

# Technical Report On Traffic Flow Analysis Tool

Course Code: SE-801

Submitted by

Saif Kamal Chowdhury  
BSSE0924

Supervised by

Dr. Naushin Nower  
Associate Professor  
Institute of Information Technology,  
University of Dhaka

Date of Submission: February 23, 2021



## Letter of Transmittal

23 February, 2021

SPL3 Coordinator  
Institute of Information Technology  
University of Dhaka

**Subject:** Submission of technical report on "Traffic Flow Analysis Tool".

Sir,

With due respect, I am pleased to submit the technical report on Traffic Flow Analysis Tool, A desktop-based application to provide analytical data on traffic flow in a specific area of Dhaka City of Bangladesh.

Although this report may have shortcomings, I have tried my level best to produce an acceptable technical report. I would be highly obliged if you overlooked the mistakes and accepted the effort that has been put in this report.

Yours sincerely,

Saif Kamal Chowdhury  
BSSE0924

## Table of Contents

Introduction .....	1
Project Description .....	1
Normal Requirements.....	1
Expected Requirements.....	1
Exciting Requirements .....	2
Scenario Based Modeling.....	3
Scenario/Story .....	3
Use Case Diagram .....	3
Level-0:.....	4
Level-1:.....	4
Activity Diagram .....	5
Methodology .....	7
Data Collection .....	7
Data Preprocessing .....	9
Noise Cancellation.....	9
Hough Transform and Road Detection.....	11
Color based Image Segmentation .....	11
Color based Semantic Segmentation .....	13
Showing Output .....	14
Tools and Platforms .....	14
Architectural Design.....	15
Representing System in the Context .....	15
Refine the architecture into components.....	16
High Level Testing Goals .....	17
Conclusion .....	18

## Table of Figures:

Figure 1 Level 0 Use Case.....	4
Figure 2 Level 1 Use Case (Sub Systems) .....	4
Figure 3 Activity Diagram for Analysis Module.....	5
Figure 4 Activity Diagram for the Prediction Module.....	6
Figure 5 Taking Screenshot with Selenium.....	7
Figure 6 Stored Image of an Area.....	8
Figure 7 Map Image with Noise Objects .....	9
Figure 8 Image after removal of Noise Objects.....	10
Figure 9 Intermediate Green Flow .....	11
Figure 10 Intermediate Red Flow .....	12
Figure 11 Intermediate Orange Flow .....	12
Figure 12 Final Input Image .....	13
Figure 13 Architecture Context Diagram.....	15
Figure 14 Instantiation of Software Architectural Components.....	16

## Introduction

This document contains functional and non-functional requirements specification and establish a requirement baseline for the development of the system. It also contains QFD, use case, activity diagram, methodology of my work and future planning.

I hope this document will help in understanding the functional and implementation details of my project "Traffic Flow Analysis Tool" and develop a baseline for remarks. This document is organized in such a systematic order that one can go through this and can easily understand the overall life cycle of my project.

## Project Description

This project aims to develop a system to provide analytical data on traffic flow in a specific area. For this reason, this system will begin with a data collection automation tool which will collect Image data of traffic flow from Google Maps. The image collected from google maps will have to be modified and changed in order to collect actual information from it.

Then the image data will be processed into textual data based on colors which define different levels of traffic intensity on routes. Along with traffic flow information, there will also be time, day of the week, special institution in the area etc. information which will be used to provide analysis of the traffic in a specific area. Following are the defined requirements for the project,

### Normal Requirements

Normal requirements are those which must be included within the software.

- Show analytical data of traffic intensity for of a given time in a specific area
- Graphical result of analysis

### Expected Requirements

Expected requirements are those that are not explicitly mentioned but are required to fulfill the overall functionality.

- The system should be user friendly.
- Time to generate output should not be much.
- The tool should be compatible with minimal hardware

## Exciting Requirements

Exciting requirements are those what are not mandatory to implement but provides extra value to the software.

- Predict traffic flow in a specific area on a given date and time

## Scenario Based Modeling

Scenario based modeling is the technique to illustrate the user's interaction with the system. This section describes how the user will interact with the system and use cases.

### Scenario/Story

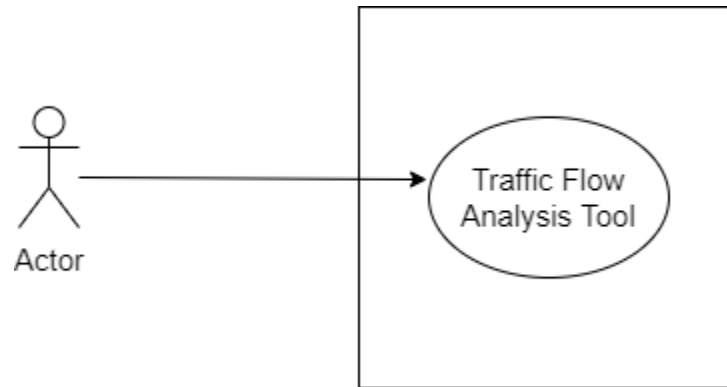
The user will be provided with an installer. After installation, he/she will be provided a graphical user interface (GUI) where he/she can select a specific area and then the day and time. Then the user will press "Generate Analysis Result" button and the backend activity of the tools will be triggered. The data that are stored for that specified area will be analyzed and finally an output image and textual result will be generated.

