DhakaTrafficDetection

by Mohammad Navid Nayyem

Submission date: 04-Feb-2021 07:13PM (UTC+0900)
Submission ID: 1501409294
File name: Dhaka_Traffic_Detection.docx (79.51K)
Word count: 2253
Character count: 12609

Group - 03

Capstone Project Proposal

Dhaka Traffic Detection

Submitted to:

Supervisor Name: Mohammad Rifat Ahmed Rashid

Assistant Professor, CSE Department, ULAB

Prepared by

Student's Name	ID
Mohammad Navid Nayyem	172014003
KH. Nafiu Nur Rashid	172014026
Fabliha Bushra	163014029

Course Title: Project

Course Code: CSE499

Group: 03

Date of Submission: 17th January 2020

Table of Contents

Table of Figures

1. Brief Abstract

Dhaka is the most densely populated city in the world, is the capital city of Bangladesh. The traffic jam of this city is a big problem and sometimes it seems that this city is cursed with traffic jam. The traffic jam of Dhaka is very complicated and it is a new complex challenge in terms of automated traffic detection. This problem can be solved using Artificial Intelligence based technology. The detection of an object is associated with computer vision. It describes a system that can detect the location of a desired object in an image. Computer vision works with digital images and videos to understand the contents or objects presented in the images or videos. Rectangular or square bounding box can be used to detect and count vehicles from the images or videos according to vehicles categories by estimated distance from the video recording of closed-circuit television (CCTV) camera. On the other hand, if the vehicles on the road can be counted, it will be very much beneficial for maintaining many systems such as managing traffic, controlling traffic and also a better parking management system can be made. A strong dataset of images needs to be created for this.

2. Problem Statement

This app-based system will basically use for maintaining traffic in Bangladesh. The traffic jam can be reduced as well as the savings of times will be increased. This is the main motto of developing this system. The objectives of our projects are: Vehicle Images Annotation, create dataset using annotated image with xml file, train a model using dataset, create an Application Programming Interface (API), develop a fully functioning Android app to detect the vehicle classes perfectly from video or images, easy to use, the wastage of time will reduce as the traffic jam will be controlled, provide good service to the users, reduce the difficulty to manage traffic and beneficial for better parking management system.

3. Aim and Objective

- Vehicle Images Annotation.
- Create dataset using annotated image with xml file.
- Train a model using dataset.
- Create an Application Programming Interface (API).
- Develop a fully functioning Android app to detect the vehicle classes perfectly from video or images.
- Easy to use.
- The wastage of time will reduce as the traffic jam will be controlled.
- Provide good service to the users.
- Reduce the difficulty to manage traffic.
- Beneficial for better parking management system.

4. Motivation

The Bangladesh people faces too many difficulties in road while going anywhere for their needs and they get frustrated most of the time because of this traffic jam in Bangladesh as there are too many vehicles out there in the road. By using Artificial Intelligence, we will try to solve the critical condition of the traffic of Bangladesh in future and try to control the traffic jam of Bangladesh. If we can do this, it will be beneficial for Bangladesh people as they want a nice solution for the traffic system. If the traffic jam can be controlled then the valuable times of people can be saved.

5. Brief Review of the Literature

Big data traffic analysis is a vital component of smart city. Public transit systems of future generation can be safer and smarter with millions of street cameras and intelligent algorithms. In the paper **"STREET OBJECT DETECTION / TRACKING FOR AI CITY TRAFFIC ANALYSIS"** the authors participated in the **IEEE Smart World 2017 NVIDIA AI City Challenge** which consists of **two** tracks of contests. The first contest **'AI City Track 1'** was on

