ISSN 2278-3091
Volume 9 No.2, March - April 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse157922020.pdf
https://doi.org/10.30534/ijatcse/2020/157922020

Design and Development of IOT, Web-Server and ML-AVPR based Intelligent Humanoid Robot for Traffic Assistance

Kantilal P Rane
Professor, KCEs COEIT, Jalgaon

ABSTRACT

In India, there is lack of traffic assisting persons (traffic police). They are basically engaged in various activities by authorities. Major problem facing is while traffic is switched OFF because of technical problems like battery backup, under maintenance condition or many more. To overcome these problems, the intelligent traffic assisting robot can be used at the square of traffic signal. When power ON, it will act as intelligent monitoring and alerting system. When traffic signal goes OFF, it will take control of signal rotating with its basic task of intelligent monitoring and alerting. Humanoid robot is designed with low power devices that takes very less power than that of required for signal operation at the time of mains power OFF. By same propose system, both manual traffic like control (by robot) and signalling control may be performed at a time if required. Intelligence required in the proposed system is for identifying specific instance as traffic breaking. Various instances may be monitored using back end program designed. But here, traffic breaking is discussed in details. With traffic breaking instance, intelligent algorithms perform tasks as identifications of traffic breaking vehicle, driver and recognition of number plate for assigning fines to particular. It is performed using machine learning model called ML-AVPR Algorithm. A novel approach for image segmentation of vehicle area, driver area and licence plate area is proposed for better analysis through Machine Learning algorithm. The segmented images are given as input to the Machine learning algorithm that identifies various required parameters in real time manner. The results are satisfactory and appeared in real time sense.

Key words: Intelligence Traffic Sensing, Humanoid Robot, Machine Learning, Web-Server.

1. INTRODUCTION

Many traffic controlling, monitoring systems were developed by many researchers [1]. According to the survey in USA, traffic signal breaking between 1997 and 2004 was about 51% of total traffic [2] so need is to control it by the means of intelligence system. In 2015, the words first digital intelligent road was developed in Germany that used road side units and vehicle side unit for inter communication of specific information about traffic and many more [3] that leads reducing number of accidents, traffic congestions and carbon emotions etc.

During 2016-2017, Autonomous vehicles were designed based on the intelligent transportation system. This vehicle autonomously followed all the traffic rules [4]. Adaptive timing controlled traffic signals based on the volume of traffic along the specific way was developed to handle the demand of real time traffic control and to handle the traffic congestion in cities and metro cities [5]. Mostly the timing for signal rotation is predefined with proper sequence. It becomes inefficient in the case of odd density of traffic, this leads to increasing in waiting time and signal jams [6]. Similar systems were developed before but not able to improve the behaviour of driver [7]. Similar work latter also reported the varying behaviour of drivers [8], but because of worst situation of traffic around all the metro cities, it was suggested to have intelligent solutions. Same system was then modified for the inclusion of more information about vehicle surrounding into the in vehicle unit [9]. Later on, in vehicle unit's simulators called Simulation of Unity 3D [10], Urban Mobility (SUMO) [11], and City Engine [12] were developed to analyze the traffic of city. All these intelligent systems are useful to metro cities but not feasible to towns and big villages as it may be useful for costlier vehicles. With smart city developments, it is needed to think about growing smart towns and smart villages. So, specific systems applicable to cost effective vehicles is needed to be developed. Intelligent observing eye is required to be developed for real time monitoring, real time analysis and real time alerting.

A combined traffic controlling, monitoring, alerting and real time performing system is needed to be developed yet that are compatible for various instances like signal breaking, vehicle tracking with less information (only by colour of vehicle or only by name or only by number etc).

