TRAFFIC FLOW ANALYSIS WITH PYTHON

BRIGHT WIREDU NUAKOH
10721072

December 19, 2022

Supervised By: Dr. Joseph K. Ansong
Department of Mathematics
School of Physical and Mathematical Sciences
University of Ghana, Legon

Outline

- Introduction
- Greenshield's Model
- Collecting Traffic Parameters On A Road Using python
- Relationship between speed and Density and Density and flow
- Conclusion

UNIVERSITY OF GHANA

Introduction

Introduction

- In mathematics and transportation engineering, traffic flow is the study of interactions between travelers including; pedestrians, cyclists, drivers, and their vehicles and infrastructure (including highways, signage, and traffic control devices)
- The aim is to understand and develop an optimal transport network with efficient movement of traffic and minimal traffic congestion problems.
- Traffic flow is also known as traffic streams

Types Of Traffic Flow

Traffic flow can be divided into two primary types namely;

- **Uninterrupted Flow**

UNIVERSITY OF GHANA

Types Of Traffic Flow

Traffic flow can be divided into two primary types namely;

- **Uninterrupted Flow**
  Flow regulated by vehicle-vehicle interactions and interactions between vehicles and the roadway.

Example

Vehicles traveling on an interstate highway are participating in uninterrupted flow

UNIVERSITY OF GHANA

Types Of Traffic Flow

Traffic flow can be divided into two primary types namely;

- **Uninterrupted Flow**
  Flow regulated by vehicle-vehicle interactions and interactions between vehicles and the roadway.

Example

Vehicles traveling on an interstate highway are participating in uninterrupted flow

- **Interrupted Flow**

UNIVERSITY OF GHANA

Types Of Traffic Flow

Traffic flow can be divided into two primary types namely;

- **Uninterrupted Flow**
  Flow regulated by vehicle-vehicle interactions and interactions between vehicles and the roadway.

Example

Vehicles traveling on an interstate highway are participating in uninterrupted flow

- **Interrupted Flow**
  Flow regulated by an external means, such as a traffic signal.

  Vehicle-vehicle interactions and vehicle-roadway interactions play a secondary role in defining the traffic flow at this level.

UNIVERSITY OF GHANA

Cont.

- Understanding what type of flow is occurring in a given situation will help you decide which analysis methods and descriptions are the most relevant.

- Our study was limited to uninterrupted flow; other traffic signals were not taken into account.

UNIVERSITY OF GHANA

Traffic Parameters

The main categories of traffic parameters are microscopic and macroscopic parameters.

UNIVERSITY OF GHANA

Traffic Parameters

The main categories of traffic parameters are microscopic and macroscopic parameters.

Microscopic Variables

- Microscopic variables consider the interaction of individual vehicles.

- These variable include parameters such as;
  1. Driver behavior
  2. Vehicle locations
  3. Distance headways
  4. Time headways
  5. Velocity
  6. Acceleration of individual vehicles

Macroscopic Parameters

Macroscopic Variables

Macroscopic variables consider the aggregate behavior of traffic flow. These variables translate the discrete nature of traffic to continuous variables.

- **Density** Is the number of vehicles per kilometer on a road at a time. We denote it as ρ. we can calculate density as

UNIVERSITY OF GHANA

Macroscopic Parameters

Macroscopic Variables

Macroscopic variables consider the aggregate behavior of traffic flow. These variables translate the discrete nature of traffic to continuous variables.

- **Density** Is the number of vehicles per kilometer on a road at a time. We denote it as ρ. we can calculate density as

$$\rho = \frac{n}{\Delta X}$$

UNIVERSITY OF GHANA

Macroscopic Parameters

Macroscopic Variables

Macroscopic variables consider the aggregate behavior of traffic flow. These variables translate the discrete nature of traffic to continuous variables.

- **Density** Is the number of vehicles per kilometer on a road at a time. We denote it as ρ. we can calculate density as

$$\rho = \frac{n}{\Delta X}$$

- **Speed And Mean Speed** For n numbers of cars, we calculate the mean speed for a location interval by averaging the speeds of all of the vehicles in this interval

$$\bar{u} = \frac{1}{n} \sum^{n} v_i$$

UNIVERSITY OF GHANA

Cont...

