DHAKA TRAFFIC DETECTION

COURSE CODE: CSE499

COURSE TITLE: CAPSTONE PROJECT

NAME OF OUR GROUP MEMBERS

- 1. Fabliha Bushra

  ID: 163014029

- 2. Mohammad Navid Nayyem

  ID: 172014003

- 3. KH. Nafiu Nur Rashid

  ID: 172014026

OUTLINES

- Introduction
- Motivation
- Classes of Vehicles
- Dataset Class Balance
- Software and Hardware Requirements
- Methodology
- Training and Validation Results
- Application Specification
- User Interface of Android App
- Vehicles Detection Output
- Vehicles Detection from Android App
- System Attributes
- Diagrams (Use Case, Context Level Dataflow, Level 1 Dataflow)
- Cost Analysis
- Scope for Future Work
- Gantt Chart

INTRODUCTION

- Dhaka is the most densely populated city in the world.
- Traffic jam problem can solve using Artificial Intelligence-based technology.
- Automated vehicles detection.
- Detect different types of vehicles from the images or videos.
- Helpful for maintaining a better parking management system.

MOTIVATION

- Transport is an important part of our day-to-day life.
- Bangladeshi people gets frustrated because of this traffic jam.
- **Dhaka Traffic Detection** is an Android application.
- Better solution for the traffic system.

CLASSES OF VEHICLE

- Total Classes: 21

DATASET CLASS BALANCE

Range	Colour	Description
0-350		Under represented
351-850		Average
850 >		Over represented