Intelligent traffic control is a today's need for all smart cities, growing cities and even small towns for monitoring to proper security. Security on square of traffic signal is more important as it may be generally inlet and outlet to most of the cities and towns. So, real time monitoring and alerting to the particular security person is needed with exact intimation of vehicle. This information may be about traffic signal

breaking, tracking of vehicles having little information like colour and name of vehicle, information matching like colour and name of vehicle through Licence plate number (if vehicle's secures data is available). There are many issues in collecting the information. Out of these issues, major and frequent issue to be handled are traffic signal breaking. So, we designed a system that provides control on traffic signal breaking by using intelligent system. Humanoid robot based intelligent system is designed that may act as manual traffic hand gesturing model for traffic control. At the same time it senses the signal breaking vehicles with the help of four cameras facing towards four ways around the square of traffic signal. Heighted cameras are facing problems of angular invisibility so the height of cameras is set along the height of average height of all vehicles. Proposed intelligent system will be providing the various parameters needed to recognize the vehicle type, vehicle name, vehicle colour, vehicle driver, and vehicle plate number. The whole software system is divided into two parts as front end algorithm and back end algorithms. By simply modifying front end algorithms, other instances may be recorded (at the place of instance of traffic breaking) with recognized parameters that are useful for security purposes. Back end algorithms compatible for all these instances are designed and will be same for all applications and instances. A case of instance for traffic signal breaking is discussed in this paper in details.

Propose system consists of Humanoid robot having capability of gesturing motions for traffic control. It was configured with Arduino controller with various electronics modules like servo driver module, MP3 player sound module useful for alerting to traffic. Six servo motors are used for the operation of basic motion of humanoid robot.

Raspberry Pi controller is used for performing intelligent operation to the system to which four cameras are attached. Instances are alerted by robot module to raspberry Pi controller to recognize the parameters to be extracted from the videos captured by traffic breaking. Raspberry Pi is configured for sensing traffic breaking, recording video, recognizing parameters like vehicle number, colour, name, type, driver face etc. It uses machine learning algorithms for extracting these parameters. Extracted parameters are displayed on web page. And to provide access to web page to security person, embedded web-server is installed in to Raspberry Pi controller. Authentication is provided for web page access. Alerting to the particular security person is possible by installing online SMS module in to the Raspberry Pi controller. Alerted person will then take care about further action to be taken for traffic breaking.

2. METHODOLOGY

Proposed system consists of 2 basic modules as 1) IOT based Gesturing Model for Traffic Controlling Humanoid Robot and 2) Vision Extracting and ML classifying/recognizing Raspberry Pi module. Electronic designs and algorithmic designs are discussed in following sub-sections.

2.1. IOT based Gesturing Model for Traffic Controlling Humanoid Robot

Humanoid Robot model (about 5 feet) is designed as IOT based Gesturing Model for Traffic Controlling with specific motion's modelling like hands motions, neck motion and speaking announcement support with mannequin humanoid body structure. Arduino Uno based system is designed for controlling various actions of robot model. Arduino is interfaced with six servo motors (servo number 1 to servo number 6) with high torque (MG995 Metal Gear Servo) for controlling hands and neck motion through 16 bit servo driver (16-Channel 12-bit PWM/Servo Driver). Pin tilt mechanisms are used for actuation of the motions with specific angels. All the motions are mapped with 0 to 180 degree for full angular motion of servo motor. For rotating whole humanoid robot around for action as traffic assistant at traffic square, 360 degree rotating servo (servo number 7) is configured with high torque. For announcing the specific and predefined instructions, the loudspeaker (with amplifier) can be interface. The mp3 Sound Player Module (Audio MP3 Player module WTV020-SD-16P) is used for storing the predefined voice files (may be in robot voice) that can be controlled using the Arduino for selection particular mp3 file for the purpose of playing on loudspeaker through amplifier. 5v, 3.5A power supply is used for the driver of steeper motor. 5v, 1A power supply is required for the operation of Arduino Uno. And separate power supply (as per the specification of loudspeaker) is required for interface to loudspeaker. All the pan tilt mechanisms are assembled on the specific position on the mannequins like at hand joints and neck joints and circuit is assembled on the back side of it. Each pan tilt mechanism are included with two servo motors for circular and horizontal motions like for neck motion, one servo motor is used for circular motion and 2nd servo motor is used for up and down motion and for every hand motion, one servo motor is used for circular motion and 2nd servo motor is used for stretching hand away of the body structure. Interfacing of Arduino Uno with servo motors is through the 16 bit servo driver and to MP3 player module (called 'IOT based Gesturing Model for Traffic Controlling Humanoid Robot') is shown in following Fig 1. Specific program is designed for embedding into the Arduino for the specific action of the humanoid robot.