- **Flow Rate** The flow rate represents the number of vehicles that passes a certain cross-section of a road or highway per time unit. We denote the flow rate as q. For a time interval ΔT at any location x_0, the flow rate is calculated as:

$$q = \frac{m}{\Delta T}$$

UNIVERSITY OF GHANA

Cont...

- **Flow Rate** The flow rate represents the number of vehicles that passes a certain cross-section of a road or highway per time unit. We denote the flow rate as q. For a time interval ΔT at any location x_0, the flow rate is calculated as:

$$q = \frac{m}{\Delta T}$$

Flow rate can also be calculated as:

$$q = \frac{m}{\sum_{i=1}^{m} h_i} = \frac{1}{\bar{h}}$$

thus, flow rate can be directly calculated from the headways. The flow rate is expressed in vehicles per hour.

UNIVERSITY OF GHANA

Greenshield's Model

- The models by Greensheild(1934) is very well known and has been in application for quite sometime.
- Greenshield propose a linear relationship between speed and density and it is given as:

$$u = u_f - \left[\frac{u_f}{k_j}\right] k....(1)$$

where u is the space mean speed at density k, u_f is the mean free speed and k_j is the jam density

Greenshield's Model

- The models by Greensheild(1934) is very well known and has been in application for quite sometime.
- Greenshield propose a linear relationship between speed and density and it is given as:

$$u = u_f - \left[\frac{u_f}{k_j}\right] k ....(1)$$

where u is the space mean speed at density k, u_f is the mean free speed and k_j is the jam density

- On the basis of a linear speed-density relationship , Greenshields also derived the following parabolic equation for the speed-flow curve: $q = u_f \left[k - \frac{k^2}{k_j}\right] .....(2)$

UNIVERSITY OF GHANA

Greenshield's Model

- The models by Greensheild(1934) is very well known and has been in application for quite sometime.
- Greenshield propose a linear relationship between speed and density and it is given as:

$$u = u_f - \left[\frac{u_f}{k_j}\right] k \ldots.(1)$$

where u is the space mean speed at density k, u_f is the mean free speed and k_j is the jam density

- On the basis of a linear speed-density relationship , Greenshields also derived the following parabolic equation for the speed-flow curve: $q = u_f \left[k - \frac{k^2}{k_j}\right] \ldots..(2)$
- From (1) and (2) we derive flow as $q = uk$.

UNIVERSITY OF GHANA

Handling Video Object In Python

This chapter explains the semi-automated procedure used to gather basic traffic variables from a video we captured in Ghana.

UNIVERSITY OF GHANA

Handling Video Object In Python

This chapter explains the semi-automated procedure used to gather basic traffic variables from a video we captured in Ghana.

To handle our video object, we will heavily rely on a Python module called OpenCV. OpenCV includes libraries for contour detection, video capturing, drawing shapes on frames, and other functions that make it easier to identify blobs of vehicles from our video.

UNIVERSITY OF GHANA

Total Distance Covered By Camera In Meters

Figure 1: A physical length recorded manually

UNIVERSITY OF GHANA

- Total Distance of referenced points in figure 1 reads 11.89m denoted as PysicalCover

UNIVERSITY OF GHANA

- Total Distance of referenced points in figure 1 reads 11.89m denoted as PysicalCover

- The total distance covered by camera is calculated as;

$$TotalDistance = \frac{xmax}{PixCover} * PhysicalCover$$

UNIVERSITY OF GHANA

- Total Distance of referenced points in figure 1 reads 11.89m denoted as PysicalCover

- The total distance covered by camera is calculated as;

$$TotalDistance = \frac{xmax}{PixCover} * PhysicalCover$$

- According to this calculation, the total road that our camera covered was 27.18 meters, or 0.02718 kilometers.

