Smart Traffic Light System Using Internet of Things

Vijay Gaikwad, Apoorva Holkar, Tejas Hande, Priyanka Lokhande, Vivek Badade

Vishwakarma Institute of Technology, Pune, India

Corresponding author: Tejas Hande, Email: tejas.hande22@vit.edu

The rapid urbanization and increasing traffic density in metropolitan areas have led to several road safety and traffic management challenges. One of the key factors contributing to these issues is the inefficiency of traditional traffic light systems, which operate based on pre-determined schedules and fail to adapt to changing traffic conditions. In this research paper, we pro-pose a smart traffic light system that leverages advanced technologies such as Internet of Things (IoT), and machine learning to optimize traffic flow and enhance road safety. The proposed system incorporates real-time traffic monitoring, predictive analytics, and adaptive signal control algorithms to dynamically adjust traffic light timings based on the current traffic situation. We evaluate the performance of the proposed system using simulations and real-world experiments, demonstrating significant improvements in traffic flow efficiency and reduction in travel time and traffic congestion. The re-sults of this research demonstrate the potential of smart traffic light systems to revolutionize traffic management and improve road safety in urban areas.

Keywords: Raspberry pi, Smart traffic light, Open CV, Internet of things, Priority Vehicle detection, ESP32 CAM

2023. In Satyasai Jagannath Nanda & Rajendra Prasad Yadav (eds.), *Data Science and Intelligent Computing Techniques*, 795–808. Computing & Intelligent Systems, SCRS, India. https://doi.org/10.56155/978-81-955020-2-8-68

1 Introduction

The increasing population and urbanization have resulted in the rapid growth of vehicles on the roads, leading to an increase in traffic congestion, accidents, and pollution. Traffic congestion not only results in the wastage of time but also causes economic losses. Inefficient traffic management systems are one of the primary contributors to these problems. Traditional traffic light systems operate based on pre-determined schedules and do not consider the real-time traffic flow, leading to delays, congestion, and accidents. Therefore, there is a need for a smarter and more efficient traffic management system that can adapt to changing traffic conditions and optimize traffic flow.

In recent years, advanced technologies such as Internet of Things (IoT), and ma-chine learning have shown tremendous potential in transforming traffic management systems. Smart traffic light systems leverage these technologies to enable real-time traffic monitoring, predictive analytics, and adaptive signal control algorithms to optimize traffic flow and enhance road safety. These systems can also reduce travel time and improve fuel efficiency, leading to a reduction in carbon emissions and air pollution.

The objective of this research paper is to present a smart traffic light system de-signed to improve traffic flow and enhance road safety in metropolitan areas. The proposed system leverages cutting-edge technologies like IoT, and machine learning to monitor and analyze real-time traffic data, enabling adaptive adjustments to traffic light timings. The effectiveness of the system will be assessed through simulations and real-world experiments. The outcomes of this study are anticipated to advance the development of intelligent and safer traffic management systems, addressing the growing traffic challenges experienced in metropolitan areas.

At the heart of this innovative traffic management system is the ESP32-CAM module, a compact and budget-friendly camera module endowed with integrated Wi-Fi and Bluetooth functionalities. Leveraging the capabilities of this module, the system facilitates real-time monitoring of urban traffic scenarios through live video streaming. This pivotal feature offers city planners and authorities immediate access to visual insights into traffic patterns, congestion, and road infra-structure.

To further augment the efficiency of the traffic management system, it seamless-ly integrates the OpenCV library. By harnessing the power of computer vision algorithms, the system achieves real-time object detection and vehicle classification. This empowers the system to accurately identify and differentiate between various vehicles and activities on the road network. By using OpenCV, the system significantly reduces false alarms by distinguishing potential traffic disruptions, such as accidents or stalled vehicles, from routine traffic flow.

2 Literature Survey

In the research paper by Yusuf et al., [1]the study focuses on the relationship between elevated traffic volume and congestion. One of the primary contributing factors identified is the traffic itself. The paper highlights the necessity for a mechanism that enables traffic lights to dynamically and seamlessly adjust signal timing based on the current traffic conditions. This smart traffic management system aims to prioritize the smooth flow of emergency vehicles and enhance safety for pedestrians.

Kanungo et al., [2] proposed the method of using video monitoring systems for smart traffic lights. It focuses on using live video feed from traffic cameras for real-time traffic density calculation and optimizing traffic light control based on vehicle density.

Pavani et al. [3] proposed an IoT-based transportation management system for alleviating traffic congestion in Muscat, Oman. The method calculates real-time traffic density through film and picture analysis. Utilizing Raspberry Pi-based signal lights with density-based technology via a mobile application aims to reduce congestion and improve traffic control. The study underscores the link between gridlock, slower speeds, extended wait times, and accidents, emphasizing the need for continuous surveillance for network growth. Technological improvements addressing traffic light timing limitations pave the way for more intelligent transportation systems.