2.2. Vision Extracting and ML classifying/recognizing Raspberry Pi module

Four cameras facing towards four ways are interfaced to the interfaced to Raspberry Pi controller for identification of traffic breaking vehicle number and driver image. Arduino and Raspberry Pi controller are in communication to each other and Arduino is programmed for sending the status of

current traffic to Raspberry Pi through serial port. According to the information received from Arduino, in case of traffic signal breaking, video of vehicle is captured and information related to vehicle name/type and vehicle colour is extracted using ML algorithm. ML is also used for vehicle number plate recognition. Raspberry Pi controller is also used for extraction of driver face for further action.

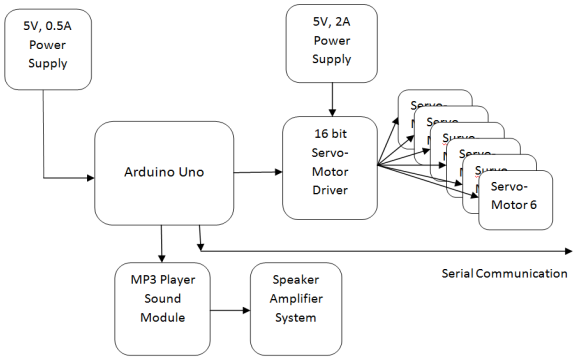

Figure 1: Block Diagram for IOT based Gesturing Model for Traffic Controlling Humanoid Robot

Interfacing of various cameras to Raspberry Pi (called Vision Extracting and ML classifying/recognizing Raspberry Pi module) for intelligent identification is shown in following Fig. 2.

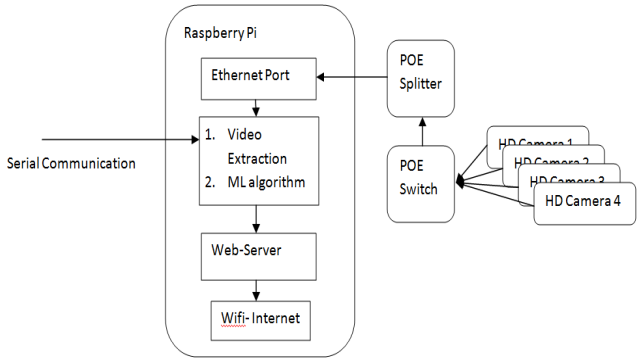

Figure 2: Block Diagram for Vision Extracting and ML classifying/recognizing Raspberry Pi module

Algorithmic steps (Arduino embedded Program), designed for 'IOT based Gesturing Humanoid Robot Model for Traffic Controlling' are specified as bellow. It may be considered front end algorithm.

1) When electric power breaks, robot system takes control and initialises all the required libraries.
2) Initialize all the initial processes like setting all servo motors at initial positions. Setting initial data required for initialization of mp3 player module. Play the initial sound to be played at the time of power ON condition. Here we played Hindi announcement as 'Hallow, Hallow, sub suno, suno, suno, robot pulis ke instruction se traffic control hogi, Hallow, Hallow, sub suno, suno, suno, robot pulis ke instruction se traffic control hogi, Hallow, Hallow, sub suno, suno, suno, robot pulis ke instruction se traffic control hogi.' Then if any action is required at the time of ON condition perform that on servo motors. Here rotates around and show action to all ways for stopping the traffic.
3) Perform loop operation as starting from step 4 to step 7.
4) Direct towards the initial position facing body/rotate towards the way1 of traffic square by rotating motor7 (rotation 90 degree in each tern). And rotating face towards the way2. Announce to traffic by long vesicle sound. Sends the current traffic way id to raspberry Pi through serial link. Direct the traffic of way2 to go by right hand up and having right to left motion for 20 times with delay of 5 sec in each action (total 100 sec.). And direct to other ways to stop by stretching left hand away horizontally.
5) Rotates around it in clockwise towards body facing towards way2 and face facing towards the way3 (no need to rotate face in each rotation of body). Announce to traffic by long vesicle sound, sends the current traffic way id to raspberry Pi through serial link and stops way2 traffic and allows way3 traffic to go for 100 sec.
6) Rotates around it in clockwise towards body facing towards way3 and face facing towards the way4. Announce to traffic by long vesicle sound, sends the current traffic way id to raspberry Pi through serial link and stops way2 traffic and allows way4 traffic to go for 100 sec.
7) Rotates around it in clockwise towards body facing towards way4 and face facing towards the way1. Announce to traffic by long vesicle sound, sends the current traffic way id to raspberry Pi through serial link and stops way4 traffic and allows way1 traffic to go for 100 sec.
8) Exit lop when electric power ON.

