# Adaptive Traffic Signal Network using MATLAB and ThinkSpeak

Kavya Sree Gogadi

May 30, 2024

# Contents


**1 Introduction** **2**

**2 Project Description** **2**
- 2.1 Objectives . . . 2
- 2.2 Methodologies . . . 3

**3 System Architecture** **3**
- 3.1 Camera . . . 3
- 3.2 MATLAB . . . 3
- 3.3 ThinkSpeak . . . 4
- 3.4 Communication . . . 4
- 3.5 Software Components . . . 4

**4 Installation and Setup** **4**
- 4.1 Hardware Setup . . . 4
- 4.2 Software Setup . . . 5
- 4.3 Steps . . . 5

**5 Data Collection and Analysis** **5**
- 5.1 MATLAB Code Explanation . . . 5
  - 5.1.1 Key Sections of the Code . . . 6
- 5.2 Data Analysis . . . 7

**6 Implementation** **8**
- 6.1 Setup and Testing . . . 8
- 6.2 Challenges and Solutions . . . 10

**7 Results and Findings** **10**

**8 Usage and Applications** **11**

**9 Conclusion** **12**

# 1 Introduction

Urban areas face increasing challenges in managing traffic flow efficiently, leading to congestion, delays, and safety concerns. In response, this project presents a pioneering solution: an Adaptive Traffic Signal Network system leveraging MATLAB, a camera, and the ThinkSpeak platform. This system aims to revolutionize traffic management strategies by harnessing real-time data collection and analysis, offering insights to alleviate congestion and enhance urban mobility.

This project endeavors to develop a comprehensive real-time traffic signal monitoring and analysis framework. The system promises to offer actionable intelligence for optimizing traffic flow and enhancing road safety through the seamless integration of IoT technologies with sophisticated data processing capabilities.

# 2 Project Description

The project aims to develop a real-time traffic signal monitoring and analysis system utilizing IoT technology in conjunction with MATLAB, a camera, and the ThinkSpeak platform. This system is designed to collect, process, and analyze traffic data with the goal of improving traffic management strategies and reducing congestion in urban areas.

## 2.1 Objectives

The primary objectives of the project include:

- Implementing an IoT-based framework for capturing real-time video data from traffic signals.
- Utilizing MATLAB for processing and analyzing the captured video data to extract meaningful insights about traffic patterns and density.
- Integrating the ThinkSpeak platform for storing, visualizing, and further analyzing the processed traffic data.
- Developing a scalable and adaptable system architecture capable of addressing diverse traffic management challenges in urban environments.

## 2.2 Methodologies

The project will employ the following methodologies to achieve its objectives:

- Hardware Setup: Installation of cameras at strategic locations to capture real-time video data of traffic signals and traffic flow.
- Software Development: Writing MATLAB scripts for video processing, data analysis, and integration with the ThinkSpeak platform.
- Data Collection and Analysis: Continuous monitoring and analysis of traffic data to identify patterns, trends, and congestion hotspots.
- System Testing and Optimization: Calibration and testing of the system to ensure accuracy, reliability, and scalability.
- Evaluation and Validation: Assessing the effectiveness of the system in improving traffic management strategies through real-world deployment and performance evaluation.

# 3 System Architecture

The system architecture of the traffic signal monitoring and analysis system comprises several interconnected components working together to capture, process, analyze, and visualize traffic data in real-time. The key components of the system include:

## 3.1 Camera

The camera serves as the primary data acquisition device, capturing real-time video footage of traffic signals and surrounding traffic flow. Positioned strategically at vantage points, the camera provides a continuous stream of video data for analysis.

## 3.2 MATLAB

MATLAB serves as the core processing engine of the system, responsible for analyzing the captured video data to extract meaningful insights about traffic density, patterns, and trends. Utilizing MATLAB's image processing capabilities, the system identifies and analyzes various aspects of traffic behavior.

### 3.3 ThinkSpeak

ThinkSpeak acts as the data storage and visualization platform, where the processed traffic data is stored, visualized, and further analyzed. ThinkSpeak provides real-time dashboards and analytics tools for monitoring traffic patterns and making informed decisions regarding traffic management strategies.