"Smart Traffic Light Management Systems: Systematic Literature review" presented by Magableh et al., [4] addresses congestion complexities by adapting to var- ying traffic flows, handling interference, and accommodating emergencies and pedestrians. IR sensors assess traffic density and adjust timing slots, while a handheld device facilitates wireless communication for emergency vehicle movement. The system improves traffic management, reduces congestion, enhances safety, and optimizes travel efficiency.

Chavan et al., proposed [5] an intelligent traffic light controller (ITLC) to over-come limitations of conventional controllers. The ITLC utilizes sensor networks and embedded technology to dynamically adjust timings based on adjacent road traffic, optimizing flow, and preventing congestion. It incorporates a GSM inter- face for real-time traffic information and alternate route selection. The ITLC outperforms fixed mode controllers, reducing waiting times, increasing vehicle distance traveled, and enabling efficient emergency mode operation. It has a simple architecture, fast response time, and potential for future expansion.

Oliveira, Manera, and Luz [6] tackle congestion and pollution with an IoT-based smart city traffic management system. They design a traffic light controller circuit with centralized control and wireless communication. Practical tests demonstrate its effectiveness, showcasing potential for smart city applications. Emphasizing the significance of smart traffic lights in congestion management and IoT integration, especially with the Internet of Vehicles (IoV), the proposed circuit aims to improve current traffic control systems for smart city development.

Razavi et al., [7]address urban traffic challenges with an innovative intelligent traffic management approach, using IoT and smart city tech. Their dynamic traffic light control system adjusts signal timings based on real-time vehicle density. Implemented with a Raspberry Pi and OpenCV, the solution shows promise for scalable, efficient urban mobility. The proposed system can be enhanced by integrating with existing smart city infrastructures, refining for diverse traffic scenarios

Lee and Chiu[8] discussed the design and implementation of a smart traffic signal control (STSC) system within a larger vehicular network infrastructure in smart cities. The STSC system integrates intelligent transportation systems (ITS) applications, such as emergency vehicle signal preemption, public transport signal priority, adaptive traffic signal control, eco-driving support, and message broadcasting. The system aims to reduce traffic congestion and improve public transport efficiency.

A simulator developed by Wiering et al., [9]called Green Light District (GLD) for experimenting with different infrastructures and comparing traffic light con-trollers. They proposed a model-based, multi-agent reinforcement learning algorithm that uses road-user-based value functions to determine optimal decisions for each traffic light. Cars at a junction vote based on their estimated advantage or gain from setting their light to green, determined by the expected difference in waiting time. Waiting time is estimated by monitoring car flow and utilizing reinforcement learning algorithms.

Bani Younes and Boukerche[10] proposed an intelligent traffic light controlling (ITLC) algorithm that efficiently schedules phases for isolated traffic lights. It considers real-time traffic characteristics at intersections. They also developed an arterial traffic light (ATL) controlling algorithm, where intelligent traffic lights coordinate to generate an efficient schedule for the entire road network.

Janardhana Rao et al., [11] propose an IoT and AI-based study addressing urban traffic management. The design includes roadside messaging devices for real-time traffic reports. Experiments show minimal error in road occupancy prediction and high accuracy in vehicle recognition. The prototype demonstrates feasibility for adaptive traffic signals and real-time updates, offering viable options for future urban traffic management. The project, funded by Omani, centers on Real-Time Feedback for Adaptive Traffic Signals.

Yousef et al., Shatnawi[12] proposed an innovative approach to traffic management using wireless sensor networks (WSNs) and intelligent traffic control algorithms. Their system addresses existing limitations in traffic surveillance and control, providing a cost-effective solution to alleviate congestion. By utilizing WSNs, the system enables dynamic and adaptive traffic control, optimizing flow based on real-time conditions. Simulation results demonstrate reduced waiting time and queue length at intersections. The research includes a practical test bed deployment, contributing to traffic control technology advancements.

Miz and Hahanov[13] propose integrating smart traffic lights to optimize traffic management, reduce human errors, and enhance road transportation efficiency. The IoT- based CTMS enables a new generation information infrastructure for effective traffic control through big data and probabilistic analysis. Resolving challenges in internet coverage, standardization, and data analytics is crucial for maximizing the potential of IoT-based e-government systems. The CTMS infrastructure involves machine-to- machine communication and technologies provided by companies like WorldSensing in the Smart Cities domain.

Singh et al., [14] propose a comprehensive solution for traffic management challenges in Chandigarh, India, utilizing IoT and OpenCV. The system, implement-ed on a cost-effective Raspberry Pi, estimates vehicle counts near traffic jam points and conveys real-time data to a controlling station via an internet-based database. Experimental findings show efficacy in providing insights for traffic control, suggesting future applications such as city-wide implementation, integration with smart city infrastructure, machine learning enhancements, predictive analytics, and user-friendly interfaces for authorities. The research positions itself as a pragmatic and scalable solution for current traffic challenges and future smart city advancements.

