International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Design and Development of Traffic Monitoring System

Md. Manoar Imran[1], Lutfi Habiba[2]

[1]Department of computer Science and Engineering
[2]Institute of Science and Technology, National University, Bangladesh

ABSTRACT
Traffic monitoring is one area that utilizes Deep Learning for several purposes. By using cameras installed in some spots on the roads, many tasks such as vehicle counting, vehicle identification, traffic violation monitoring, vehicle speed monitoring, etc. can be realized. In this paper, we discuss a Deep Learning implementation to create a vehicle Classification system by track the vehicles movements. To enhance the system performance and to reduce time in deploying Deep Learning architecture, hence pre-trained model of YOLOv3 is used in this research due to its good performance and moderate computational time in object detection. This research aims to create a vehicle classification system to help human in classify and movement track the vehicles that cross the street. The counting is based on four types of vehicle, i.e. car, motorcycle, bus, and truck, while previous research counts the car only. As the result, my proposed system will capable to count the vehicles crossing the road based on video captured by camera with the highest accuracy of 97%.

Keywords: Deep Learning, YOLOv3, Classification

1. INTRODUCTION

Traffic monitoring system is one of the application areas of Computer vision technology. It can be used to monitor vehicle movement on the road and classify them into different categories. Object detection has crucial role in this part. Firstly, detect the necessary object from video and track it in the video canvas. Depending on the movement path we can take a decision about the journey of the detected object. Machine Learning approach needs preprocessing approach to complete this task. The inaccuracy of detection also occurs when some changes happen to the surface of the road, road repair, road damage, or any obstacles on the road, because those can disturb the image subtraction process. While Deep Learning (DL) approach gives more flexible performance without having to pre-process the image and extract the feature using several methods, even though it is computationally expensive, and it needs large amount of data to train the networks. In this work, we developed a vehicle classification and movement monitoring system using Deep Learning algorithm. Pertained YOLOv3 is used as the DL architecture that is well known with its good accuracy in object detection compared to other DL architectures. This work is motivated by previous research that was mostly tested using videos in the highways that are only passed by cars, bus, or truck, and there are no motorcycles. Besides, any buses or trucks are simply considered as ‘car’ without classifying them into more detail classes as ‘bus’ or ‘truck’ when counting. While some traffic monitoring system may need more detail information about the type of the vehicles whether it is a car, truck, bus, or motorcycle. Were also mostly tested in good traffic condition with good

driving manner, so that the counting gives accurate result. Hence, in this work we focus on developing a system that count the number of vehicles crossing the road where the counting is based on the type of the vehicle itself, i.e., car, bus, truck, and motorcycle based on Deep Learning algorithm with YOLOv3 architecture.

2. EXISTING SYSTEM

This research was conducted in several phases, starting with some literature reviews to explore the result from previous related studies in order to find what problems that should be solved in the current research, also to decide what methods should be implemented. The system is designed once the problem and the methods are clearly decided. As mentioned in Introduction section, YOLOv3 is used as the algorithm to detect the vehicles that cross the road. The performance of the system is measured by its accuracy in classifying the vehicles. It is compared to the real number of vehicles counted by human.

2.1 Existing System Architecture

The vehicle classification system has two main modules, i.e., Object Detection Module and tracking module as given in Figure 2.2 the first module reads every single frame from the video and doing vehicle detection using YOLOv3 algorithm. This module results the location of every detected vehicle, i.e., the bounding box coordinates. Then, the second module counts the number of vehicles that crossing the road based on the coordinates or location of the vehicles. So, the result of object detection module plays significant role in this system, because once the vehicle is not detected, then it will not be counted.

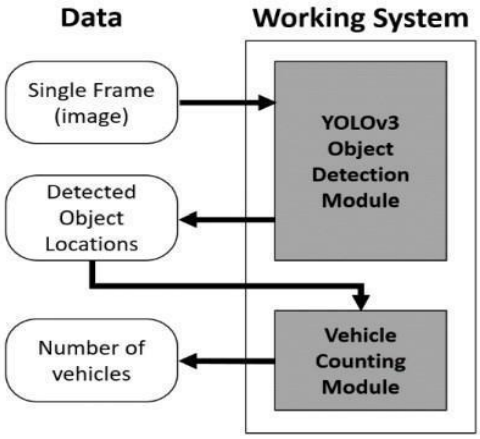

Figure 1: System Architecture

2.2 Performance Measurement

The performance of the system is measured by comparing the difference between real number of vehicles and number of tracked vehicles by the system. The percentage of accuracy is calculated using equation.

$$Accuracy = \left(1 - \frac{|RC - SC|}{RC}\right) \times 100\%$$

Where RC is real number of vehicles counted by human and SC is number of vehicles counted by the system.