Setting The frames for blob detection

Figure 2: Drawing Shapes On The Frames

UNIVERSITY OF GHANA

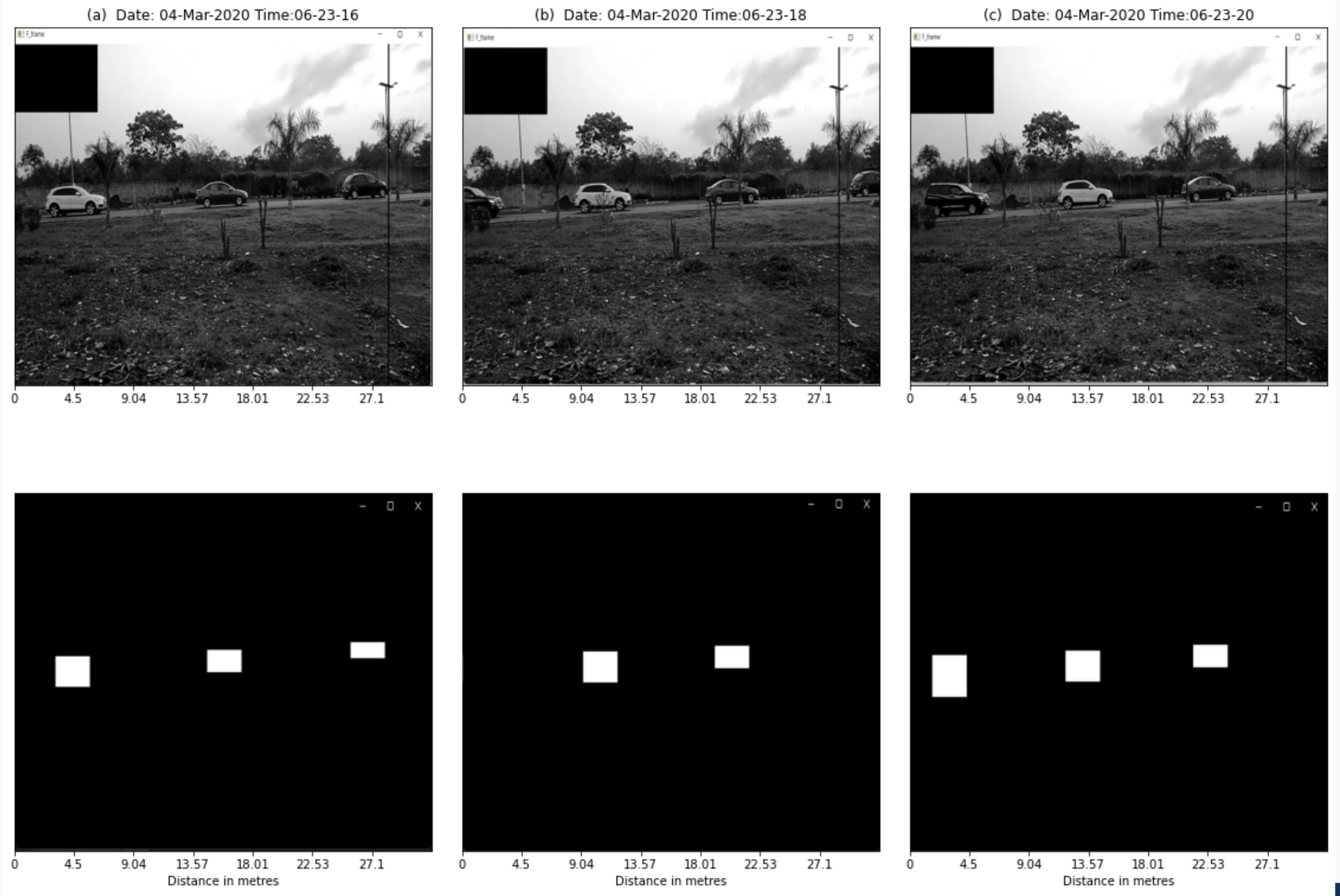

Figure 3: Detected Blobs Of Vehicles

UNIVERSITY OF GHANA

Getting The Details Of Individual Vehicles

- To remove any unwanted spots in the newly created video, cv2.GaussianBlur and cv2.threshold are applied on the video.
- cv2.connectedComponents is used to detect the area of bloobs as well as the x and y centriods of individual blobs of vehicle, which are then stored in a five dimensional array.
- The blob area, x centriod, and time are then used to sort the details of each unique individual vehicle.
- Finally, individual vehicle details are stored hierarchically based on the vehicle's time of entry into a new array, which will be used to calculate speed and density.

UNIVERSITY OF GHANA

Estimating Speed

- A vehicle's speed is the distance it travels per a unit of time.

UNIVERSITY OF GHANA

Estimating Speed

- A vehicle's speed is the distance it travels per a unit of time.

- Each car's x-centimeters will be plotted against the time that it was discovered as points. The speed of the car is determined by the slope of the line fitted into those points. Polyfit is a python module used in doing this.