Bali et al., [15] propose a "Smart Traffic Management System using IoT Enabled Technology" to tackle traffic congestion in smart cities. Their innovative solution introduces IoT-enabled "Green Corridors" for emergency vehicles, enhancing response times and efficiency. The research underscores the potential of IoT technology in creating efficient and life-saving traffic management solutions for urban areas

Sharif et al., [16] advocate for Smart Traffic Systems (STS) in future smart cities, proposing a cost-effective approach with low-cost vehicle-detecting sensors every 500 meters. Utilizing IoT for data collection and real-time updates, the method proves efficient and affordable for public traffic management. Future plans involve widespread implementation, integration with predictive analytics for proactive traffic management, and potential extensions of IoT applications in broader smart city infrastructure. This paper underscores the significance of the proposed low-cost STS for real-time traffic updates, contributing to more efficient and affordable solutions for future smart cities.

“Smart traffic management system using Internet of Things” by Javaid et al., [17] proposes a smart traffic management system for metropolitan areas, utilizing IoT and a decentralized approach to address urban mobility challenges. Overcoming limitations of previous systems, it employs cameras and sensors for traffic density data, predicting congestion with an algorithm. RFID prioritizes emergency vehicles, and fire sensors detect emergencies, enhancing response times. Validated in a Pakistani city, the system integrates a mobile application and a web platform for real-time information and future road planning. The findings showcase improved traffic optimization and emergency management. Future applications include widespread deployment, algorithm refinement, and further integration with emergency services for holistic urban development.

Rizwan et al., [18] propose a Smart Traffic Management System (STMS) for smart cities, identifying flaws and offering a cost-effective real-time solution. Low-cost vehicle sensors integrate with IoT for swift data acquisition. Real-time streaming data undergoes Big Data analytics, enhancing predictive traffic management. A user-friendly mobile app provides real-time insights and alternative routes, showcasing cost-efficiency and real-time updates. Future prospects include widespread implementation, refined predictive models, mobile app enhancements, and integration with broader smart city features, promising scalable solutions for urban development.

"Traffic management system using IOT" by Sahu et al., [19] presents a smart traffic management system utilizing eight sensors distributed over four lanes and the Internet of Things (IoT), the system that optimizes traffic signals to re-duce congestion. In situations of traffic congestion, emergency vehicles can be prioritized thanks to manual Wi-Fi control. The results are expected to result in less fuel being used, less air pollution, and improved traffic flow. The prototype integrates a mobile application for real-time monitoring and control, guarantee-ing a comprehensive approach to smart traffic management, and it also demonstrates cost-effectiveness, opening the door for large-scale implementation.

Hossain and Shabnam [20] address global traffic congestion, proposing an IoT-based smart traffic control system. It monitors and regulates traffic conditions, aiming to reduce pollution, resource waste, and idle time. Results highlight IoT's viability for traffic management and advance knowledge of various IoT-based strategies. Future plans involve ongoing research and deployment of reliable smart traffic control technologies.

Siripatana et al., [21] proposed an intelligent traffic signal system using image processing to dynamically adjust waiting times in real-time. Developed with Lazarus and OpenGL, the system demonstrates a 45.35% decrease in crossing waiting times, emphasizing efficiency. The project promotes STEM education, integrates IoT and AI for social development, and extends beyond traffic control to align with smart city objectives. Future developments may involve broader smart city integration and ongoing technological breakthroughs for increased adaptability.

Pandurangan et al., [22] proposed an IoT-based traffic management system, adjusting signal timings dynamically based on vehicle volumes. The system employs ZigBee transceiver modules and the Arduino IDE for real-time car count communication between traffic lights, reducing gridlocks. To prioritize emergency vehicles in traffic bottlenecks, the study introduces an image processing system. The findings suggest improved traffic flow, emphasizing the potential benefits of future smart city integration and ongoing technological enhancements for increased efficiency.

Naresh K S et al., [23] proposed a smart traffic management system to address urban congestion, optimizing signal timings based on real-time sensor or map-ping data. The study introduces an autonomous radio wave signaling method for prioritizing emergency vehicles, utilizing precise GPS tracking instead of traditional UV or proximity sensors. Future plans include large-scale implementation and integration with broader smart city initiatives, aiming for improved traffic flow and emergency vehicle mobility. The focus is on ongoing technological advancements for enhanced efficiency and adaptability.

3 Methodology

This project involves two primary components: hardware and software, which work collaboratively to create an intelligent traffic management system. Below, we present a comprehensive methodology for developing the Smart Traffic Light System.
System Design:

To monitor traffic on the road, raspberry pi and webcam is used. The video captured from the webcam is sent to raspberry pi. Raspberry pi has python program running on it which is done by installing rasbian the official operating system and installing python ide in it. This code uses OpenCV library and RPi.GPIO library for Raspberry Pi GPIO. It captures video frames from a camera, processes them to detect objects of a specific colour, and controls GPIO pins of a Raspberry Pi based on the detection results.