Figure 01

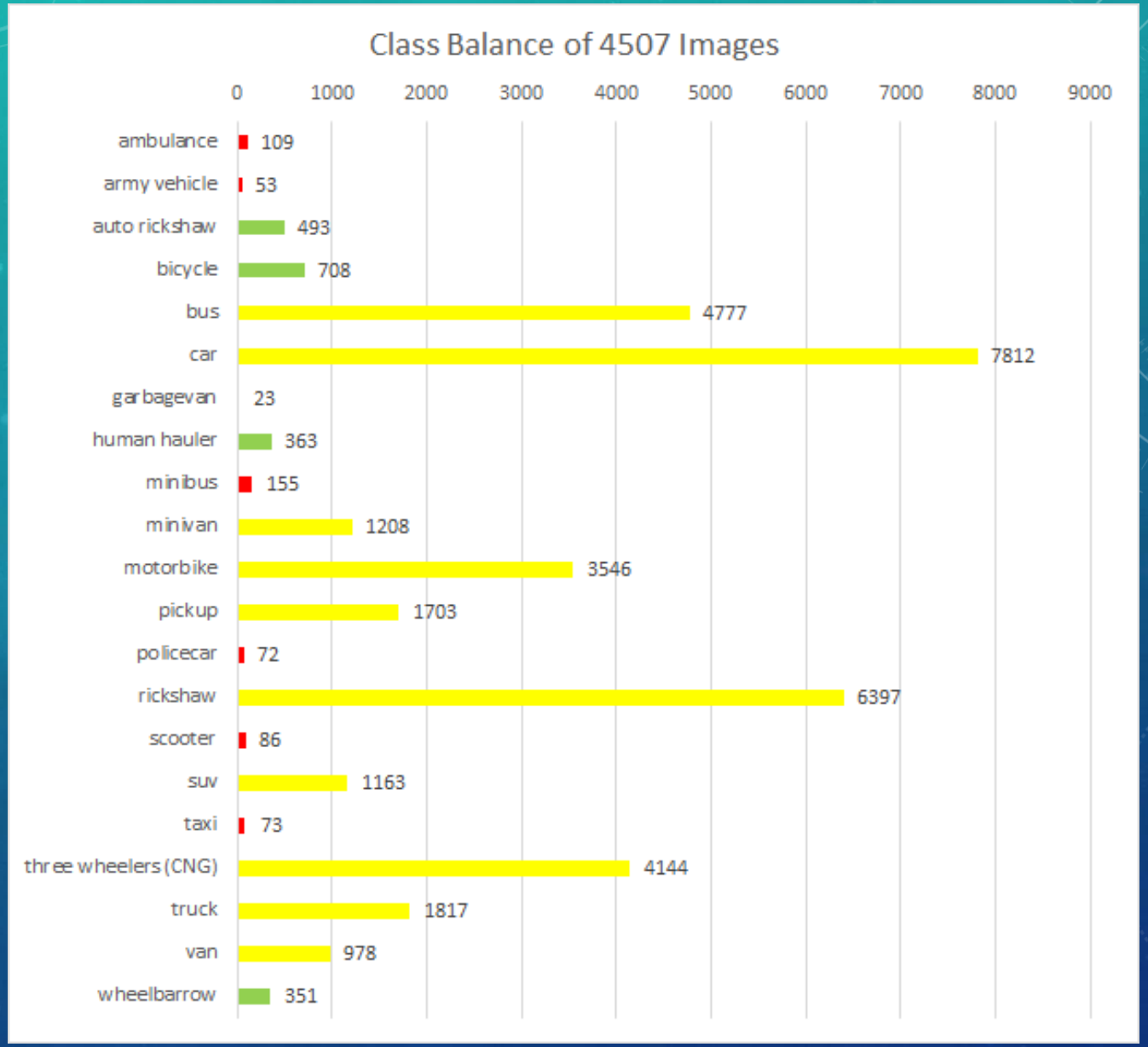


Figure 02: Sample statistics of 4507 images

SOFTWARE AND HARDWARE REQUIREMENTS

- **Software Requirements(for user)**

  (i) Anaconda Prompt

  (ii) Python v3.8.3

  (iii) Android Studio v4.1.2

  (iv) Git Bash

  (v) Kaggle Notebook

  (vi) Adobe XD

- **Hardware Requirements(for user)**

  (i) Android Smartphone with Minimum Android 11.0 Supported

METHODOLOGY

Architecture of YOLOv5

- **Model backbone**
  - Extract important features from the given input image.
- **Model Neck**
  - Generates features pyramids that helps to identify the same object with different sizes and scales.
- **Model Head**
  - Applies anchor boxes on features and generates final output vectors with class probabilities, object ness scores and bounding boxes.