Another option will be provided to the user. He will press "Generate Prediction" button and a pop up will be generated where he/she will be asked to enter a specific time and area. Upon user request, the system will generate the output the given time. An image will be shown containing the predicted result.

### Use Case Diagram

Use Case diagram describes the system in terms of user interaction. The gradual breakdown shows the clearer overview of the interaction maps with the subsystem. A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

Level-0:

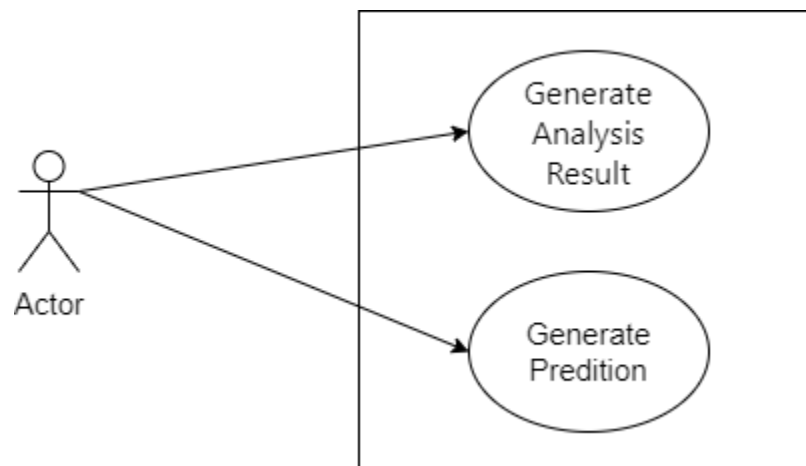


*Figure 1 Level 0 Use Case*

**Primary actor:** User

**Goal in context:** The diagram refers to the overview of the 'Traffic Flow Analysis tool'

Level-1:



*Figure 2 Level 1 Use Case (Sub Systems)*

**Primary actor:** User

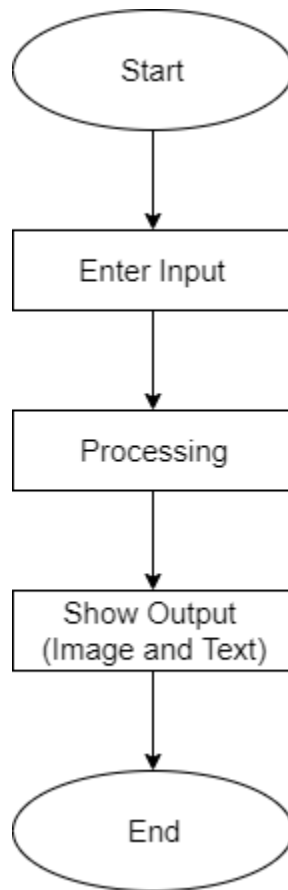
**Goal in context:** The diagram shows all the modules of the Traffic Flow Analysis tool.



## Activity Diagram

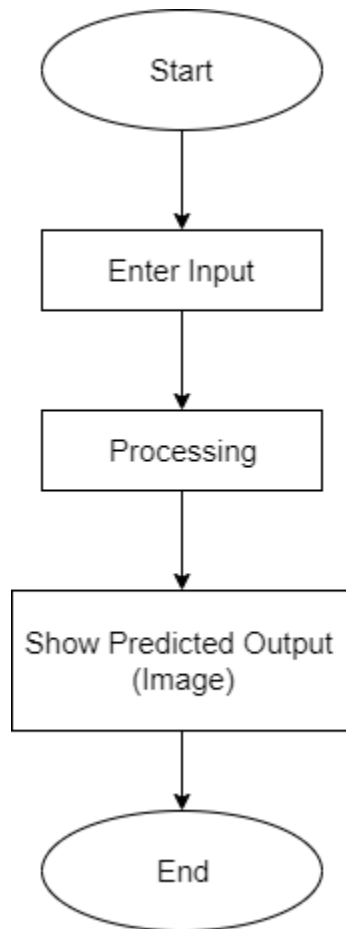
Activity diagram shows the order in which the user interacts with the system. As my system offers 2 different features to the user the following 2 diagrams are drawn to provide a high-level overview of how the system works.

The first one is to generated analytical result of the traffic flow for a specified area and time. User will enter the necessary input that includes area and day with the time of day. This will be processed in the system and finally an image and textual output will be shown to the user.



*Figure 3 Activity Diagram for Analysis Module*

The second feature is almost identical as the first one just except the fact that user will enter a future date instead of a specific day and time. The input will be processed in the system and a predicted image with Traffic Flow will be shown to the user.