The code first initializes the GPIO pins as outputs. It then sets up the camera, defines the colour range to detect, and initializes some variables.

The program enters a while loop where it captures frames from the camera and processes them to detect the colour range of interest. It performs several image processing steps on the frames, including thresholding, dilation, and erosion, to isolate the objects of interest. It then finds the contours of the objects and draws circles around them.

The program uses the position of the detected objects to determine their location on a 2D plane. The positions are used to control the GPIO pins of the Raspberry Pi to control motors or other hardware. Finally, the program enters another while loop to continue detecting the objects and control the GPIO pins until the pro-gram is terminated.

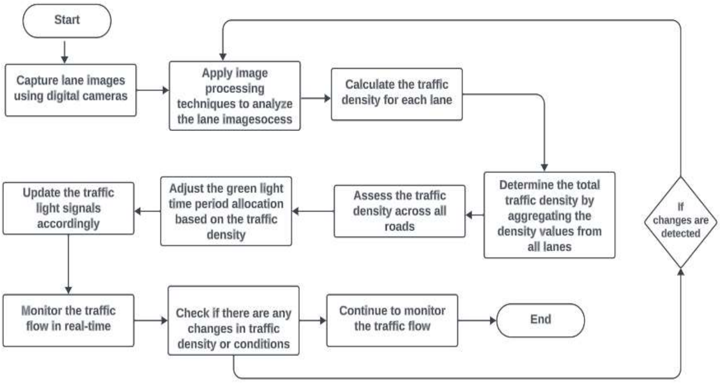


Figure 1: The block diagram provides an overview of the Smart Traffic Light System's components and interactions. It displays the Raspberry Pi at the core, connected to a camera for video input. The GPIO pins control the traffic lights. OpenCV processes the video frames, performing color thresholding and object detection. The system's decision-making, based on detected objects, influences the traffic light signals for efficient traffic management.

Implementation steps involved in creating an IOT based Solar Power Monitoring System:

3.1 Hardware Setup

To kickstart the Smart Traffic Light System project, gather the essential hardware components. These include the Raspberry Pi board, a camera, and any supplementary components, such as LED lights for the traffic signals. It's crucial to ensure that all hardware connections adhere to the provided block diagram (see Figure 1), maintaining a secure and stable setup. Additionally, for optimal results, position the camera at an elevated vantage point to effectively monitor all four roads.

3.2 Camera Configuration

3.2.1 Camera Initialization

To commence video capture, initialize the camera with cv2.VideoCapture(0). This initializes the camera for video acquisition.

3.2.2 Resolution Settings

Fine-tune the camera's resolution by setting the width to 480 pixels using cam.set(4, 480) and the height to 480 pixels with cam.set(5, 480). These configurations define the number of pixels in the horizontal and vertical dimensions, respectively.

3.2.3 Frame Rate Configuration

Set the desired frame rate using cam.set(6, 30). This choice establishes a video capture speed of 30 frames per second (FPS). Frame rate plays a pivotal role in determining how many frames or images are captured and displayed per second.

With these configurations, you ensure that the camera captures video at a resolution of 480x480 pixels while maintaining a smooth frame rate of 30 FPS. This is particularly vital for various applications, including computer vision and image processing tasks.

3.3 Connection Setup

Connect the LED lights to the appropriate GPIO (General Purpose Input/Output) pins on the Raspberry Pi as indicated in the code. Additionally, ensure that the webcam is securely connected to one of the USB ports on the Raspberry Pi.

3.4 Software Configuration

Begin by installing the Raspberry Pi's operating system, such as Raspbian, following the provided guidelines. Next, install a Python Integrated Development Environment (IDE) on the Raspberry Pi. This IDE streamlines the coding process, making it cleaner, faster, and an overall better experience.

3.5 The steps involved in Image detection and processing

3.5.1 Colour Thresholding

Prior to image processing, it's essential to establish specific color thresholds. In this context, define lower and upper threshold values for detecting a particular color, often referred to as "mid blue," within the HSV (Hue, Saturation, Value) color space. The following methods are employed:

3.5.1.1 colorLower and colorUpper are numpy arrays that set the acceptable range of blue hues, saturations, and values for objects of interest within the image.

3.5.1.2 Convert video frames from the default BGR (Blue, Green, Red) color space to the HSV color space usingcv2.cvtColor(frame,cv2.COLOR_BGR2HSV)

3.5.2 Image Processing:

After defining the color thresholds, the system engages in several image processing steps:

3.5.3 Apply Color Filtering:

cv2.inRange(hsv, colorLower, colorUpper) is used to apply color filtering and isolate objects of interest (in this case, blue-colored objects) from the converted HSV image. This function creates a binary mask in which white pixels correspond to the blue objects within the specified color range, while black pixels represent other colors.