### 3.4 Communication

The communication module facilitates seamless data exchange between the components of the system. Utilizing Wi-Fi connectivity, the system transmits processed data from MATLAB to the ThinkSpeak platform for storage and visualization.

### 3.5 Software Components

The software components of the system include:

- MATLAB: For video processing and data analysis.
- ThinkSpeak: For data storage and visualization.
- Firmware: Code for interfacing the camera with MATLAB and ThinkSpeak, ensuring smooth data transmission and integration.

## 4 Installation and Setup

### 4.1 Hardware Setup

- **Camera Installation:** Position the camera to capture the traffic signal and traffic flow.
- **Power Supply:** Connect the camera to a stable power source.
- **Network Connection:** Ensure the camera and computer running MATLAB are connected to the same Wi-Fi network.

## 4.2 Software Setup

1. **MATLAB Setup:**
   - Install necessary toolboxes for image processing and ThinkSpeak integration.
   - Write or load the MATLAB script for processing video data and sending it to ThinkSpeak.
2. **ThinkSpeak Configuration:**
   - Create a ThinkSpeak account and set up a new channel.
   - Configure fields in the channel to store traffic data.
   - Generate API keys for data transmission.

## 4.3 Steps

1. **MATLAB Code Execution:**
   - Run the MATLAB script to start capturing and processing video data.
   - Ensure the script sends data to the configured ThinkSpeak channel.
2. **ThinkSpeak Dashboard:**
   - Set up dashboards and widgets on ThinkSpeak to visualize the data.
   - Monitor the real-time data and analytics.

# 5 Data Collection and Analysis

## 5.1 MATLAB Code Explanation

The provided MATLAB code captures video frames using a webcam, processes these frames to detect traffic density at different directions (north, south, east, west), and sends the processed data to ThinkSpeak.

### 5.1.1 Key Sections of the Code

**Camera Initialization:**

```
cam2 = webcam('HP Wide Vision HD Camera');
```

Initializes the camera for capturing video.

**Frame Capture Loop:**

```
while(framecnt < 100)
```

Captures and processes 100 frames.

**Image Processing:**

```
inputframe = snapshot(cam2);
Rgbframe = inputframe;
a = imread('mask.bmp');
b = rgb2gray(a);
c = b > 253;
[Label Num] = bwlabel(c);
prop = regionprops(Label, 'BoundingBox');
```

Processes the captured frame to identify regions of interest using a mask.

**Region Analysis and Traffic Detection:**

```
for i = 1: Num
    filename = strcat(num2str(i), '.bmp');
    Bbox = prop(i).BoundingBox;
    Bbox = round(Bbox);
    backgroundframe = imread(filename);
    currentframe = imcrop(inputframe, Bbox);
    Area = 200;
    Thre = 100;
    [result normalizetrafficarea] = videodetection(backgroundframe,
    currentframe, Thre, Area);
    if i == 1
        str1 = 'south';
        south = normalizetrafficarea;
    elseif i == 2
        str1 = 'west';
```

```
        west = normalizetrafficarea;
    elseif i == 3
        str1 = 'east';
        east = normalizetrafficarea;
    else
        str1 = 'north';
        north = normalizetrafficarea;
    end

    if result == 1
        Rgbframe = insertObjectAnnotation(Rgbframe, 'rectangle',
        Bbox, str1, 'Color', {'red'});
    else
        Rgbframe = insertObjectAnnotation(Rgbframe, 'rectangle',
        Bbox, str1, 'Color', {'cyan'});
    end
end
```

Analyzes each labeled region to detect traffic density and annotates the results on the image.

**Data Transmission to ThinkSpeak:**

```
thingSpeakWrite(2565857, {north, south, east, west}, 'WriteKey',
'ZDYU8JXFFVDMVO3V');
disp('data written');
pause(20);
```

Sends the processed data to ThinkSpeak for storage and visualization.

## 5.2 Data Analysis

- **Data Collection:** Video data is continuously captured and processed by MATLAB to determine traffic density in four directions.
- **Data Analysis:** MATLAB processes the data to extract meaningful traffic patterns and density information.
- **Data Storage:** Processed data is sent to ThinkSpeak for real-time storage and visualization.

# 6 Implementation

## 6.1 Setup and Testing

- The camera and MATLAB script were calibrated to ensure accurate data collection.
- Adjustments in camera angles, optimisation of MATLAB code, and ensuring a stable network connection were implemented to resolve challenges.

