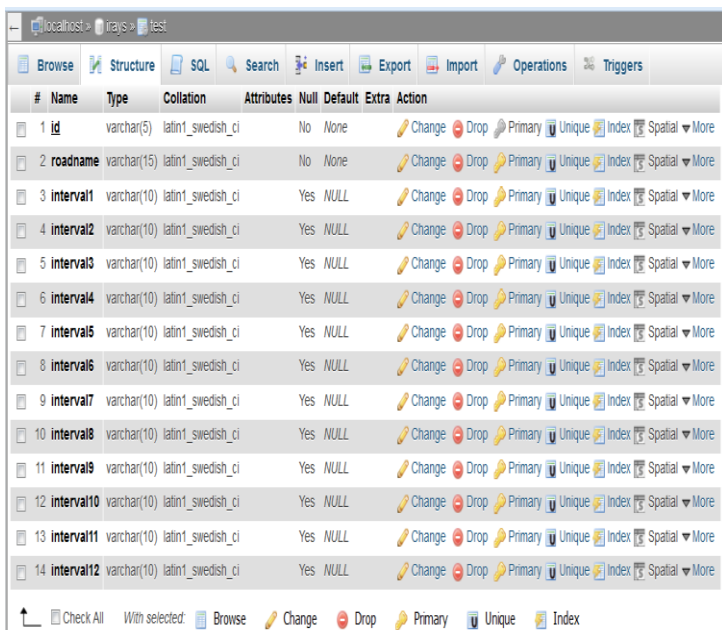
Figure 8

Trays database is created on the Wamp Server where two tables are created “main” and “test”. The time interval that we have considered for our project is between 6pm-6am. Four quarters of 3 hours each i.e 6-9,9-12,12-3,3-6 are named as quarter1, quarter2, quarter3 and quarter4 respectively that we have taken into consideration. For quarter 1, when camera sensor sends data to the raspberry pi, after every 15 minutes processed data is transmitted to the “test” table of irays database. Thus after completion of 1st quarter, we have 12 records in the test table. On the basis of these 12 records final density value for that quarter is calculated and stored in the “main” table.



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	varchar(5)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial More
2	roadname	varchar(15)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial More
3	date	date			No	None		Change Drop Primary Unique Index Spatial More
4	quarter1	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
5	quarter2	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
6	quarter3	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
7	quarter4	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More

Figure 9



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	varchar(5)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial More
2	roadname	varchar(15)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial More
3	interval1	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
4	interval2	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
5	interval3	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
6	interval4	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
7	interval5	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
8	interval6	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
9	interval7	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
10	interval8	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
11	interval9	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
12	interval10	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
13	interval11	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More
14	interval12	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial More

Figure 10

For further analytics process, data stored in the database can be used to improve performance of the system. Data can be retrieved from the database for a particular road for a particular day.



Traffic Management System

Date:

Road Name:

Figure 11

DETAILS OF THE ROAD :

ROAD ID : 101

ROAD NAME : JVPD

DATE : 2017-04-07

Quarter1(6PM-9PM):HIGH

Quarter2(9PM-12AM):MEDIUM

Quarter3(12AM-3AM):MEDIUM

Quarter4(3AM-6AM):LOW

Figure 12

6. CONCLUSIONS

This paper describes a new intelligent street lighting system which integrates new technologies to offer higher efficiency and considerable savings of electricity consumption[2]. This can be achieved using the highly efficient LED technology along with real time video processing[4] being carried out on the raspberry pi module to detect the vehicles on the road and intelligently adjust the intensity of the street lamp. In addition, control system manages switching of the light on the street lamps only when necessary, thereby increasing the lamps' lifetime. The proposed system is particularly suitable for street lighting in urban and rural areas where the traffic is low during the night time. The independent nature of the power-supply network enables implementing the system in remote areas where the classical installations are prohibitively expensive. The system is always flexible, extendable, and fully adaptable to user needs. It can be implemented on lamps in gardens and offices where huge electricity consumption takes place. The detection of various objects will help in automatically changing the proper lighting system.

REFERENCES

- [1] Fabio Leccese, "Remote-Control System of High Efficiency and Intelligent Street Lighting Using a ZigBee Network of Devices and Sensors". IEEE TRANSACTIONS ON POWER DELIVERY, VOL. 28, NO. 1, JANUARY 2013.
- [2] N Pradeep Kumar, , Dr. Ravi Kumar Jatoth, "Development of Cloud Based Light Intensity Monitoring System Using Raspberry Pi". 2015 International Conference on Industrial Instrumentation and Control (ICIC),Col/ege of Engineering Pune, India. May 28-30,2015.
- [3] Chulian Zhang, Hamed Tabkhi, Gunar Schirner, "A GPU-based Algorithm-specific Optimization for High-performance Background Subtraction"
- [4] V. Kastrinaki, M. Zervakis, K. Kalaitzakis, "A survey of Video processing techniques for traffic applications", Image and Vision Computing" 21(2003) 359 –381

- [5] Deepali Shinde, Mithilesh Said, PratiknShetty, Swapnil Gharat, "Optimizing real time GPU kernels using Fuzzy Inference System", IJARSE, Vol. No. 2, Issue No.9 , ISSN-2319-8354(E).
- [6] Raspberry pi community, "<http://www.raspberrypi.org/products/modelb-plus>"
- [7] http://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html
- [8] <https://sourceforge.net/p/raspberry-gpio-python/wiki/BasicUsage> .