3.5.4 Blur the Image:

To reduce noise and smooth out the binary mask, cv2.blur(mask, (3, 3)) is employed. This blurring operation helps in creating a cleaner mask and removing small imperfections, making it easier to work with the mask during further processing.

3.5.5 Dilate the Image:

cv2.dilate(mask, None, iterations=10) is used to perform dilation on the binary mask. Dilation increases the size of white regions (representing blue objects) in the mask, accentuating object features and making them more prominent.

3.5.6 Erode the Image:

Subsequently, cv2.erode(mask, None, iterations=1) is applied to erode the binary mask. Erosion reduces the size of white regions while preserving the general structure. This step is typically used to detach connected objects and further refine the mask.

3.5.7 Detect Contours:

Finaly,cv2.findContours(thresh,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE) is utilized to detect contours in the binary mask obtained after thresholding. Contours represent the shapes and boundaries of isolated objects in the mask. The retrieved contours can be further analyzed, and the detected objects can be processed or classified based on their shapes and positions.

3.6 Priority Assignment:

The system follows specific rules to assign priorities to vehicles based on their positions within the frame. If a quadrant has priority, switch the traffic lights accordingly. Adjust the traffic light signals to either favor vertical or horizontal traffic flow. Update the traffic light states based on the detected vehicle positions and priority. In instances where a large vehicle, such as a bus, is detected, it is granted priority status. This status influences traffic management in the quadrant where the large vehicle is located.

3.7 Calibration:

Once the hardware and software setup are complete, it's time to run the program on the Raspberry Pi for calibration. During this phase, place four uniquely identifiable objects at strategic positions corresponding to the joints of the road sides. The camera will capture images of these objects, and the system should be programmed to recognize them as the four joints of the roadsides.

3.8 Testing:

Post-calibration, it is imperative to conduct thorough testing to verify the system's functionality. This involves closely monitoring its performance in detecting and tracking vehicles from different lanes. Any necessary adjustments or refinements to the system's code or hardware setup are made based on the outcomes of these tests.

By meticulously following this methodology and employing OpenCV and VideoCapture methods, the Smart Traffic Light System effectively monitors traffic and optimizes traffic signal control in real-time based on prevailing conditions

Components Used:

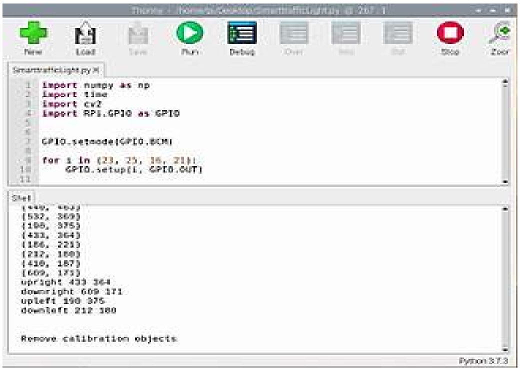
1 Raspberry pi:
Raspberry Pi is a series of small single-board computers developed by the Raspberry Pi Foundation in association with Broadcom. Raspberry Pi boards are typically credit card-sized and consist of a main processor, RAM, storage options (microSD card slot), input/output ports (USB, HDMI, Ethernet, etc.), and GPIO (General Purpose Input/Output) pins for connecting external devices and sensors.

2 USB Camera:
A USB camera, also known as a USB webcam, is a type of digital camera that connects to a computer or other devices via a USB port. It is designed for capturing and transmitting video and often audio in real-time over the internet or for local recording.

A USB camera, or webcam, connects to devices via USB, capturing and transmitting real-time video and often audio. Widely used in fields like computer vision, robotics, and surveillance, USB cameras offer advantages in ease of use, affordability, and compatibility. They vary in form, from compact webcams to larger ones with interchangeable lenses, incorporating camera sensors, lenses, and built-in microphones. When selecting one for research, consider factors like image quality, resolution, frame rate, low-light performance, and compatibility with hardware and software. Some may include advanced features such as autofocus, zoom, pan, tilt, and exposure control, catering to specific research needs.

3 OpenCV Library:
Popular open-source software called OpenCV is used for computer vision and machine learning. It offers a variety of algorithms and methods for processing images and videos, finding objects, and other things. It provides flexibility and ease of integration and supports a variety of programming languages. Robotics, augmented reality, and medical imaging are just a few of the industries where OpenCV is extensively employed. It makes difficult computer vision tasks simpler and makes it possible for programmers to design sophisticated vision-based applications.