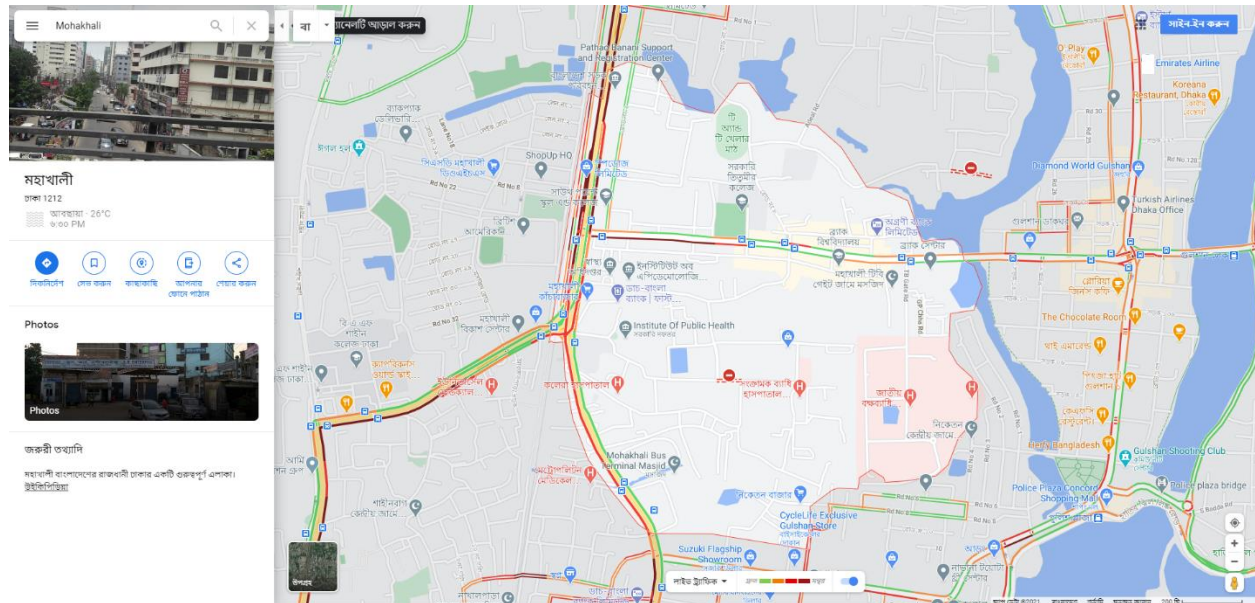


*Figure 4 Activity Diagram for the Prediction Module*

# Methodology

## Data Collection

For this project, I am using data that are collected from Google map. I will be continuously collecting screenshots with a fixed time frame for a given source and destination. A sample is provided below:



*Figure 5 Taking Screenshot with Selenium*

As this is impractical to collect and handle these data manually and it is risky to collect such pixel sensitive data with hand, I have decided to use Selenium web driver to automate the task. I have written a program using selenium web driver that does the following:

- Opens Google map in the browser with modified URL to load with Traffic Data
- Select the Search-Box
- Enter the name of the area
- Take a screenshot of the map
- Store it in the local computer in a specified path

Following is an Example of a stored image:

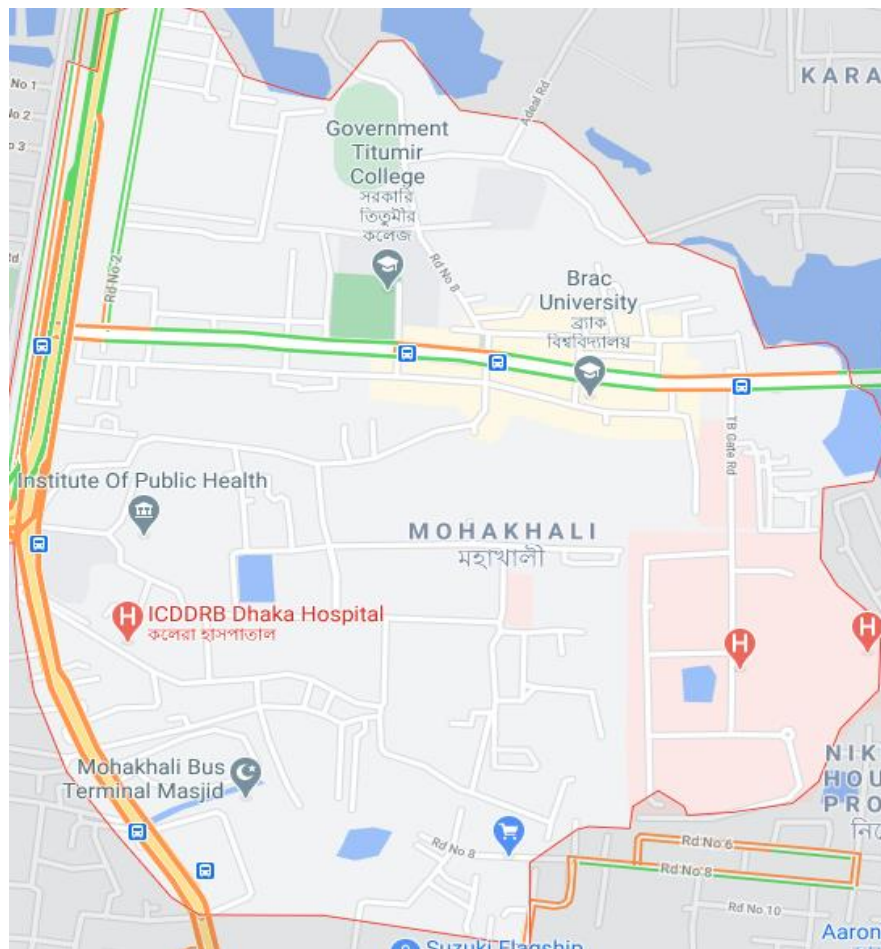


Figure 6 Stored Image of an Area

## Data Preprocessing

This one is so far the most challenging part of my project. As there are no defined measurement can be made from the Google map data, I have to do a lot of preprocessing works to extract out some meaningful information from it. Currently, I am working to find out some text data from the images I have been collecting. I needed to use some algorithms and techniques in the processing step. Few of them are listed below:

1. Noise Cancellation (Template Matching)
2. Hough Transform and Road Detection
3. Color Wise Segmentation
4. Image Masking
5. Color based Semantic Segmentation

### Noise Cancellation

The image generated from the previous step contains some noises. For example, the logo of restaurants, hospitals and others will cause some problems to extract data as these are irrelevant to my work. A manual, fine tuning of RGB value filtering was enough to obtain a satisfactory level noise free image. Example of image with noise objects:

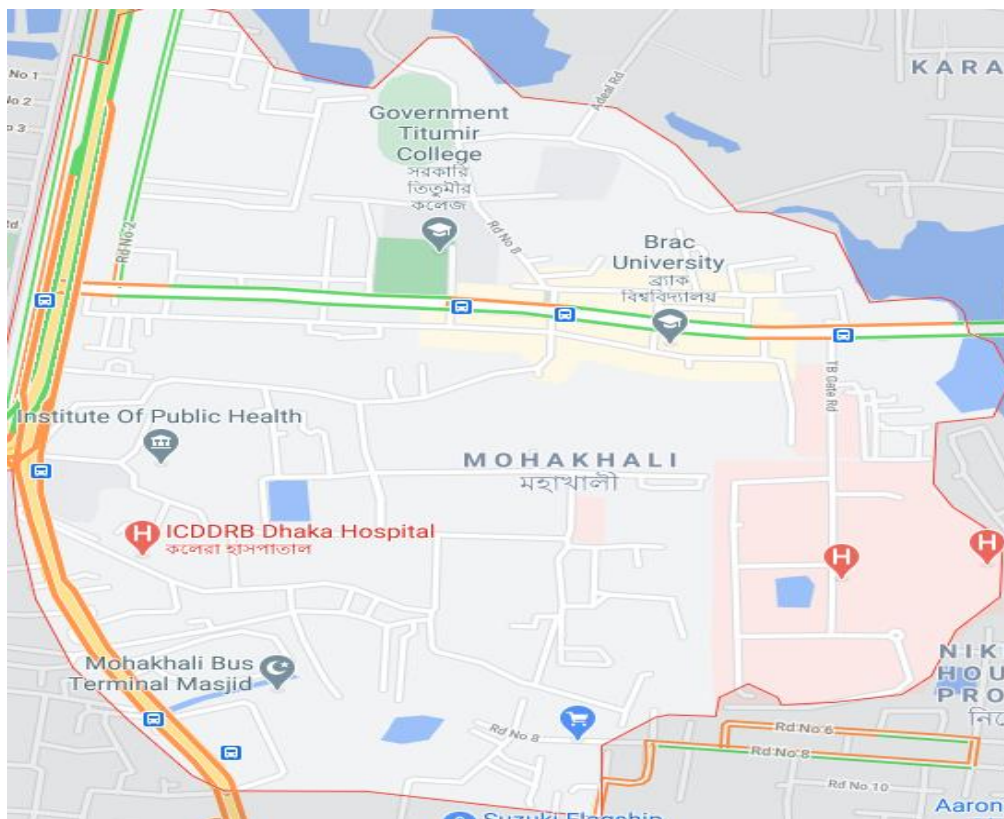


Figure 7 Map Image with Noise Objects

After template matching and color-based filtering the following is an example of Image with reduction of noise