2.3 Requirements

To build this project we need some hardware and software and a language to program those devices. Those lists are mentioned below

2.3.1 Hardware Requirements

- ✓ IP Camera
- ✓ Computer with NVidia GPU
- ✓ Software Requirements
- ✓ Python
- ✓ OpenCV
- ✓ YOLOv3
- ✓ TensorFlow
- ✓ Pycharm

2.3.2 Hardware Requirements

This section specifies the hardware requirements for a Project Server deployment based on the datasets defined in how datasets affect performance and capacity in Project

3. PROPOSED SYSTEM DESIGN

Design section is the section where anyone can easily understand the whole system or the part of it. It emphasis on translating design specification to performance specification is system design.

3.1 Block Diagram of Traffic Monitoring System

If we put all the components together and connect everything each other it will be look like the following figure:

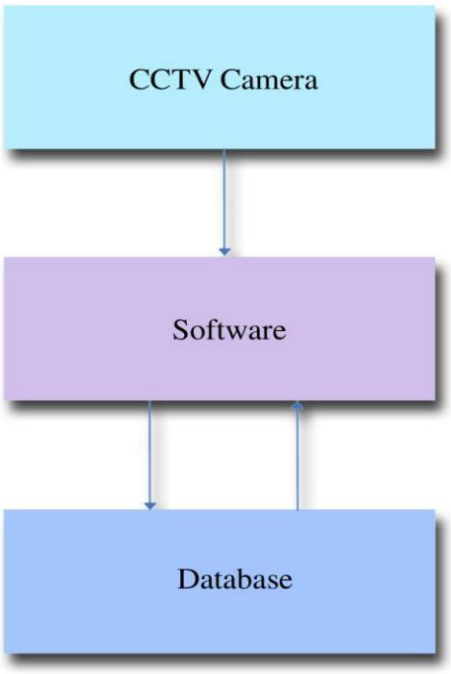

Figure 2: Block Diagram of Traffic Monitoring System

3.2 Flowchart of Traffic Monitoring System

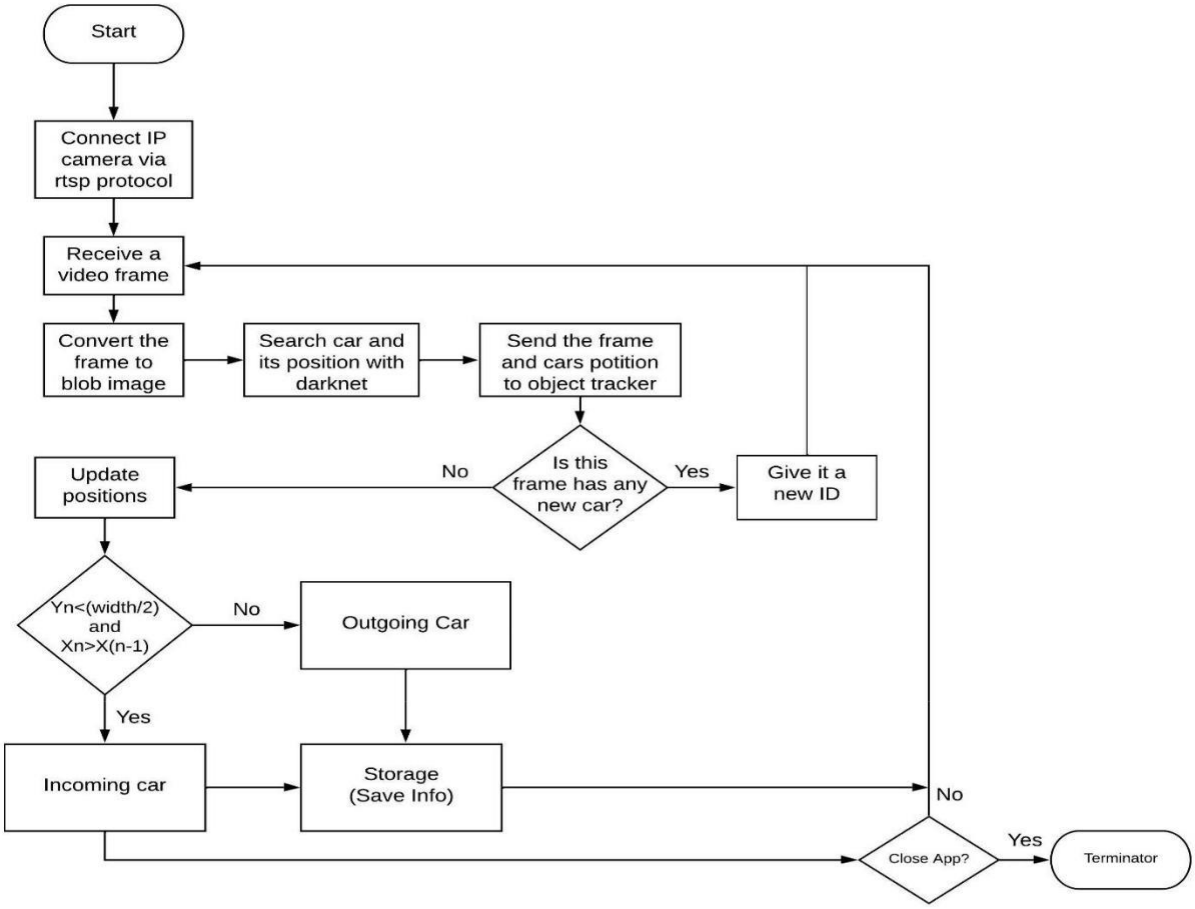

Figure 3: Flowchart of Traffic Monitoring System

3.4 Working procedure of whole system