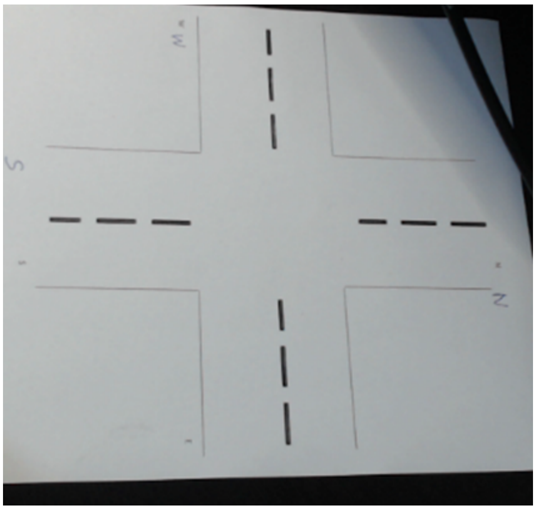

Figure 1: Image Captured by the Camera

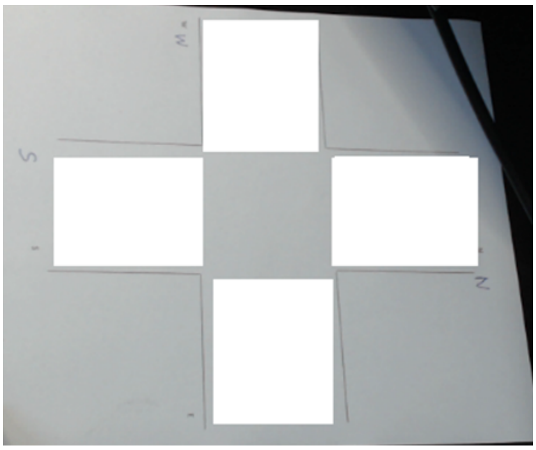

Figure 2: Masked Image

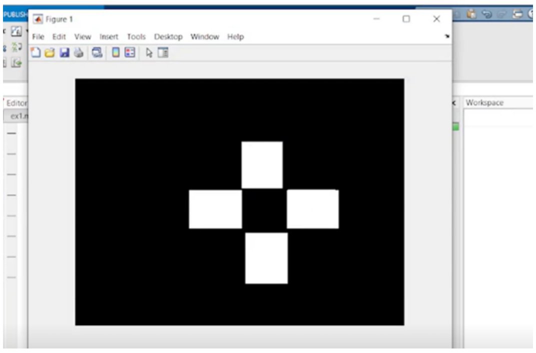

Figure 3: Converting the picture into Binary

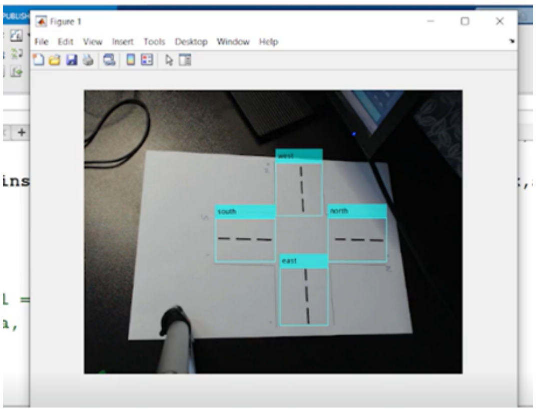

Figure 4: Testing Boundaries

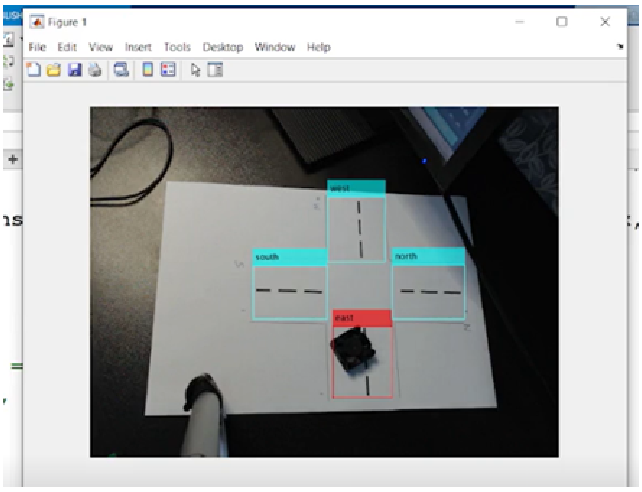

Figure 5: Testing with Object for Traffic Detection

## 6.2 Challenges and Solutions

- **Camera Positioning:** Adjusted angles for optimal capture.
- **Data Processing Speed:** Optimized MATLAB code for faster processing.
- **Network Reliability:** Ensured a stable Wi-Fi connection.

# 7 Results and Findings

- **Traffic Patterns:** Analysis revealed distinct traffic patterns and peak traffic hours.


Figure 6: Results Over ThinkSpeak

- **Signal Optimization:** Recommendations for traffic signal timing adjustments were made based on the data.
- **Visualizations:** ThinkSpeak provided real-time visualizations of traffic data, aiding in quick decision-making.

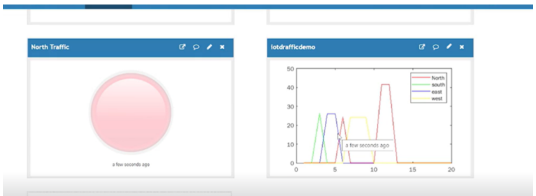


Figure 8: Data Comparision over all Boundaries

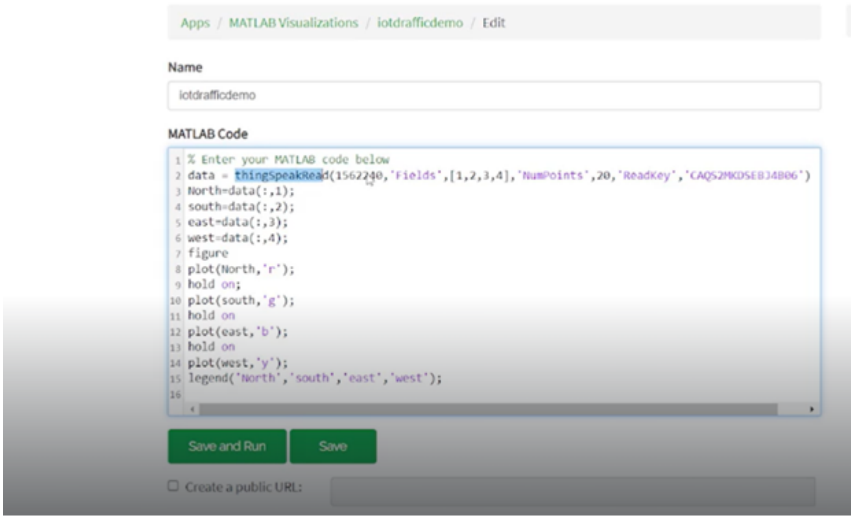


Figure 7: MALTALB Visualization

# 8 Usage and Applications

- **Traffic Management:** Authorities can use the system to monitor and optimize traffic signals in real-time.
- **Urban Planning:** Data insights assist in long-term urban planning and infrastructure development.
- **Future Enhancements:** Integration with AI for predictive analytics and expansion to multiple traffic signals.

# 9 Conclusion

In conclusion, the Adaptive Traffic Signal Network system presented in this project demonstrates the potential of leveraging modern technologies such as IoT, MATLAB, and ThinkSpeak for effective traffic management in urban areas. By collecting real-time data from traffic signals, processing it with MATLAB, and visualizing it on the ThinkSpeak platform, the system offers valuable insights for traffic management authorities.

Through the analysis of traffic patterns, peak hours, and congestion hotspots, the system provides a basis for optimizing traffic signal timings and implementing proactive measures to alleviate congestion and improve road safety. Furthermore, the real-time visualization capabilities of ThinkSpeak enable stakeholders to make informed decisions promptly, leading to more efficient traffic management strategies.

Moving forward, the system can be enhanced further through integration with advanced AI algorithms for predictive analytics and expanded to cover a broader network of traffic signals. By continuously refining and expanding the system, we can address the evolving challenges of urban traffic management and contribute to the creation of smarter, more sustainable cities.