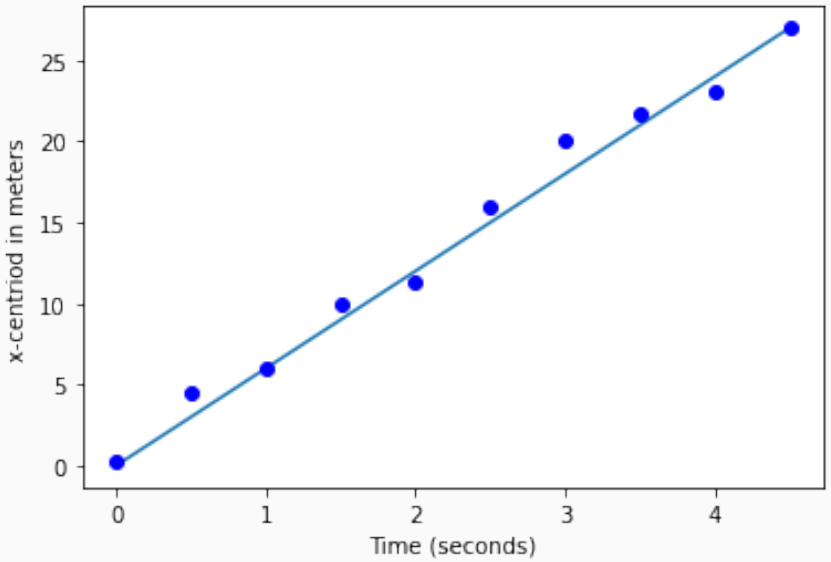

Figure 4: Speed Graph

UNIVERSITY OF GHANA

Trajectories Of Few Cars

- Trajectories of few detected cars

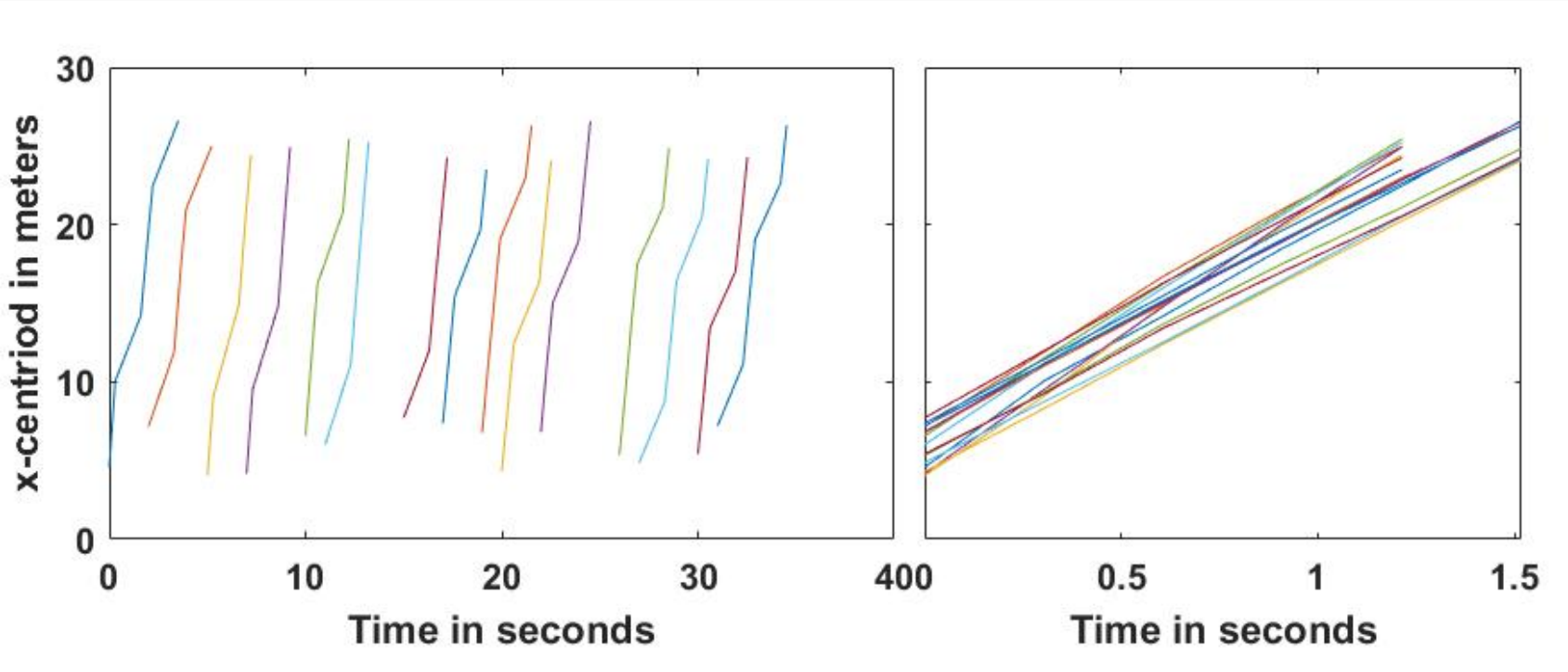

Figure 5: The Left hand side of this figure shows Space-Time graphs of a few different vehicles that illustrate where they are roughly starting and the point the leave the section of the road. On the right hand side depicts each vehicle's beginning point set to time 0

Estimating Density

- We compute the density at each time by dividing the number of cars on the road stretch by its total length of road section. i.e

$$\rho = \frac{NumberofCars}{0.02718}$$

UNIVERSITY OF GHANA

Estimating Density