Four IP HD cameras are interfaced to Raspberry Pi through Ethernet port [13]. We can capture the images or videos for specified duration by using the specific programming [14]. Out of the 4 cameras facing towards the all the 4 ways, 3 cameras are activated at a time and one camera is deactivated.

Algorithmic steps (Raspberry Pi Embedded Program) for vision extraction and ML classification/recognition are specified as bellow. It is considered as back end algorithm.

1) Scan the serial port for any data from Arduino serial port.

2) If serial data contains the number between 1 and 4 then valid number is received. In invalid number is received then system discards.
3) If valid number is received say 2.
4) Camera facing way2 is deactivated. Cameras facing way3, way4 and way1 are activated.
5) If any vehicle is identified in moving condition from any one or any two or all ways from way3, way4 and way1, corresponding videos are captured as a video of signal breaking vehicle.
6) Various frames are extracted from captured videos.
7) And from various frames, its number plate area and driver area are cropped out.
8) Best driver face area and best number plate area from similar faces and similar plate images are identified respectively by using Novel approach of Video Extraction Method. It extracts the required best face area and best number plate area by comparing boundary extracted from boundary detection algorithm [15]. Proper boundary shape matching [16] is used for identification of best face image and best vehicle number plate area image. Best vehicle image is also selected for identification of vehicle type from various frames by similar techniques.
9) Indentified best face image, best vehicle number image and best vehicle image (identified from various frames of video) are sent to Web-server for storage and for further processing/recognition.
10) The best vehicle image is used to identify the vehicle type. This algorithm uses machine learning (ML) algorithm say 'ML-AVPR Algorithm' for vehicle image classification using ML with ALPR on-Premise SDK [17] for Raspberry Pi controller and displays the colour, name/type of vehicle. It is also used to recognize the number using Licence Number Plate Recognition.
11) The identified best vehicle image with identified parameters, best driver face image with corresponding plate image and recognized vehicle number are displayed in web-page designed that may be accessed by security authority (or Police station authority) for further processing or further fine collection.
12) Provide access to web-page by using secure authentication process with specific username and password.
13) Repeat the process from step 1) to step 13) for next signal breaking vehicle.

2.3. Serial communication between Arduino and Raspberry Pi

In proposed system, data related to the current Go way has to be sent to Raspberry Pi through serial communication (Fig 3). For this serial communication between Arduino and Raspberry Pi, it is required to connect Arduino USB Plug to Raspberry Pi through USB cable and needed to check the connection between Arduino and Raspberry Pi by specific commands.

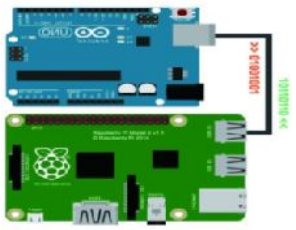

Figure 3: Serial communication between Raspberry Pi and Arduino

2.4. IP cam interface to Raspberry Pi

For IP cameras are interfaced to the proposed system through Raspberry Pi controller to capture the video of signal breaking vehicle. Camera system requires a Raspberry Pi controller, a Micro SD card, a Low-profile USB Flash Drive at least 8GB, a POE Splitter, a POE Switch/wifi router and four HD IP cameras. For proposed system, Raspberry Pi3 controller is used that can have all capability of camera access and video processing. USB flash drive 8GB SanDisk Cruzer Fit CZ33 is used for storage the excess captured data, A TP-LINK TL-POE10R POE Splitter is used as a compact device. A POE switch/router is needed to power the camera since we have to use four IP cameras (Fig. 4).

Raspberry Pi controller is needed to assign a static IP for the communication to rest of the network. And all the cameras are also needed to assign the Static IPs with same subnet mask and gateway. As we selected IP cameras who's IPs can be configure with the embedded web pages inside its own controller. So, initially IPs are individually set. It is also required to add the name server to accessing from outside world.