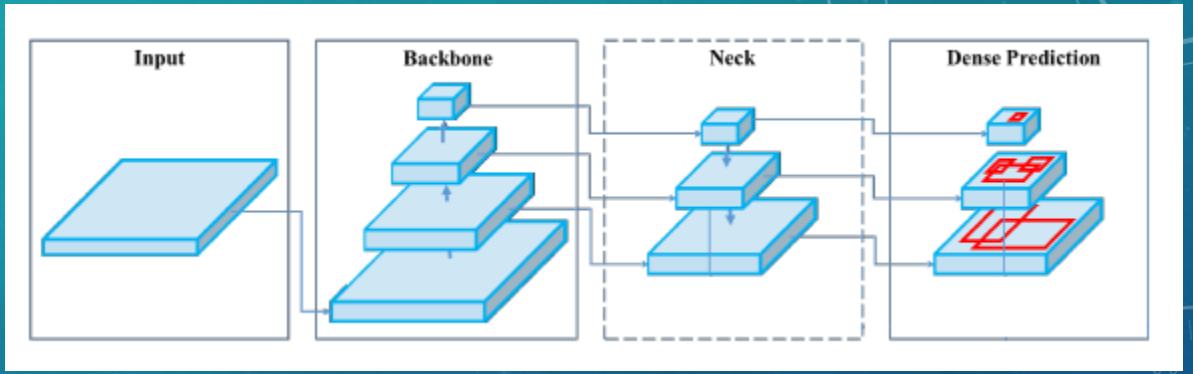


Figure: Object Detection Process

Other Aspects of YOLOv5

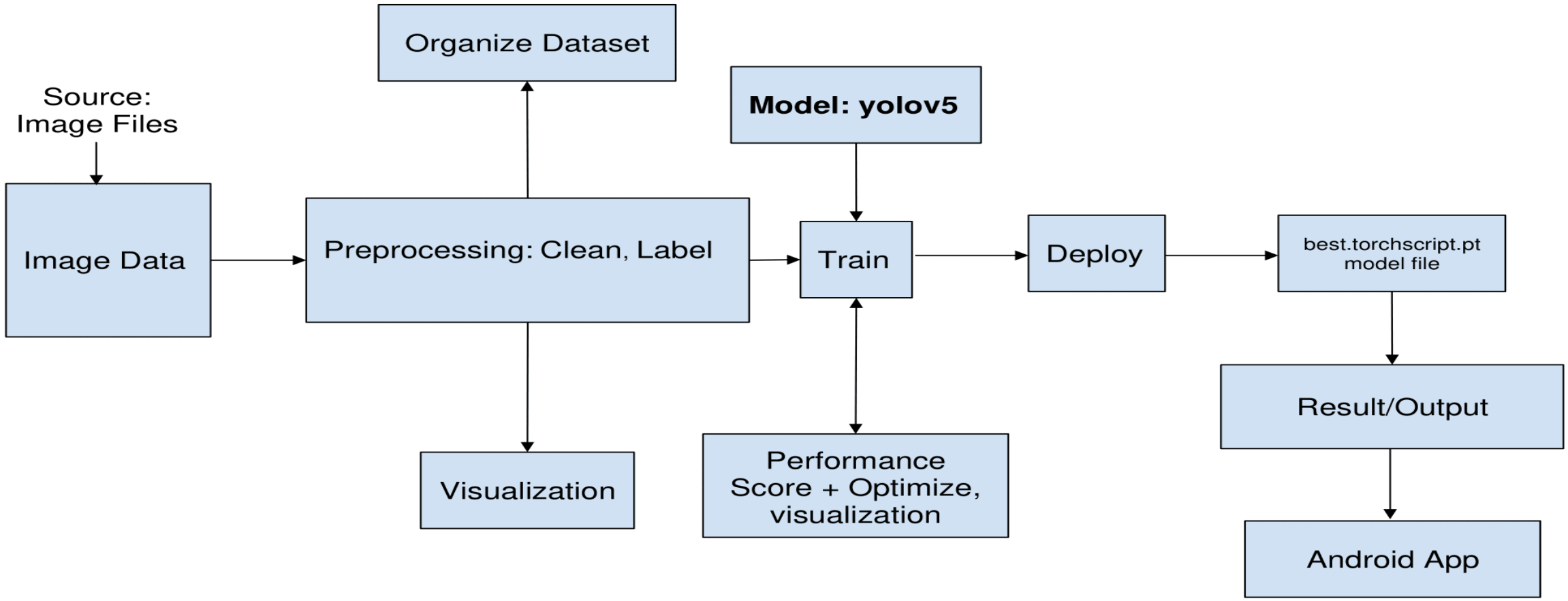
Augmentation Used: Mosaic

Bounding Box Anchor: Learned from dataset automatically with K-means and genetic learning algorithm