In this project I will use OpenCV for image manipulation and YOLOv3 for object detection.My code will connect the IP Camera with rtsp protocol. I will use OpenCV to read frame by frame from camera.Each frame will be converted to blob image then sent to YOLOv3 object detector.YOLOv3 will detect cars and their type with its position, height and width. Then it will be passed to our tracker function and it will get an id number.If the same car is sent to the tracker another time, then it will update the object position without changing its id.Tracker will track it movement continuously.When the object will cross the finish line it will be counted a completed id.

4. PROPOSED SYSTEM IMPLEMENTATION

The project will be a software project that helps us to vehicle detection and movement tracking.

4.1 Environment Setup

- Download and install python
- Download and install pycharm
- Download yolo v3 weight and cfg file
- Install TensorFlow ,OpenCV and numpy

4.2 Vehicle detection

For vehicle detection I've used yolo v3. It's an object detection algorithm. Its pre-trained model comes with 80 different type of classes. But we have used Person, Car, Truck, Bus, and Motorbike for our project.

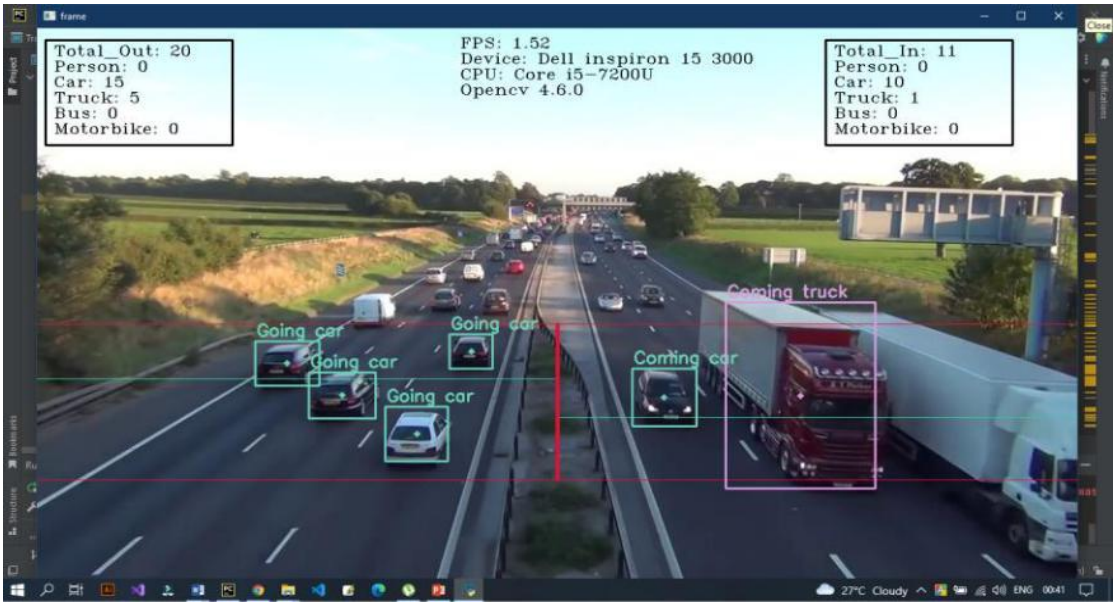

Figure 4: Vehicle detection

4.3 Motion tracking

Here used deepsort for motion tracking. When a new car is detected by yolo v3 then a unique id generated. Then in the next frame deepsort match all object with previous frame object. If any match found and it previous object.