- We compute the density at each time by dividing the number of cars on the road stretch by its total length of road section. i.e

$$\rho = \frac{NumberofCars}{0.02718}$$

- We determine the density of the road and the vehicle's speed whenever a new car is encountered on the road. The density at that location as well as the vehicle's speed are both recorded by our program.

UNIVERSITY OF GHANA

Speed-Density Relationship

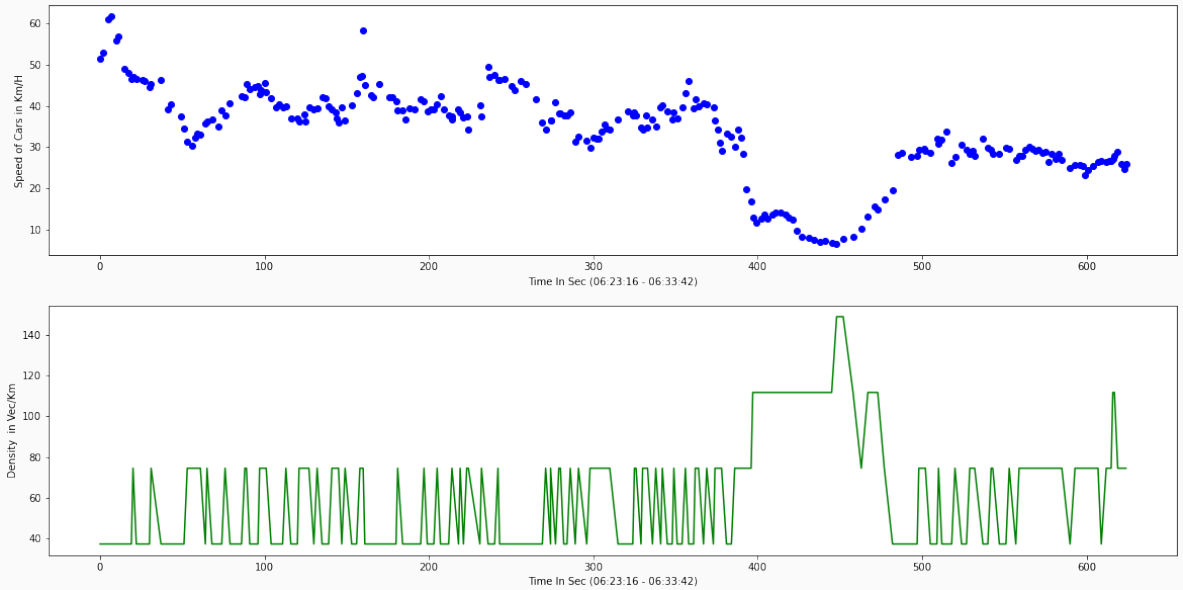

Figure 6: Speed Against Density

- Density goes up as Speed goes down
- As we have more cars on the road, their speed decrease
- Greenshield proposed that there is linear relationship between speed and density

UNIVERSITY OF GHANA

Traffic Flow Rate (Flux)

- Using the Greenshied's project we find flow by multiplying density by speeds at each time.