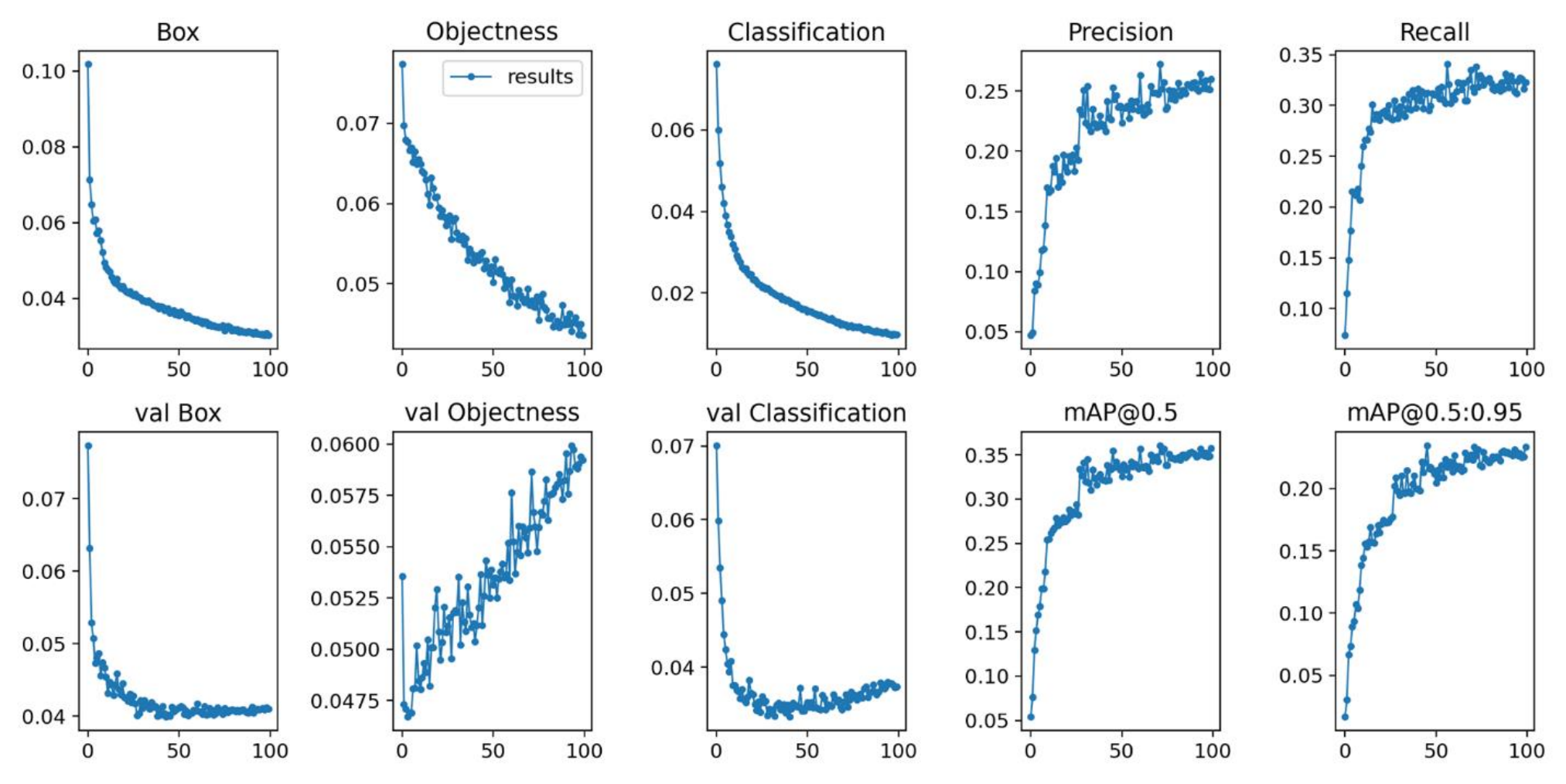
Reference: Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

METHODOLOGY (CONT.)

Machine Learning Pipeline



TRAINING AND VALIDATION RESULTS



APPLICATION SPECIFICATION

- Android-Based Application.
- Detect different categories of vehicles from videos or images.
- **Four different types of operations:**