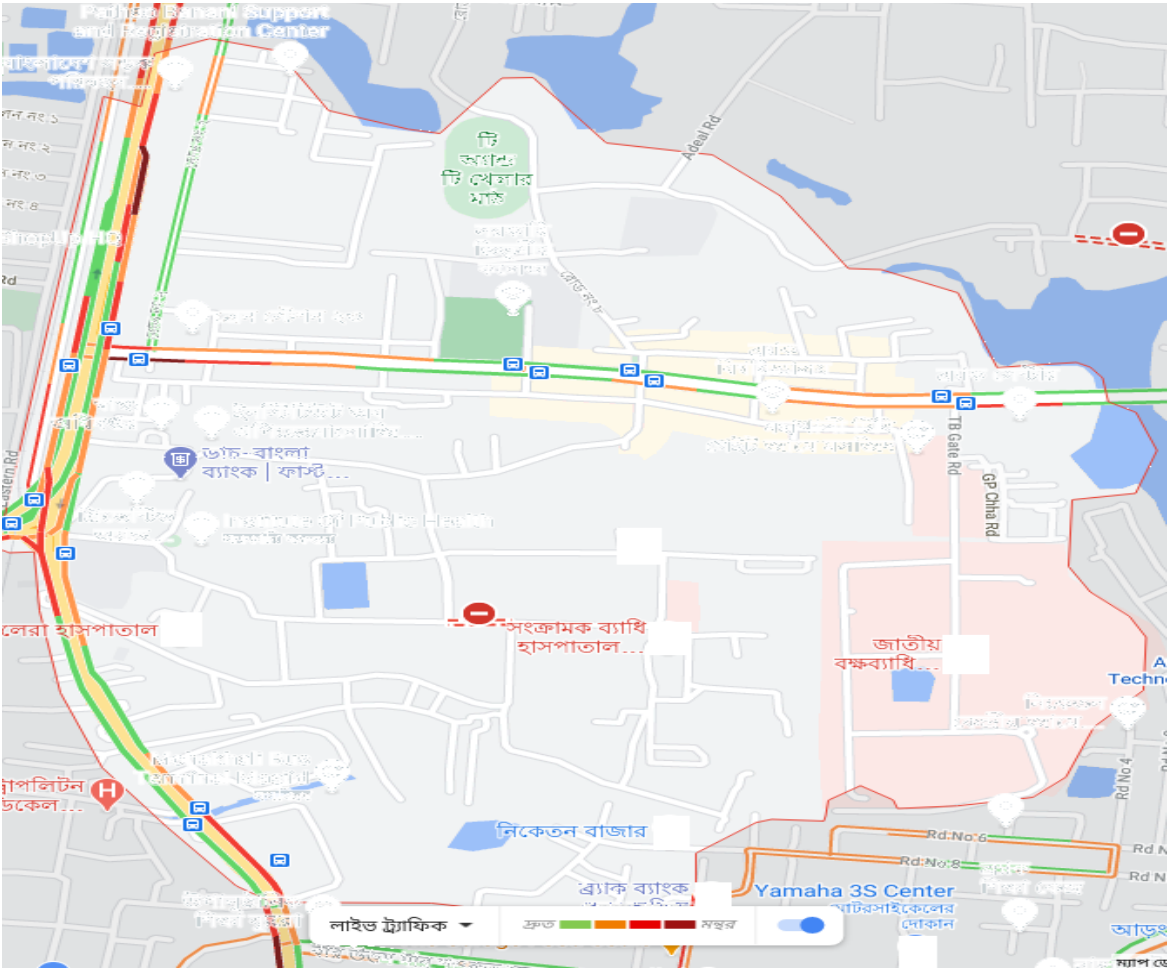


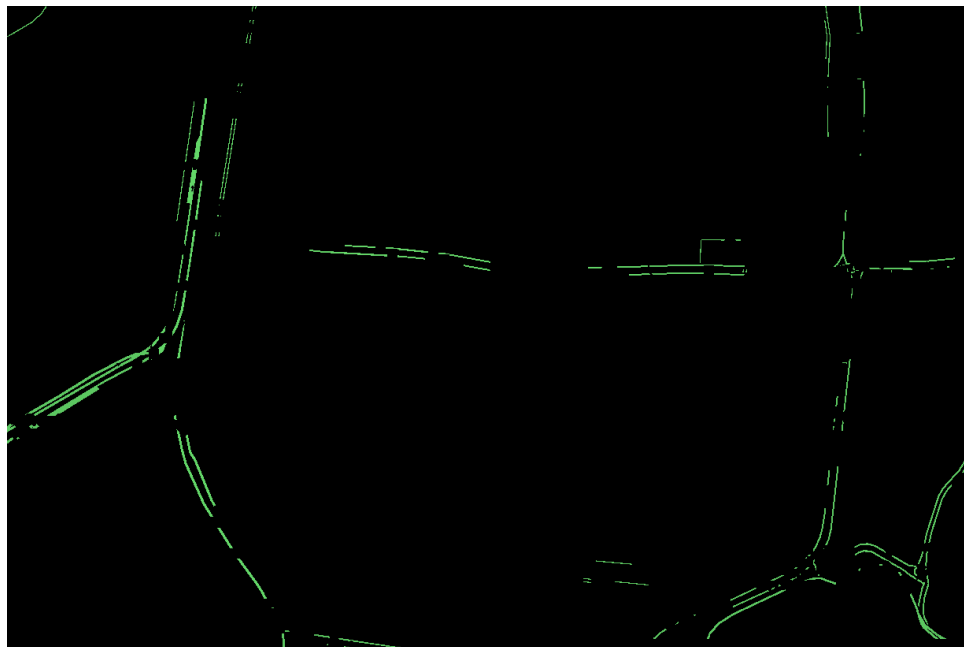
Figure 8 Image after removal of Noise Objects