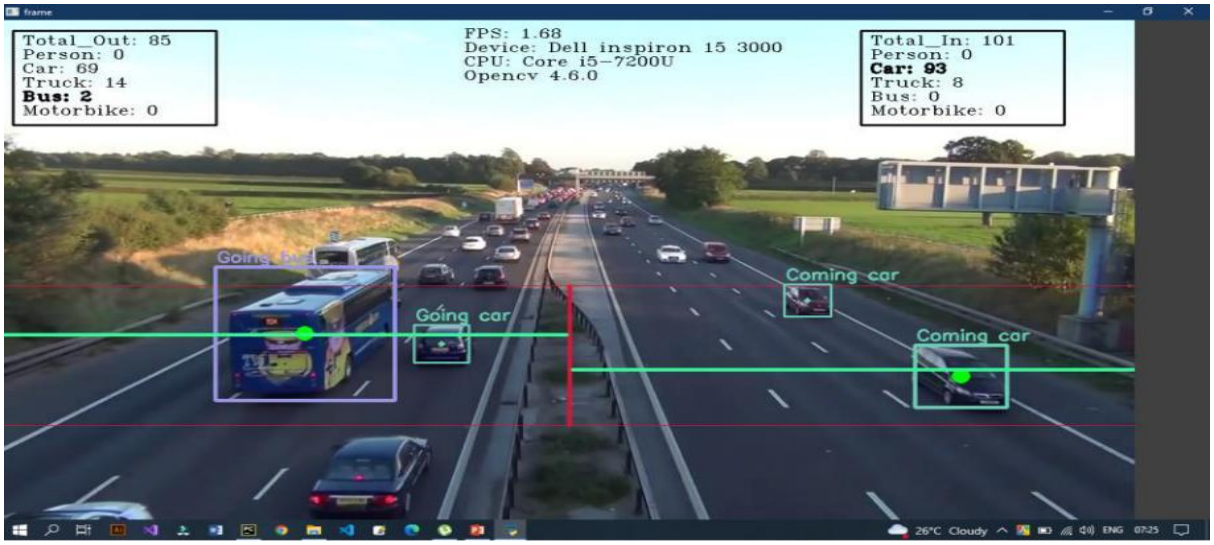

Figure 5: Motion tracking

4.4 TensorFlow tracking

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. TensorFlow can be used in a wide variety of programming languages, including Python,

JavaScript, C++, and Java. This flexibility lends itself to a range of applications in many different sectors.

TensorFlow.nn is a module for executing primitive neural network operations on models. Some of these operations include variations of convolutions, activation functions (Softmax, RELU, GELU, Sigmoid, etc.) and their variations, and other Tensor operations (max-pooling, bias-add, etc.)

4.5 OpenCV

OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.

4.6 Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

5. FUTURE WORK

The project will be a software project that helps us to vehicle detection and movement tracking.

- Speed measurement of cars
- Wrong side entry
- Lan violation
- License plate recognition

6. CONCLUSION

Traffic Monitoring System for Monitor vehicle movement Classify them into categories and Count incoming and outgoing cars.This system will help to monitor overall traffic of a street

7. REFERENCE

1. A Sport Athlete Object Tracking Based on Deep Sort and Yolo V4 in Case of Camera Movement | IEEE Conference Publication | IEEE Xplore.
2. D Comaniciu and P Meer, "Mean shift: A robust approach toward feature space analysis [J]", IEEE Transactions on pattern analysis and machine intelligence.
3. R. Van Der Merwe, A. Doucet, N. De Freitas and E. A. Wan, "The unscented particle filter", *Advances in neural information processing systems*, pp. 584-590, 2001
4. R. E. Kalman, *A new approach to linear filtering and prediction problems*, 1960
5. M. Keuper, S. Tang, Y. Zhongjie, B. Andres, T. Brox and B. Schiele, *A multi-cut formulation for joint segmentation and tracking of multiple objects*, 2016.
6. B. Lee, E. Erdenee, S. Jin, M. Y. Nam, Y. G. Jung and P. K. Rhee, "Multi-class multi-object tracking using changing point detection", *ECCV*, pp. 68-83, 2016.