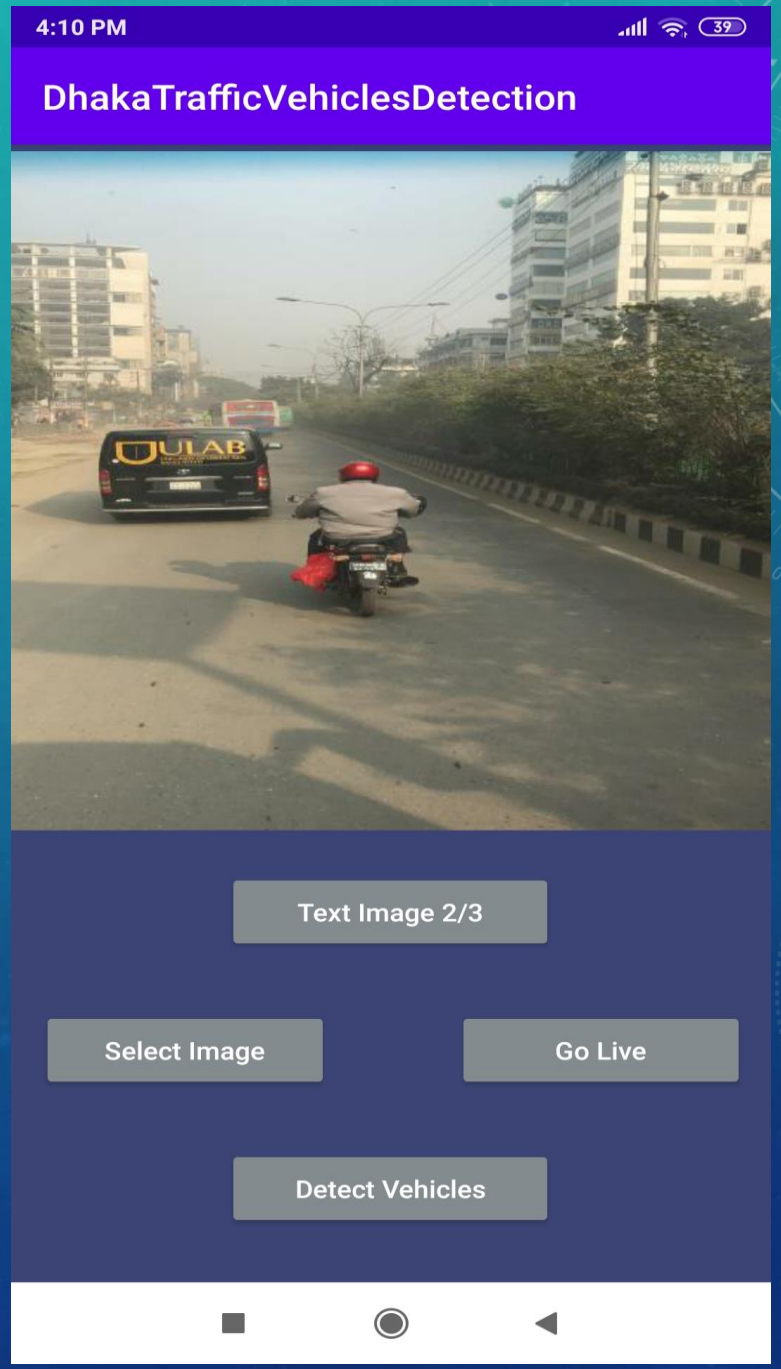
  (i) Detect the vehicles from three built in test images.

  (ii) Choose images from android phone’s gallery and detect vehicles from them.

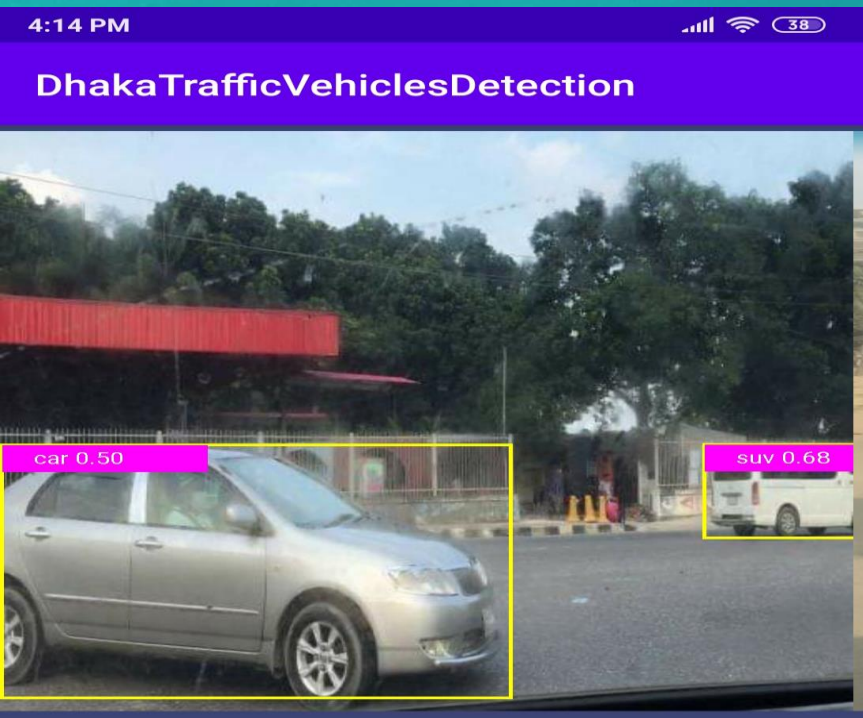
  (iii) Detect vehicles from image or live video.