To setup the video steaming, motion model is installed with its supporting libraries. It is also needed to configure the motion files as log file and motion data file. Interval for taking snapshot from each camera is set as 01 second so that after every second, snapshot is captured and stored for further processing. System is configured so that used image data is deleted after 01 day so that to free the memory for further captures. Firewall is configured to access the defined port for video stream. Now as per the specified instances (signal breaking) from NOT GO ways along traffic signal, system can access the data from corresponding cameras. And data is further processed using Video Extraction Algorithm and ML-AVPR Machine Learning Algorithm for getting extracted information.

Figure 4: IP Cameras Interfaces to Raspberry Pi through POE Splitter and POE switch

2.5. Video Extraction Algorithm

The required information from captured video is 1) the image of vehicle (vehicle that breaking the traffic signal) 2) image of its driver face area and 3) image of licence number plate area of that vehicle.

Video capture from four attached camera are also configured to sense only the incoming vehicles towards the camera system but not leaving from system so that it includes all the required views like face of driver, number plate and vehicle image.

Object identification using boundary shape contest algorithm [16] is used to identify the shape of possible vehicles like bikes, cars, buses and trucks etc. The boundary area is extracted from this algorithm. This boundary area is cropped to get the rectangle shaped image of vehicle. Out of the cropped image, the rectangular shape within specific limit is identified as a Licence Number Plate Area using same algorithm [16].

It was recommended to use four HD cameras with automatic light focussing system for night visualization and proper image capture. As use of HD cameras are proposed for the system, the area of driver position can be visible even in motion. The front glass area is identified by using Object identification using boundary shape contest algorithm [16] and cropped to get screen image through which the left most driver position of relative size is cropped as a driver image. It also captures the image of driver from side glass area for more security or if it is not captured from front glass.

Three extracted images as image of vehicle, Licence Plate Area Image and Driver image are stored and also given as the input to the ML-AVPR Machine Learning algorithm to identify the detail information like type/ name of vehicle and the number of vehicle.

2.6. ML-AVPR Machine Learning algorithm

Raspberry Pi based ML-AVPR Algorithm Machine Learning algorithm are designed using ALPR on-Premise SDK [17] for mainly two operations as 1) colour, name/type of vehicle identification and 2) number plate recognition.

ML-AVPR Algorithm: ALPR on-Premise SDK [17] is installed on Raspberry Pi controller to analyze the images of vehicles and to respond with decoded licence plates. It requires internet for installation of SDK but at the time of use it doesn't required internet access. It needs to subscribe the SDK plan. It needs to sign-up [18] and sign in [19] using SDK. Docker is needed to install on Raspberry Pi. Steps to process using SDK are 1) Get SDK image 2) create the container to use the host's network directly. It takes 10 to 20 sec for initialization and 3) Use specific run command 'platerecognizer/alpr-raspberry-pi' in the new terminal to get the result image with various parameters.
Raspberry Pi3 is used with 500MB of free Ram and is configured to run the container automatically on system start-up using 'Start up on Boot' option [20].

2.7. Web-Server

Web-servers are designed using .net platform [21]. It is installed on Raspberry Pi for the storage of web pages and required data. It is used to store the extracted information from ML-AVPR Machine Learning Algorithm. It is also stored with the specific static web page that contains the traffic breaking instances with driver's extracted face, vehicle image, vehicle type/name, vehicle licence plate number for further accessing by security authority. Web authentication is provided by using Username and password option.

2.8. Web Page

The Webpage with information to be accessed are stored in SD card that is interfaced to Raspberry Pi controller. Raspberry Pi is configured with web-server Header files and Accessing Protocol [22] for proposed web-server. Raspberry Pi IP address is needed to port forward in order to view webpage from external network by competent authority. Web accessing authentication is provided by designing web page using password based Sign-up and Sign-in process.

We may use other servers like for proper webpage handling like servers in [23] and [24] and machine learning algorithm [25].

3. RESULTS

Experimental Setup for IOT based Gesturing Model for Traffic Controlling Humanoid Robot is shown in Fig. 5.