4 Results

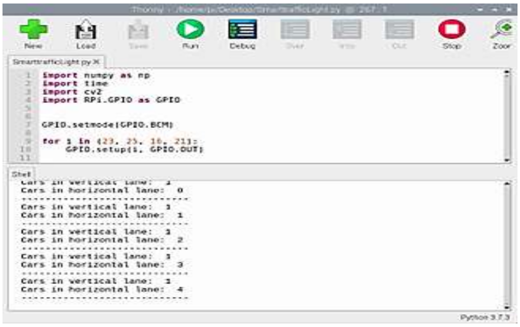


```
import numpy as np
import time
import cv2
import RPi.GPIO as GPIO


GPIO.setmode(GPIO.BCM)

for i in (23, 25, 16, 21):
    GPIO.setup(i, GPIO.OUT)

```

```
(532, 369)
(190, 375)
(433, 364)
(186, 223)
(212, 180)
(410, 187)
(609, 171)
upright 433 364
downright 609 171
upleft 190 375
downleft 212 180


Remove calibration objects
```

Figure 2: Console view After the calibration

[Figure 2 shows the console view after the calibration process, indicating that the system is ready for operation.]



```
import numpy as np
import time
import cv2
import RPi.GPIO as GPIO


GPIO.setmode(GPIO.BCM)

for i in (23, 25, 16, 21):
    GPIO.setup(i, GPIO.OUT)
```

```
Cars in vertical lane:  1
Cars in horizontal lane:  0
----------------------------
Cars in vertical lane:  1
Cars in horizontal lane:  1
----------------------------
Cars in vertical lane:  1
Cars in horizontal lane:  2
----------------------------
Cars in vertical lane:  1
Cars in horizontal lane:  3
----------------------------
Cars in vertical lane:  1
Cars in horizontal lane:  4
----------------------------
```

Figure 3: Console view Detection of cars

[Figure 3 displays the console view during the detection of cars, for the real-time monitoring of traffic.]

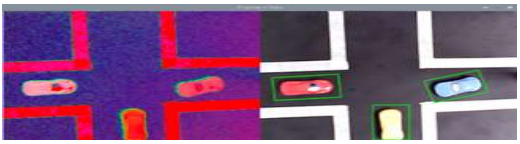


Figure 4: Frame + hsv created by program for detection of cars

[Figure 4 illustrates the frame and HSV (Hue, Saturation, Value) created by the program for the detection of cars. HSV is a color space that separates the intensity information from color information, aiding in more effective color-based object detection.]

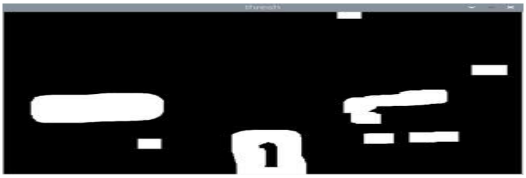
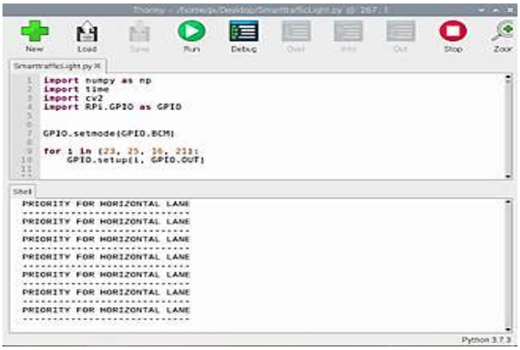


Figure 5: Thresh created by program for detection of cars

[Figure 5 presents the threshold (Thresh) created by the program for the detection of cars. Thresholding is a crucial step in image processing, where pixels are classified as object or background based on their intensity values.]

Figure 6: Masks created by program for detection of cars

[Figure 6 shows the masks created by the program for the detection of cars. Masks help isolate specific regions of interest in an image, in this case, the areas where cars are detected.]



```
import numpy as np
import time
import cv2
import RPi.GPIO as GPIO


GPIO.setmode(GPIO.BCM)

for i in (23, 25, 16, 21):
    GPIO.setup(i, GPIO.OUT)
```

```
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
PRIORITY FOR HORIZONTAL LANE
----------------------------
```

Figure 7: Console and camera view of Detection of Priority Vehicle

[Figure 7 provides a combined view of the console output and the camera feed during the detection of a priority vehicle. This demonstrates the system's ability to prioritize certain vehicles based on predefined rules, such as density of vehicles.]

5 Limitations

- Accuracy of the measurement: The accuracy of system depends on how much car color varies from the background(road). It may face a problem while detecting black and gray cars.
- System Compatibility: The system is only compatible with Raspberry pi as it pro-vides IO pins to control signal lights. To use circuit boards similar to Raspberry pi, user may have to rewrite the code and modify the circuit.
- Cost: System cost varies depending on quality of camera and Raspberry pi version. The cost of these components may be high, which makes the system costlier.
- Maintenance: The system may require periodic maintenance and calibration to ensure the accuracy of a system. The maintenance is needed time to time and for that technical knowledge and effort is required.
- Power Supply: In case of power shortage, it may damage circuit permanently so make sure that system have enough power supply. Also, in case of power cuts manual restart of the system and calibration is required.
- Compatibility with different types of cameras: The system is designed for USB types of cameras not for the Bluetooth or wi-fi models.
- Legal and Regulatory Challenges: The implementation of the system may encounter legal and regulatory challenges, as it could require approvals from relevant authorities. Failure to comply with regulations can have legal implications. To overcome this limitation, it is essential to collaborate closely with regulatory authorities and obtain the necessary approvals, ensuring adherence to legal requirements.