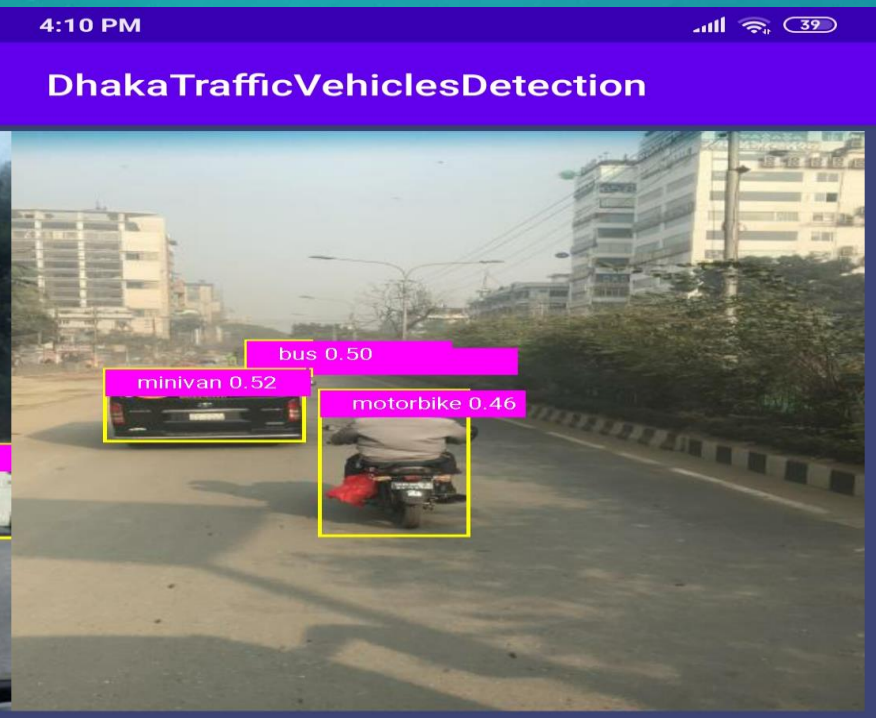
  (iv) Detect Vehicle Operation.