## Hough Transform and Road Detection

Hough transform is a simple feature extraction technique that is used to detect lines, circle or any other simple shapes in a given picture. I have used Hough transform for detecting parallel lines from the collected image. This technique works pretty well in terms of line detection. However, my goal is to detect not only lines but also number of roads in the area and identify them as Road 1, Road 2 etc. This is a work in progress and may be part of the end project.

## Color based Image Segmentation

Google map provides traffic update of specified route with 4 different colors; red, dark red, green, orange. As detection of these colors are not possible with one go, I have to produce some intermediate images and finally merge them together. Intermediate images are listed below:



*Figure 9 Intermediate Green Flow*



Figure 10 Intermediate Red Flow

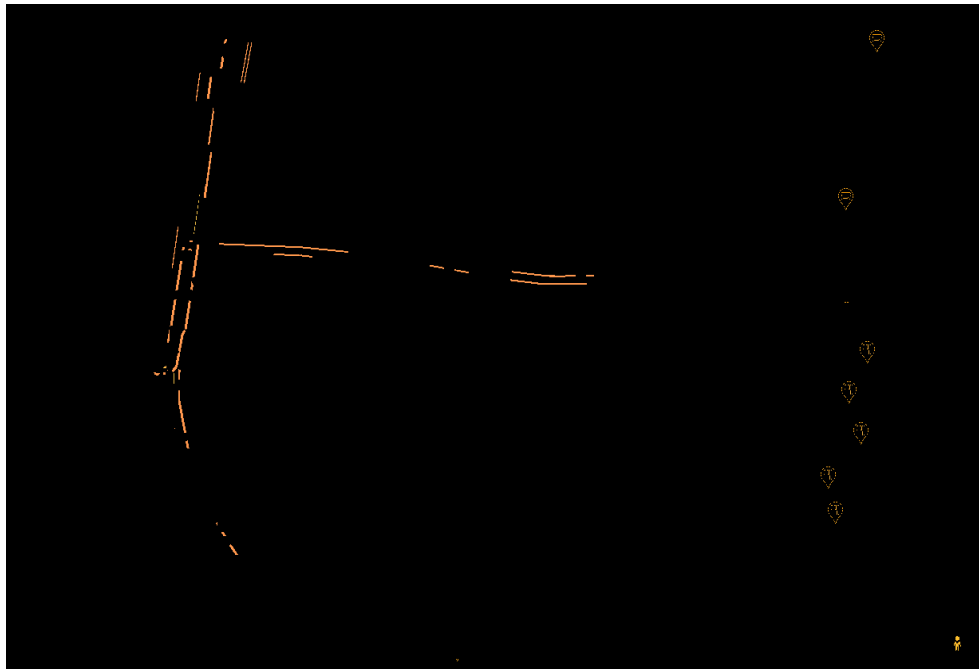


Figure 11 Intermediate Orange Flow



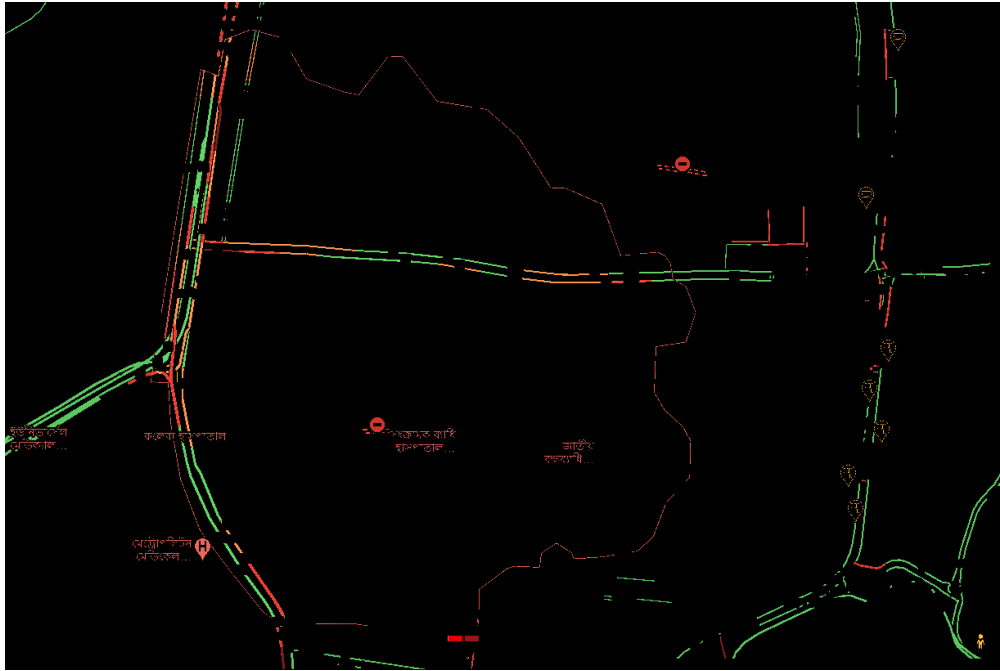


Figure 12 Final Input Image

### Color based Semantic Segmentation

As per my work progress, I am in this stage of the project. To experiment machine learning models, I need to extract some textual data. I am planning to generate a table like this,

Day	Time	Is Holiday	Segment 1	Segment 2	Segment 3	Presence of Institution
Sat	11.00	Yes	1	2	1	Y
Sun	15.45	No	4	2	2	Y
Mon	20.00	No	2	3	1	Y