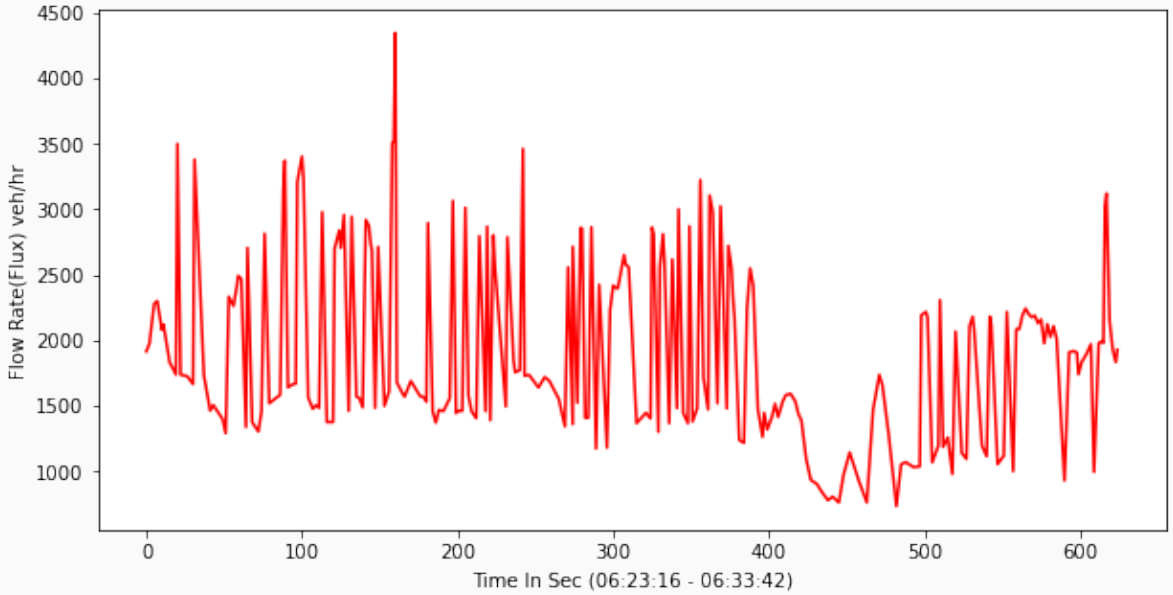

Figure 7: Flow Rate

- The average flow rate of the road under consideration is 1932 vehicles per hour from the time 06:23:16am to 06:33:42am.

UNIVERSITY OF GHANA

Conclusion

- Together with our Python program and the Greenshield's $q = uk$ model, we are able to determine traffic flow of the Legon Bypass road from the time 06:23:16am to 06:33:45am.
- The analysis is applicable on all other roads including highways to help solve traffic problems on road.
- The program developed can be used as a building block for other students interested in studying traffic flow with Python.
- In the future we seek to advance this dissertation by automatically detecting the blobs of vehicles in the captured video.

UNIVERSITY OF GHANA

Python Code For Reading Frames In videos Using OpenCV

```
100 def first_frame():
101     global rescale_frame,vid,rat,f_frame,xmin,ymin,xmax
    ,ymax,ycorner,xcorner,xend
102 
103     #rescalling the default frame size to diffrent size
104     f_frame = rescale_frame(f_frame, percent=75)
105 
106     i=0
107     while i<2:
108         if rat == True :
109             #converting the image to gray image
110             gray = cv2.cvtColor(f_frame, cv2.
    COLOR_BGR2GRAY)
111             cv2.imshow("F_frame",gray)
112             #get the action action of the mouse
113             cv2.setMouseCallback("F_frame", onMouse)
114             #posNp = np.array(posList)
115             if cv2.waitKey(25) & 0xFF ==ord('q'):
```

```
116             if cv2.waitKey(25) & 0xFF ==ord('q'):
117                 break
118         else:
119             i+=1
120     #finding the maximum position clicked in the x and
        y location
121     for i in posList:
122             if i[0] > xmax:
123                 xmax=i[0]
124             if i[1] >ymax:
125                 ymax =i[1]
126
127     xend= math.floor(xmax-(10/100)*xmax)
128     xcorner= math.floor((20/100)*xmax)
129     ycorner= math.floor((20/100)*ymax)
```