visual detection, they built street object detector for vehicle and person to classify properly. The first contest **'AI City Track 2'** was on transportation applications, they developed a traffic analysis framework based on vehicle tracking that can assist the surveillance and visualization of the traffic flow. Both developed methods demonstrated practical and competitive performance on real-world traffic videos provided in the challenge contest. For First Track challenge, the authors combined two state-of-the-art object detection models, **R-CNN and ResNet**, to build a faster accuracy object detector. For Second Track challenge, they combined their object detector with a hypergraph based Multi-Object Tracking (MOT) method. They developed an efficient traffic analysis approach to analyze traffic flow patterns on the real-world traffic videos. The datasets used to train the vehicle detection module is included the AI City dataset and other standard datasets. The NVIDIA AI City (AIC) dataset consists of **three** subset of traffic videos taken from three different United States locations with different video resolutions. The locations are Silicon Valley intersection, Virginia Beach intersection, and Lincoln, Nebraska. Videos are recorded under different lighting conditions from day and night time. The labels for the datasets are car, SUV, small truck, medium truck, large truck, pedestrian, bus, van, group of people, bicycle, motorcycle, traffic signal-green, traffic signal-red and traffic signal-yellow. From 80 hours videos, about 150,000 key frames are manually annotated with bounding boxes around the objects of interest with corresponding labels. For the object detection task, the AIC dataset is divided into three subsets according to video resolution: The AIC480 dataset contains videos with 720x480 pixel resolution, the AIC1080 dataset contains videos with 1920x1080 pixel resolution and the AIC540 dataset is obtained by spatially down sampling frames from AIC1080. First Track objection detection and classification results are evaluated using three measures: **the F1-score, mean Average Precision (mAP) and the Intersection over Union (IoU).** The objective of First Track of AI City Challenge was to detect and classify all street objects from videos. The modern convolutional neural networks for object detection including the Faster **R-CNN, R-FCN, SSD** and **YOLO** which achieves remarkable performance in both accuracy and running time. The accuracy for modern object detectors they choose the Faster **R-CNN and ResNet101 as the basic model** for street object detector. In this paper, a practical system for street object detection, tracking and traffic analysis was described. The proposed approach will provide a solution for smart transportation, street surveillance, traffic safety, and can ultimately lead to a smarter city. **[1]**

In computer vision field, object detection plays a vital role. Object detection means locating/identifying objects in frame of video sequence. The main application areas of object detection: Motion based recognition, automated surveillance, video indexing, human-computer interaction, traffic monitoring, vehicle navigation and etc. Most of the object detection mechanism used information from single frame for detecting an object. But some of the object detection mechanism used temporal information which is computed from sequence of frames. The first step of real time object detection is to identify the region of interest of the video. Some of the object detection methods are: Point Detector, Background Modelling, Segmentation, Optical Flow and Supervised Classifier. Background modelling has two main approaches. One is Recursive algorithm and another one is Non-recursive algorithm. **Object Detection Algorithms:** An accuracy of detecting object in background subtraction based object detection algorithms, An optical flow based object detection algorithms and frame differencing based object detection algorithm. **BACKGROUND SUBTRACTION BASED MOTION DETECTION AND AGENT BASED TRACKING:** A Scene feature based algorithm was proposed in for detecting counter flow in security related surveillance in airport. They addressed the two main problems: 1. Most of the cameras deployed in security surveillance networks have poor resolution. It will create negative effects on tracking algorithm. 2. 24/7 basis operation of automatic video analytics algorithm some time will provide higher false positive rate in tracking. To avoid such problems they used novel classifier to identify scene feature in the image and KLT optical flow tracking algorithm. **BACKGROUND SUBTRACTION BASED OBJECT DETECTION USING CANNY EDGE OPERATOR:** When the number of discretely moving object increases, the understanding of video scene becomes more difficult. Motion feature (viz location, scale, score (magnitude), direction and velocity) filtering based event detection was proposed in to detect event in crowded area. When the moving objects are relatively fast to frame rate, the detection can be especially difficult. To overcome these problem, Manisha Chaple, proposed background subtraction based detection and optical flow algorithm for tracking in. And also, they used centroid in frame to detect the distance and velocity of the moving object. They established statistical based reliable background model and used dynamic optimization threshold method to detect moving object. In order to eliminate noise and eliminate the background disturbance problem, they applied dilation and erosion processing. Due to occlusion, lightning changes and other factor, abandoned object detection in complex video surveillance is too difficult. Yingli Tian, modeled three Gaussian

mixture-based Background subtraction. Among upper all of these, optical flow-based detection provides more accuracy than other techniques. **[2]**