## Showing Output

After the user selects a specific area for a day, I will show some analytical data in image format in the User interface and in some tabular data as well. In case of prediction, I would generate a similar map image like google with traffic data for the select area for the selected date.

## Tools and Platforms

For data collection I have been using Python language and selenium web driver to automate the browser to collect data. And for working with data processing and analysis, I am using python language. I am using PyCharm IDE for development. For the desktop application, I am using .Net Core Desktop application development

## Architectural Design

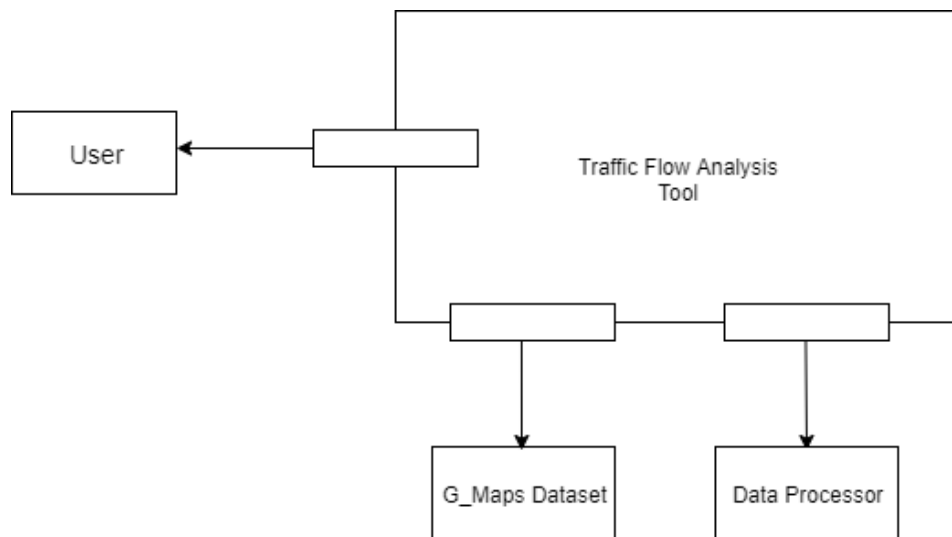
At the beginning of the architectural design the system must be put into the context it exists. Software in this era are no longer standalone entity but a part of the communication end of an ecosystem. So, software should be defined in terms of its relation with other entities and the type of interaction. This chapter provides a brief overview of the architectural design of my system.

### Representing System in the Context

At the architectural design level, the system is described with an Architectural Context Diagram (ACD). The main purpose of this diagram is to provide a high-level overview of the system in terms of its relation with external entities and the type of relation. System that interacts with the target system (the system to be developed) are further classified into the following:

- Superordinate system
- Subordinate system
- Peer level system

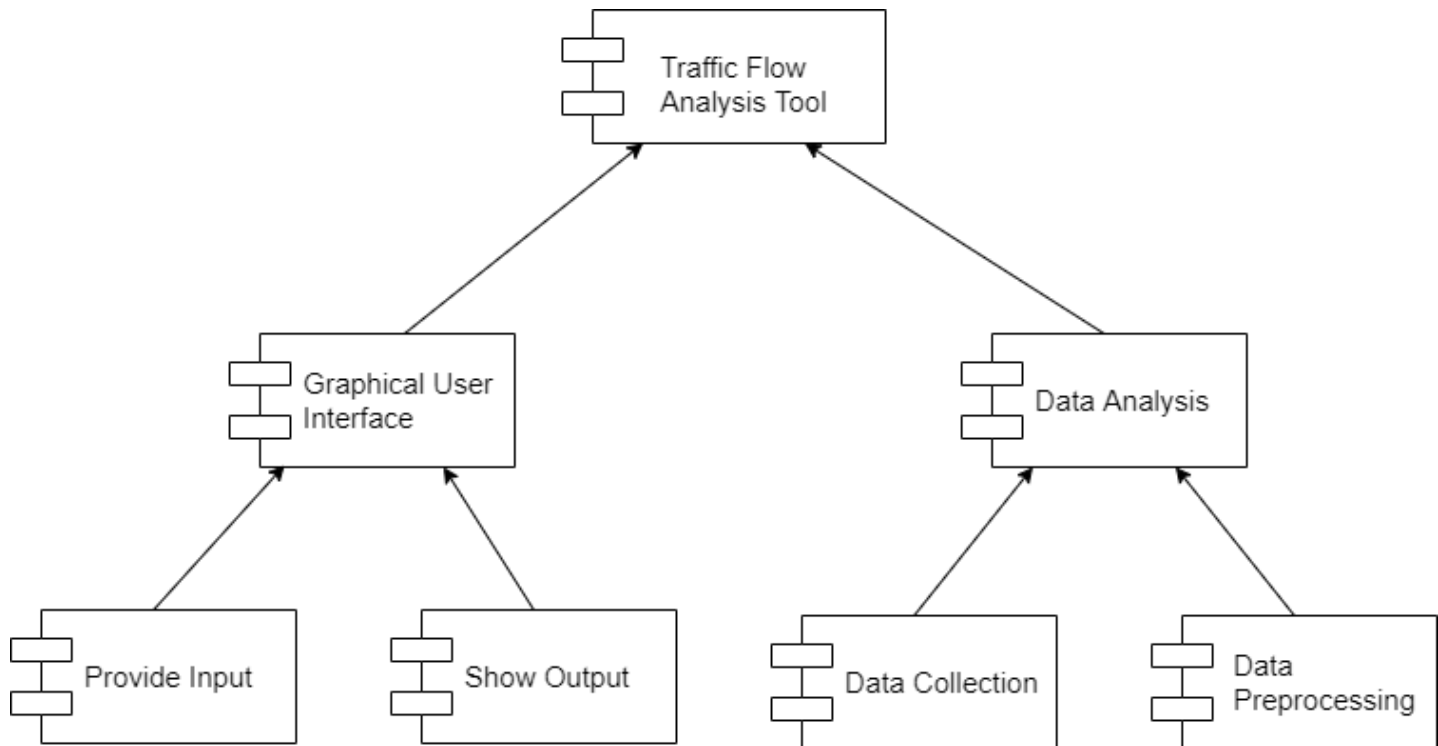
The ACD diagram of the system 'Traffic flow analysis Tool' is provided below:



*Figure 13 Architecture Context Diagram*

## Refine the architecture into components

As the system is refined into components the structural description of the system starts to be emerged. The components that are specific to the system functionalities are considered to be the most important ones and others that helps to provide functionality such as the Graphical User Interface is called to be the infrastructure ones. The application domain is essential to be discussed in the architectural design of the system. As described in the Architectural Context Diagram the following components that participates in data flows within and outside the system are the following:



*Figure 14 Instantiation of Software Architectural Components*

## High Level Testing Goals

The application will undergo high level testing which is popularly known as Black-box testing techniques. Black-box testing is a method of software testing that examines the functionality of an application without going deeper into its internal structures or workings. In this phase, all the higher-level components of the application will be tested to ensure that the application meets its basic requirements.

Goals of the higher-level testing can be described as follows:

- To demonstrate that the application meets its requirements
- To ensure that it produces valid and appropriate result/output
- To find any existing bugs or faults, design issues etc.
- For ensuring that the app runs smoothly and works perfectly
- For improving user experience

Test ID	Purpose	Input	Expected Output
1	To see if areas can be selected in the UI	Click on <b>Select Area</b> dropdown on the UI	A list of selectable areas should show
2	Check if day selection dropdown is working	Click on <b>Select Day</b> dropdown on the UI	A list of valid days should show
3	Check functionality	Provide all the valid inputs	Textual and Graphical Output should show
4	Close application/ Application is not in a hang state	Click on <b>Close Application</b> or the close button on the window	The application should close and stop all activity

After the completion stage of the project, other test-cases may appear and this list should be updated accordingly.

## Conclusion

Traffic flow analysis and prediction based on google map's image data is not going to be an easy one as there is no existing project out there. As there are no defined approach to predict traffic like this, the accuracy of this project is somewhat uncertain. The extreme traffic problem in Dhaka city needs to be analyzed and technological endeavor should be posed. This can be a little step towards this goal. Although there are lots of wonderful app or tools available out there but no specific tool for analysis or future traffic update for Bangladesh. Due to the limitation of data collection or complex and stochastic characteristics of traffic scenario this work is going to be very challenging for sure.

In future, I will try to expand this project by considering few more attributes that are especially relevant to Bangladesh perspective and try out different algorithms to provide better prediction result.