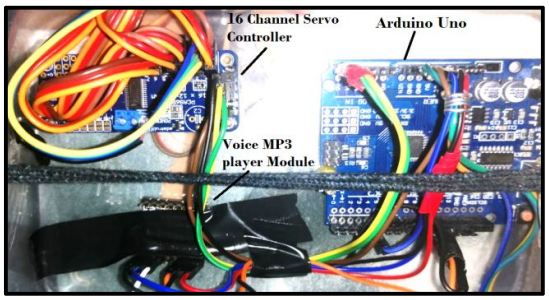

Figure 5: Experimental Setup for IOT based Gesturing Model

Experimental Setup for Vision Extracting and ML classifying/recognizing Raspberry Pi Module is shown in Fig. 6.

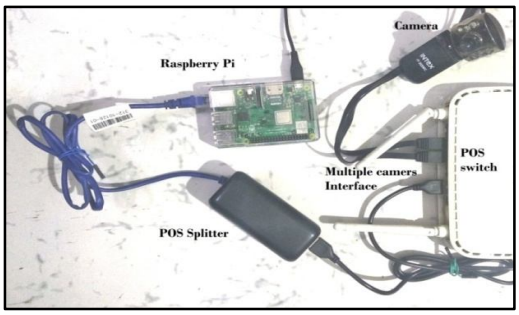

Figure 6: Vision Extracting and ML classifying/recognizing Raspberry Pi Module

Following figure shows the real view Girl Robot Model (Humanoid Robot) used for traffic control which is rotating along itself by fixed interval and hand gestures for traffic control.

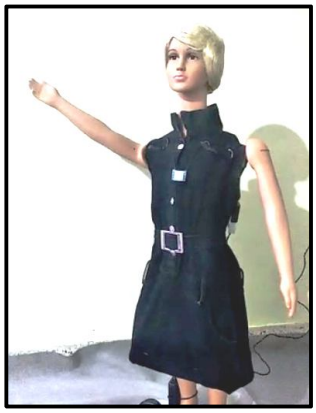

Figure 7: Humanoid Robot Model as traffic Assisting Robot

Fig 8.a-Fig 8.c shows the results of vehicles at different angles, different positions and different sizes with their corresponding obtained parameters like plate number, colour, type/orientation, approximate year of manufacturing and processing time using proposed algorithms

.

Figure 8.a: Result of ML-AVPR Algorithm

Figure 8.b: Result of ML-AVPR Algorithm

Figure 8.c: Result of ML-AVPR Algorithm

Driver face extraction is performed using Video Extraction Algorithm and the extracted vehicular view and extracted faces at various positions are visualized in Fig. 9 and Fig. 10.

Figure 9: Identified Vehicle with Driver View

Figure 10: Identified Driver at Different Views

Web Page is protected by username and password and the authentication page for accessing website is shown in following figure.

Figure 11: Webpage for Sign-up and Sign-in

After proper authentication, webpage displays the extracted information from the captured video of specific instance (traffic signal breaking). The various instances are visualized in the webpage as shown in Fig. 12.

Figure 12: Webpage having Captured Instances

4. CONCLUSION

Day by day traffic is increasing and related crimes are also increased. To control the crimes, intelligent eyes are the today's need. Proposed system is best fit for the high traffic areas and can be installed at traffic signal square. A novel approach of extraction the information from captured video using shape contest object matching is well performed with in the specified application. Pre-trained ML-AVPR Machine Learning algorithm is successfully designed and installed on Raspberry Pi for identification of vehicle related parameters that are useful for traffic control, monitoring and alerting purposes. Alerting to the particular security authority is also tested using installation of online SMS pack on Pi controller. System can be easily modified for different instances or for different applications by modifying the IOT based gesturing module only that may required less efforts. Web pages are accessible to the authority using security password for further action.

REFERENCES

[1] Subramaniyam, Siva, Karthikeyan, Sivaraman, Srinivasan, Shreyas Ramachandran, A.K, Veeraraghavan, **A Survey on IoT Based Intelligent Road Traffic and Transport Management Systems**, 10.15680/IJIRCCE.2017. 05012005, 2017.

[2] R. Subramanian, L. Lombardo, **Analysis of fatal motor vehicle traffic crashes and fatalities at intersections**, 1997 to 2004, Tech. Rep, 2007.

[3] H. Breuer, **Urban mobility: radars key role on smart roads**, 2017, https://www.siemens.com/innovation/en/home/pictures-of-the-future/mobility-and-motors/urban-mobility-radar-technology-for-highways.html.