USER INTERFACE OF ANDROID APP

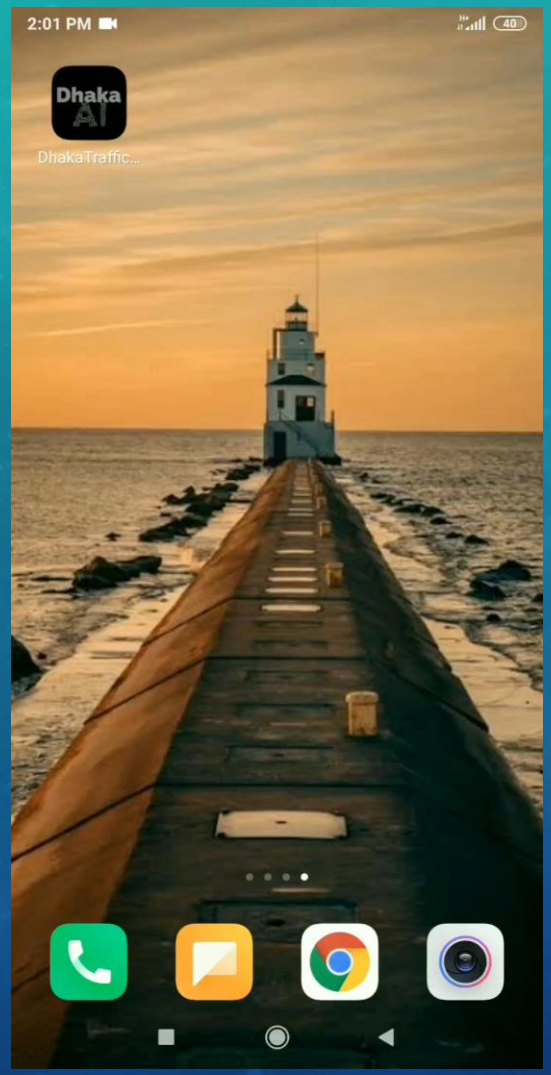


VEHICLES DETECTION OUTPUT





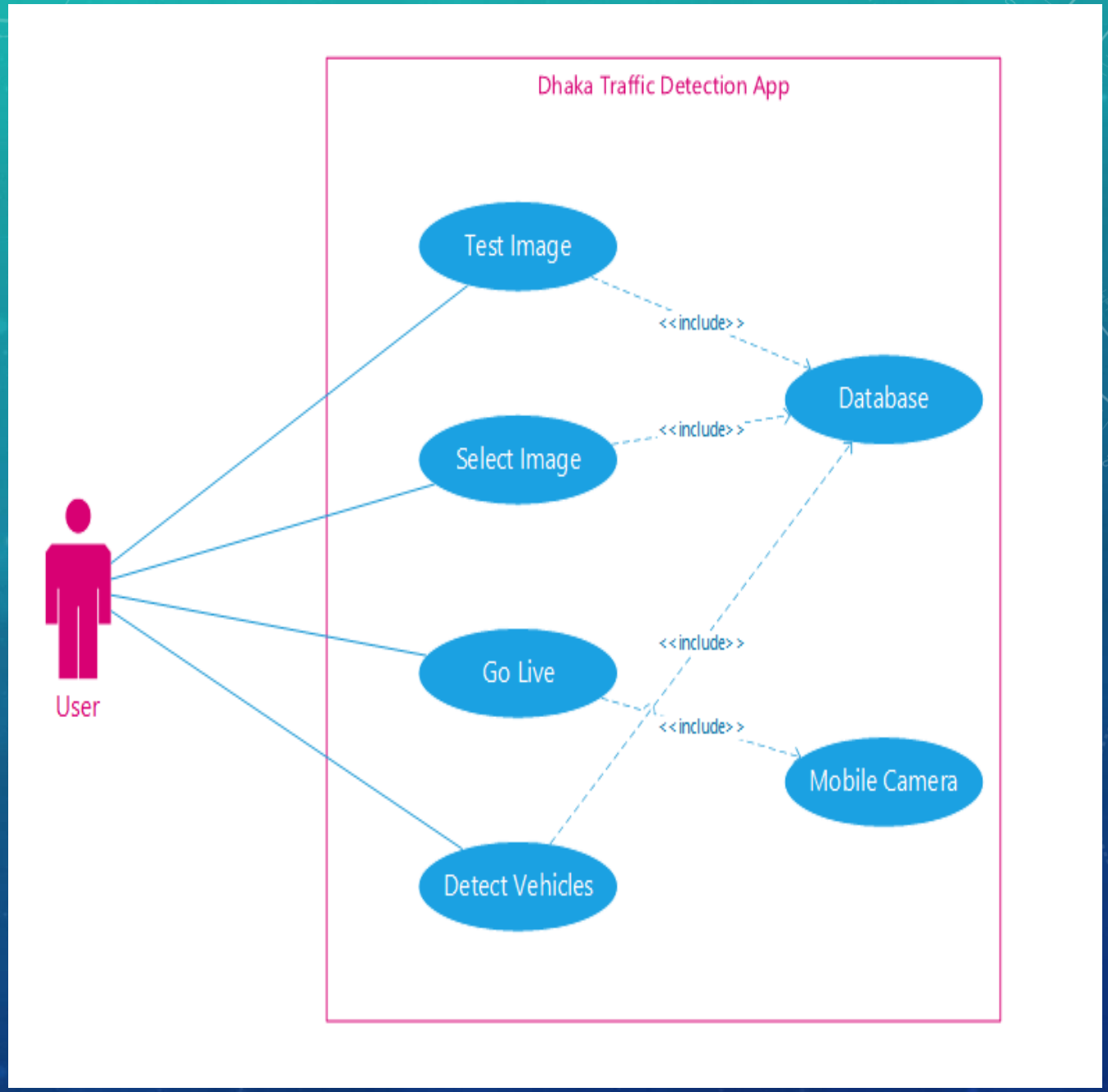
VEHICLES DETECTION FROM ANDROID APP



SYSTEM ATTRIBUTES