6 Future Scope

One potential future direction for our traffic monitoring system is to leverage advanced data sets to accurately detect and differentiate between cars, trucks, and priority vehicles, thereby resolving the issue of confusion that currently exists between trucks and priority vehicles. By implementing additional coding enhancements, we can expand the capabilities of our system to include measuring the speed of cars on the road. This can be achieved by utilizing high-quality cameras and an IoT network to capture images of the number plates of cars that are exceeding the speed limit. Subsequently, an online bill can be generated and sent to the vehicle owner for appropriate action and compliance. Moreover, we can establish an IoT network that facilitates communication between two traffic lights, thereby enhancing efficiency. In the event of an emergency vehicle passing through one traffic signal, this information can be transmitted to the other signal located in the direction the emergency vehicle is heading. This proactive communication between traffic signals enables the signals to prepare and allocate lanes for the emergency vehicle even before it reaches the intersection, optimizing traffic flow and ensuring a swift response for the emergency vehicle.

7 Conclusion

In conclusion, this research paper proposed a smart traffic light system that harnesses the power of advanced technologies such as IoT, and machine learning to address the challenges of road safety and traffic management in metropolitan areas. The traditional traffic light systems, operating on fixed schedules, often fail to adapt to changing traffic conditions, resulting in congestion, delays, and accidents. The proposed system aimed to optimize traffic flow, enhance road safety, and improve overall travel efficiency. By incorporating real-time traffic monitoring, predictive analytics, and adaptive signal control algorithms, the smart traffic light system dynamically adjusted traffic light timings based on the current traffic situation. Through simulations and real-world experiments, the system's performance was evaluated, demonstrating significant improvements in traffic flow efficiency, reduction in travel time, and alleviation of congestion.

The results of this research highlight the potential of smart traffic light systems to revolutionize traffic management in urban areas. By leveraging advanced technologies, these systems can adapt to real-time traffic conditions and prioritize emergency vehicles leading to safer roads and reduced environmental impact. Furthermore, the paper discussed future directions, including enhanced vehicle detection capabilities, speed measurement, and proactive communication be-tween traffic signals, which further contribute to the development of smarter and more efficient traffic management systems.

In summary, the proposed smart traffic light system has the potential to transform urban traffic management, improve road safety, and create more sustainable and efficient transportation systems in metropolitan areas.

References

[1] A'isya Nur Aulia Yusuf, Ajib Setyo AriFin, Fitri Yuli Zulkifli. "Recent develop- ment of smart traffic lights"https://www.researchgate.net/publication/349891471_Recent_development_of_smart_traffic_lights

[2] Anurag Kanungo, Ayush Sharma, Chetan Singla. "Smart Traffic Lights Switching and Traffic Density Calculation using Video processing" https://www.researchgate.net/publication/269310721_Smart_traffic_lights_switching_and_traffic_density_calcula- tion_using_video_processing

[3] Pavani, P. Krishna Priya, Manish Gupta, Kapil Joshi, Bajila Swathi, Mohit Tiwari : "The Use of Images Analysis for IoT-Based Smart Traffic Densities Management", 2023 3rd International Conference on Advance Computing and Innovative Technologies in En-gineering (ICACITE) | Publisher: IEEE https://ieeexplore.ieee.org/document/10182734

[4] Aws Magableh, Mohanad Almakhadmeh, Nawaf Alsrehin, Ahmad F Klaib: "Smart Traffic Light Management Systems: Systematic Literature review"

https://www.researchgate.net/publication/342621214_Smart_Traffic_Light_Management_Systems_A_Systematic_Literature_Review

[5] Shilpa S. Chavan, Dr. R. S. Deshpande, J. G. Rana. “DESIGN OF INTELLIGENT TRAFFIC LIGHT CONTROLLER USING EMBEDDED SYSTEM” https://iee-explore.ieee.org/abstract/document/5395443

[6] Luiz Fernando Pindo de Oliverira, Leandro Tiago Manera, Pailo Denis Garcez da Luz. “Development of a Smart Traffic Light Control System with Real-Time Monitoring”https://ieeexplore.ieee.org/abstract/document/9187855\

[7] Meisam Razavi,MehdiHamidkhani,Rasool Sadeghi : “Smart Traffic Light Scheduling in Smart City Using Image and Video Processing” 2019 3rd International Conference on Internet of Things and Applications (IoT) | Publisher: IEEE https://ieeexplore.ieee.org/document/8808836