[4] C. Olaverri-Monreal, **Autonomous vehicles and smart mobility related technologies**, Info-communications journal, vol. 8, no. 2, pp. 17–24, 2016.

[5] V. Astarita, V. P. Giofrè, G. Guido, and A. Vitale, **The use of adaptive traffic signal systems based on floating car data**, Wireless Communications and Mobile Computing, vol. 2017, Article ID 4617451, 13 pages, 2017. https://doi.org/10.1155/2017/4617451

[6] USDOT, **Traffic signal timing manual**, 2017, https://ops.fhwa.dot.gov/publications/fhwahop08024/chapter5.htm.

[7] C. Sommer, R. German, and F. Dressler, **Bi-directionally coupled network and road traffic simulation for improved IVC analysis,** IEEE Transactions on Mobile Computing, vol. 10, no. 1, pp. 3–15, 2011. https://doi.org/10.1109/TMC.2010.133

[8] M. Krause, K. Bengler, **Traffic light assistant evaluation of information presentation,** Advances in Human Aspects of Road and Rail Transportation, PP. 166, 2012.

[9] B. Bernais, A. Lotz, and H. Pu, **Design and implementation of a traffic light assistance system based on c2x and tsi messages**, in Proceedings of the AmE 2016-Automotive meets Electronics; 7th GMM-Symposium, PP. 1–6, 2016.

[10] Unity Technologies, **Unity - fast facts**, 2016, https://unity3d.com/public-relations.

[11] SUMO, **DLR - Institute of Transportation Systems - SUMO – Simulation of Urban Mobility**, 2016.

[12] Esri, **Esri cityengine - 3d modelling software for urban environments**, 2017, http://www.esri.com/software/cityengine.

[13] Shete, Ruturaj, Sabale Mayuri, **Video Surveillance using Raspberry Pi Architecture**, 2015.

[14] Sneha Singh, Pradnya Anap, Yogesh Bhaigade, Prof. J. P. Chavan, **IP Camera Video Surveillance using Raspberry Pi**, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 2, February 2015. https://doi.org/10.17148/IJARCCE.2015.4272

[15] Wei-Ying Ma, B. S. Manjunath, **EdgeFlow: a technique for boundary detection and image segmentation**, IEEE Transactions on Image Processing, Vol. 9, No. 8, August 2000. https://doi.org/10.1109/83.855433

[16] Belongie, Serge & Malik, Jitendra, Puzicha, **Shape Matching and Object Recognition Using Shape Contexts**, IEEE Transactions on Pattern Analysis and Machine Intelligence, Jan. 2002.

[17] http://doc.openalpr.com/cloud_api.html#on-premises-sdk

[18] http://app.platerecognizer.com/accounts/signup/

[19] http://app.platerecognizer.com/accounts/login/?next=sdk/

[20] http://app.platerecognizer.com/sdk/#boot

[21] Mohammad Ziya Ul Haq, **Embedded Web server for Industrial Applications using RaspberryPi**, International Journal of Engineering Science and Computing, PP: 6371-6394, April 2017.

[22] http://www.instructable.com/id/ServDuino-Arduino-Webserver/

[23] R.Srilakshmi, Dr.M.Jaya Bhaskar, **Dual Server based Security Protocol in MANET using Elliptic Curve Cryptography: A Cluster Head Selection Scenario**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 8. No. 4, pp- 1621 – 1629, 2019. https://doi.org/10.30534/ijatcse/2019/87842019

[24] Devi Afriyantari Puspa Putri, Aulia Rachmawati, **Honeypot Cowrie Implementation to Protect SSH Protocol in Ubuntu Server with Visualisation Using**

Kippo-Graph, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 8. No. 6, pp- 3200 – 3207, 2019. https://doi.org/10.30534/ijatcse/2019/86862019

[25] Dr. D. K. Kirange, Dr. J.P. Chaudhari, Dr. K. P. Rane, Dr. K. S. Bhagat, Dr. Nandini Chaudhri, **Diabetic Retinopathy Detection and Grading Using Machine Learning**, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 8(6), PP. 3570-3576, 2019. https://doi.org/10.30534/ijatcse/2019/139862019