First_frame.py

UNIVERSITY OF GHANA

Code To Calculate The Distance Covered By Camera

```
100 def distance_covered_from_two_points():
101     PhysicalCover = 11.89
102     global vid,rat,f_frame,postList
103     width = vid. get(cv2. CAP_PROP_FRAME_WIDTH )
104     height = vid. get(cv2. CAP_PROP_FRAME_HEIGHT )
105     i=0
106     while i<2:
107         if rat == True :
108             gray = cv2.cvtColor(f_frame, cv2.
    COLOR_BGR2GRAY)
109             cv2.imshow("Total distance covered",gray)
110             cv2.setMouseCallback("Total distance
    covered", on_mouse)
111
112             if cv2.waitKey(25) & 0xFF ==ord('q'):
113                 break
114         else:
115             i+=1
```

```
116     try:
117         PixCover = points[1][0]-points[0][0]
118         metersPerPix= PhysicalCover/PixCover
119         TotDistance = (xmax/PixCover)*PhysicalCover
120         TotDistanceKM = TotDistance/1000
121         print('Total distance covered in Kilometers is:
    ',TotDistanceKM)
122     except:
123           print("Something went wrong")
124     return [TotDistanceKM,TotDistance,metersPerPix,
    PixCover]
```

Distance_covered_by_camera.py

Code To Detect Blobs In A Frame And Write New Video

```
100 def rectangle():
101     global hwidth,out
102     empty = 0*f_frame
103     t_msec = 1000*(7*60+29)
104     vid.set(cv2.CAP_PROP_POS_MSEC, t_msec)
105     rat, re_frame =vid.read()
106     re_frame = rescale_frame(re_frame, percent=75)
107     i=0
108     while i<3:
109         if rat == True :
110             gray = cv2.cvtColor(re_frame, cv2.
        COLOR_BGR2GRAY)
111             cv2.imshow("Click for Locations For Your
        Rectangles",gray)
112             cv2.setMouseCallback("Click for Locations
        For Your Rectangles", last_Onmouse)
113             if cv2.waitKey(25) & 0xFF ==ord('q'):
114                 break
115         else:
```

```
116             i+=1
117     xco = [], yco = []
118     for i in range(0,len(Locations)):
119         xco.append(Locations[i][0]-hwidth)
120         yco.append(Locations[i][1])
121     Xn=len(xco)
122     xdist = (2*hwidth)*np.ones(Xn)
123     height = heights[0:Xn]
124     for i in range(0,Xn):
125         rect = cv2.rectangle(empty,(xco[i],yco[i]),(int
    (xco[i]+xdist[i]),yco[i]+height[i]),(255,255,255)
    ,-1)
126     i=0
127     while i<2:
128         if rat == True :
129             cv2.imshow("rect",empty)
130
131             if cv2.waitKey(25) & 0xFF ==ord('q'):
132                 break
133         else:
134             i+=1
```

References

- Gary Bradski and Adrian Kaehler. Learning OpenCV: *Computer vision with the OpenCV library.* " O'Reilly Media, Inc.", 2008.
- LH Immers and S Logghe. "Traffic flow theory". In: *Faculty of Engineering,Department of Civil Engineering, Section Traffic and Infrastructure, Kasteelpark Arenberg* 40.21 (2002), pp. 887–891.
- Simon Narh Kemausour. "Traffic flow and traffic discontinuities on road: Madina-Okponglo highway and the Legon bypass road". In:*College Of Basic and Applied Sciences, Department of Mathematics, Traffic flow and traffic discontinuities on some principal streets of Accra: Madina-Okponglo highway and the Legon bypass road,* Simon N. Kemausour (September, 2020), pp. 59–61.

- CHINEDU OKAFOR. Top 5 African countries with the most vehicles on the road (Sub-Saharan). Sept. 2022. :https://africa.businessinsider.com/local/lifestyle/top-5-african-countries-with-the-most-vehicles-on-the-z2dmbj3.
- Axay S Shah et al. "SELECTION OF BEST SUITABLE TRAFFIC MODEL". In: *National Conference on Recent Trends in Engineering & Technology India*. 2011.

UNIVERSITY OF GHANA

UNIVERSITY OF GHANA