[8] Wei-Hsun Lee, Chi-Yi Chiu. “Design and Implementation of a Smart Traffic Sig- nal Control System for Smart City Applications” -https://www.mdpi.com/1424- 8220/20/2/508

[9] Marco Wiering, Jelle van Veenen, Jilles Vreeken, Arne Koopman. “Intelligent Traffic Light” https://dspace.library.uu.nl/bitstream/handle/1874/17996/wiering_04_intelligent_traffic.pdf?sequence=2

[10] 1Maram Bani Younes, Azzedine Boukerche. “Intelligent Traffic Light Controlling Algo- rithms Using Vehicular Networks” https://ieeexplore.ieee.org/abstract/docu- ment/7277098

[11] T. V. Janardhana Rao, R. Arun Sekar, B. Drakshyani, P Kalyan Chakravarthi, N. Ku-mareshan, S. P. Karthi: “IOT based Smart solution for Traffic congestion at Metro Cit-ies”, 2022 International Conference on Innovative Computing, Intelligent Communica-tion and Smart Electrical Systems (ICSES) | Publisher: IEEE https://ieeexplore.ieee.org/document/9914174

[12] Khalil M. Yousef, Jamal N. Al-Karaki, Ali M. Shatnawi. “Intelligent Traffic Light Flow Control System Using Wireless Sensors Networks” https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=87f25a4f6bda6a0a255efff7426e55a24362431f

[13] Volodymyr Miz, Vladimir Hahanov. “Smart traffic light in terms of the Cognitive Road traffic management system (CTMS) based on the Internet of Things” https://ieeex- plore.ieee.org/abstract/document/7027102

[14] Satbir Singh, Baldev Singh, Ramandeep, Baljit Singh, Amitava Das “Automatic Vehicle Counting for IoT based Smart Traffic Management System for Indian urban Settings” 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT- SIU) https://ieeexplore.ieee.org/document/8777722

[15] Vikram Bali, Sonali Mathur, Vishnu Sharma, Dev Gaur “Smart Traffic Management System using IoT Enabled Technology” 2020 2nd International Conference on Ad-vances in Computing, Communication Control and Networking (ICACCCN) | IEEE https://ieeexplore.ieee.org/document/9362753

[16] Abida Sharif,JianpingLi,MudassirKhalil,RajeshKumar,Muhammad Irfan Sharif,Atiqa Sharif “Internet of things — smart traffic management system for smart cities using big data analytics” 2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP) | Publisher: IEEE https://ieeexplore.ieee.org/document/8301496

[17] Sabeen Javaid,AliSufian,SaimaPervaiz,Mehak Tanveer: “Smart traffic management system using Internet of Things”, 2018 20th International Conference on Advanced Communication Technology (ICACT) | Publisher: IEEE https://ieeexplore.ieee.org/document/8323770

[18] Patan Rizwan, K Suresh, M. Rajasekhara Babu: “Real-time smart traffic management system for smart cities by using Internet of Things and big data”, 2016 International Conference on Emerging Technological Trends (ICETT) | Publisher: IEEE https://ieeexplore.ieee.org/document/7873660

[19] Sanjay Kumar Sahu, Atul Basant, Taman Vasudev,Kusagra Khati, Nikhil Lawrence : “TRAFFIC MANAGEMENT SYSTEM USING IOT”, 2021 JETIR April 2021, Volume 8, Issue 4 www.jetir.org (ISSN-2349-5162) https://www.jetir.org/papers/JETIR2104274.pdf

[20] Sunny Hossain, Farzana Shabnam: “A Comparative Study of IoT Based Smart Traffic Management System” 2021 IEEE International Women in Engineering (WIE) Confer-ence on Electrical and Computer Engineering (WIECON-ECE) | Publisher: IEEE https://ieeexplore.ieee.org/document/9829636

[21] BaslaSiripatana;KittipongNopchanasuphap;SompornChuai-Aree: “Intelligent Traffic Light System Using Image Processing”, 2021 2nd SEA-STEM International Conference (SEA-STEM) | Publisher: IEEE https://ieeexplore.ieee.org/document/9668057

[22] Raji Pandurangan, Vadlamudi Sai GuruKrishna, Golla Vinay Kumar, Kalamakuntla Bhavani: “IOT Based Traffic Controlling System”, 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON) | Publisher: IEEE https://ieeexplore.ieee.org/document/10083882

[23] Naresh K S1, Francis J kalliath2, Shravan Aruljothi3, Arni Tharakaram Hariram4, Harshith Shivakumar: “Smart Traffic management system using IOT”, International Journal of Engineering Trends and Technology (IJETT- Scopus Indexed) – Special Is-sues - ICT 2020 https://www.internationaljournalssrg.org/uploads/specialissuepdf/ICT-2020/2020/OTHERS/P119.pdf