We live in era of technology where we moving towards making a smart city. A smart city with a smart AI based traffic monitoring system. The author of this paper called "**Automated AI Based Road Traffic Accident Alert** "they are proposing a more advance traffic monitoring system which can identify and detect moving objects like cars, bikes etc. in live camera feeds. We make an **advance Artificial intelligence**-based system can detect occurrence of accident and alert to nearby hospitals/ambulance or Traffic policemen in real-time. Their system is based on **Neural Network** and **Deep Learning** of object detection along computer vision technology and several methods and algorithms**.** They **detect**, **classify**, **track** and compute moving object **velocity** using **convolution neural network (CNC).** Accident detection system trained by using **Regression** base algorithm called **Yolo** algorithm. In this system first they create their own dataset designing CNN. **RCNN** is used to extract region by applying selective search. The method is named RCNN because Regional proposals are combined with CNNs. Here two basic concepts are combined and applied in the R-CNN. The first concept is to apply an efficient convolutional neural network from bottom to up region proposals to locate and the second concept is to apply supervised training for field-specific tuning task when insufficient training images in entered into the system, resulting in significant improvement of performance. They use CNN (Convolutional Neural Network) for Feature extraction. Take an image as input **extract region proposal** using **R-CNN**. After attribute calculation using CNN, they classify region using. Here **Caffe Deep** Learning framework was used to carry out this training. **YOLO (you only look once)** algorithm achieves its result by applying a neural network on an image. **[3]**

6. Gantt Chart

Tasks	Start Date	End Date	Duration (Days)
1. Planning			
1.1. Project Proposal	7-Nov	9-Nov	2
1.2. Project Summary	10-Nov	12-Nov	2
1.3. Collect Picture for Annotation	13-Nov	28-Nov	15
2. Research			
2.1. LabelImg	29-Nov	30-Nov	1
2.2. Yolov5	1-Dec	3-Dec	2
2.3. Dataset	4-Dec	9-Dec	5
3. Implementation (Part - I)			
3.1. Model Train	10-Dec	9-Jan	30
4. Design			
4.1. Android Application Design	10-Jan	16-Jan	6
4.2. API Design	17-Jan	10-Feb	24
5. Implementation (Part - II)			
5.1. API Implementation	11-Feb	24-Feb	13
5.2. Android App Development	25-Feb	16-Apr	50
6. Testing Phase			
6.1. Android Application Testing	17-Apr	19-Apr	2
7. Follow-up			
7.1. Presentation & Final Defense	20-Apr	20-Apr	0
7.2. Final Report	21-Apr	24-Apr	3

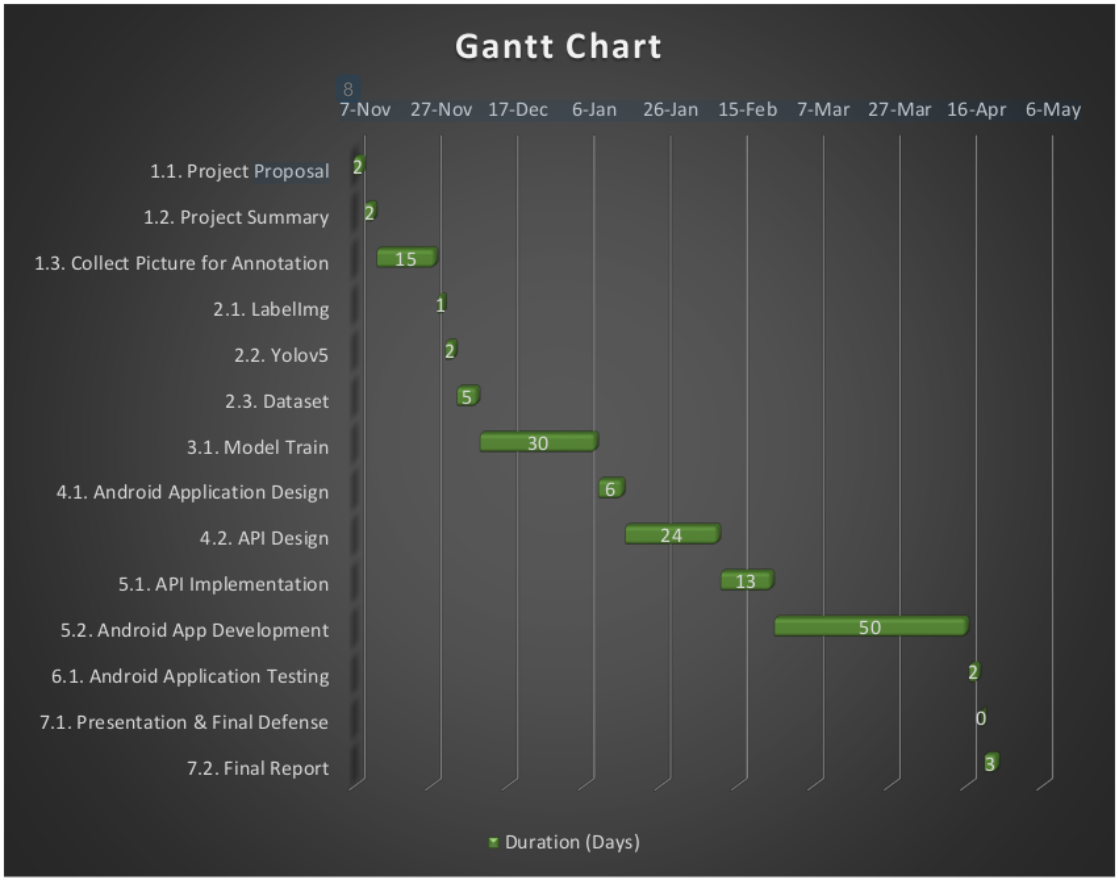

Figure 1: *Detailed Timeline of Each Task and when it will be completed (Gantt Chart)*

7. Description of the Final Product or Products to be Completed

The outcome of our project will be an Android-Based Application which will detect different categories of vehicles and try to recognize it from the videos or images and also count the number of vehicles. It will be beneficial for maintaining the traffic jam of Bangladesh in future. A statistical graph can be generated from the video recording of how many vehicles passes through any specific road in Bangladesh for the last 24 hours. By using the results from the graph, we can measure the percentages of specific vehicles travelled through any route and find out which route is mostly remained busy for a certain period of time. If it is implemented correctly then there will be less need of traffic police for controlling the traffic system in Bangladesh. We expect to share the results of our project in an exhibition.

For our project, we trained our dataset by using YOLOv5. YOLO is, it a real-time object detection framework and stands for "You Only Look Once". Meaning the image is only passed once through the FCNN or fully convolutional neural network.

Instead of making prediction on many regions of an image, YOLO passes the entire image at once into a CNN. The CNN that predicts the labels, bounding boxes and confidence probabilities for objects into an image.

YOLO steps:

1. Divide the image into cells with an **S x S** grid.
2. Each cell predicts **B** bounding boxes.
3. Return **bounding boxes** above confidence threshold.

8. Reference List

(i) Albany.edu. 2021. *STREET OBJECT DETECTION / TRACKING FOR AI CITY TRAFFIC ANALYSIS*. [online] Available at: <https://www.albany.edu/faculty/mchang2/files/2017_09_AICity_Challenge_Team5.pdf> [Accessed 16 January 2021].

(ii) *Researchgate.net*. [Online]. Available: https://www.researchgate.net/publication/274569172_A_survey_on_Real_time_Object_Detection_and_Tracking_Algorithms. [Accessed: 16-Jan-2021].

(iii) Ijstr.org. 2021. *Automated AI Based Road Traffic Accident Alert System: YOLO Algorithm*. [online] Available at: <https://www.ijstr.org/final-print/aug2019/Automated-Ai-Based-Road-Traffic-Accident-Alert-System-Yolo-Algorithm.pdf> [Accessed 16 January 2021].

DhakaTrafficDetection

ORIGINALITY REPORT

31%	28%	18%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.albany.edu Internet Source	14%
2	www.ijert.org Internet Source	3%
3	datax.berkeley.edu Internet Source	2%
4	www.ijstr.org Internet Source	2%
5	Submitted to University of Northampton Student Paper	2%
6	Submitted to St Joseph Engineering College, Mangalore Student Paper	1%
7	Submitted to Higher Education Commission Pakistan Student Paper	1%
8	Submitted to Buckinghamshire Chilterns University College Student Paper	1%

9	smart-city-sjsu.net Internet Source	1%
10	Unnikrishnan Kizhakkemadam Sreekumar, Revathy Devaraj, Qi Li, Kaikai Liu. "Real-Time Traffic Pattern Collection and Analysis Model for Intelligent Traffic Intersection", 2018 IEEE International Conference on Edge Computing (EDGE), 2018 Publication	1%
11	Submitted to Intercollege Student Paper	<1%
12	Submitted to Savitribai Phule Pune University Student Paper	<1%
13	www.google.com Internet Source	<1%
14	Aleksandr Fedorov, Kseniia Nikolskaia, Sergey Ivanov, Vladimir Shepelev, Alexey Minbaleev. "Traffic flow estimation with data from a video surveillance camera", Journal of Big Data, 2019 Publication	<1%
15	psasir.upm.edu.my Internet Source	<1%
16	tom-stehule.com Internet Source	<1%
17	ijicic.org	

Internet Source

<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		