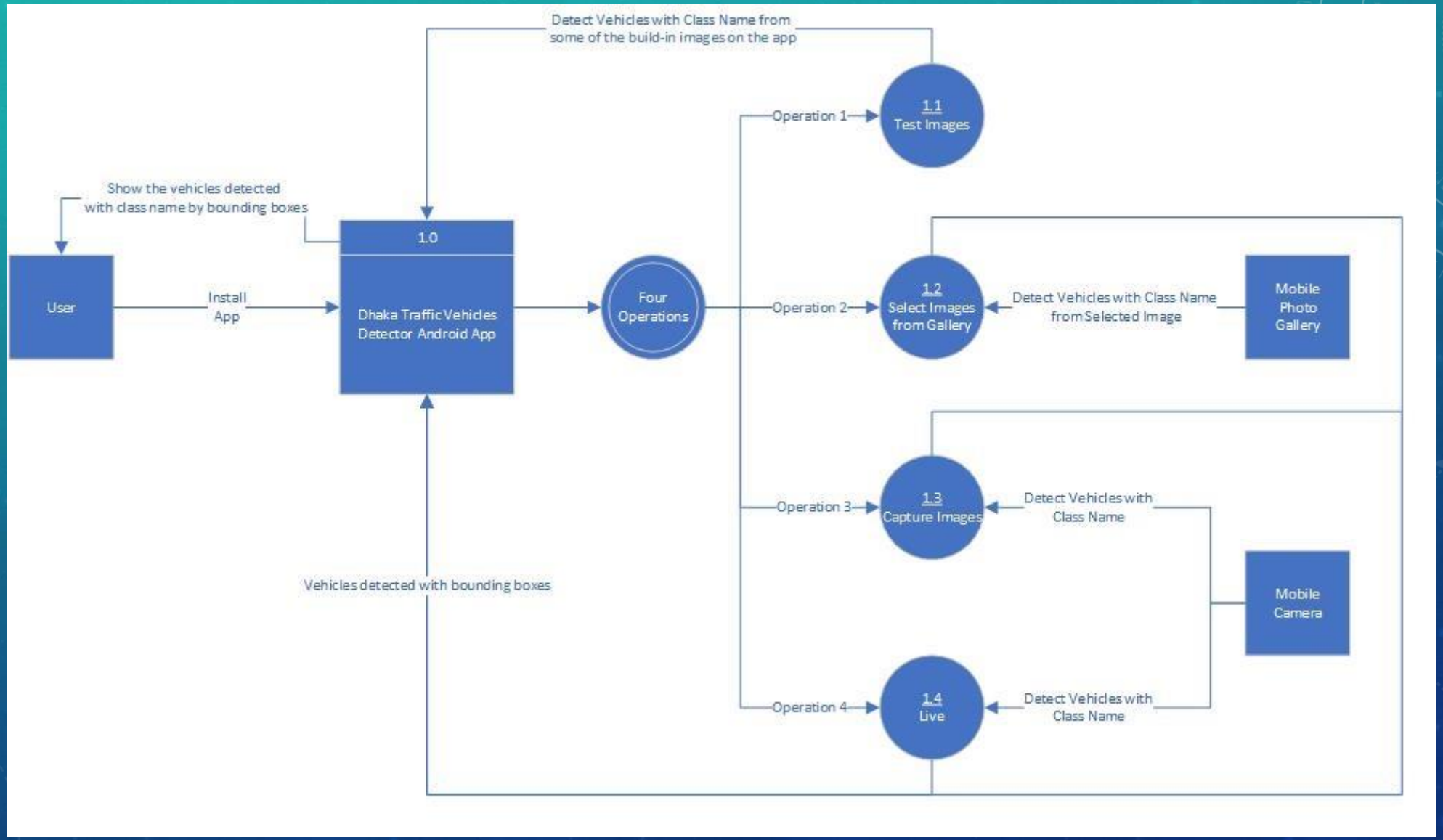
- **Test Image**
- **Select Image**
- **Go Live**
- **Detect Vehicles**

USE CASE DIAGRAM



CONTEXT LEVEL DATAFLOW DIAGRAM

LEVEL 1 DATAFLOW DIAGRAM



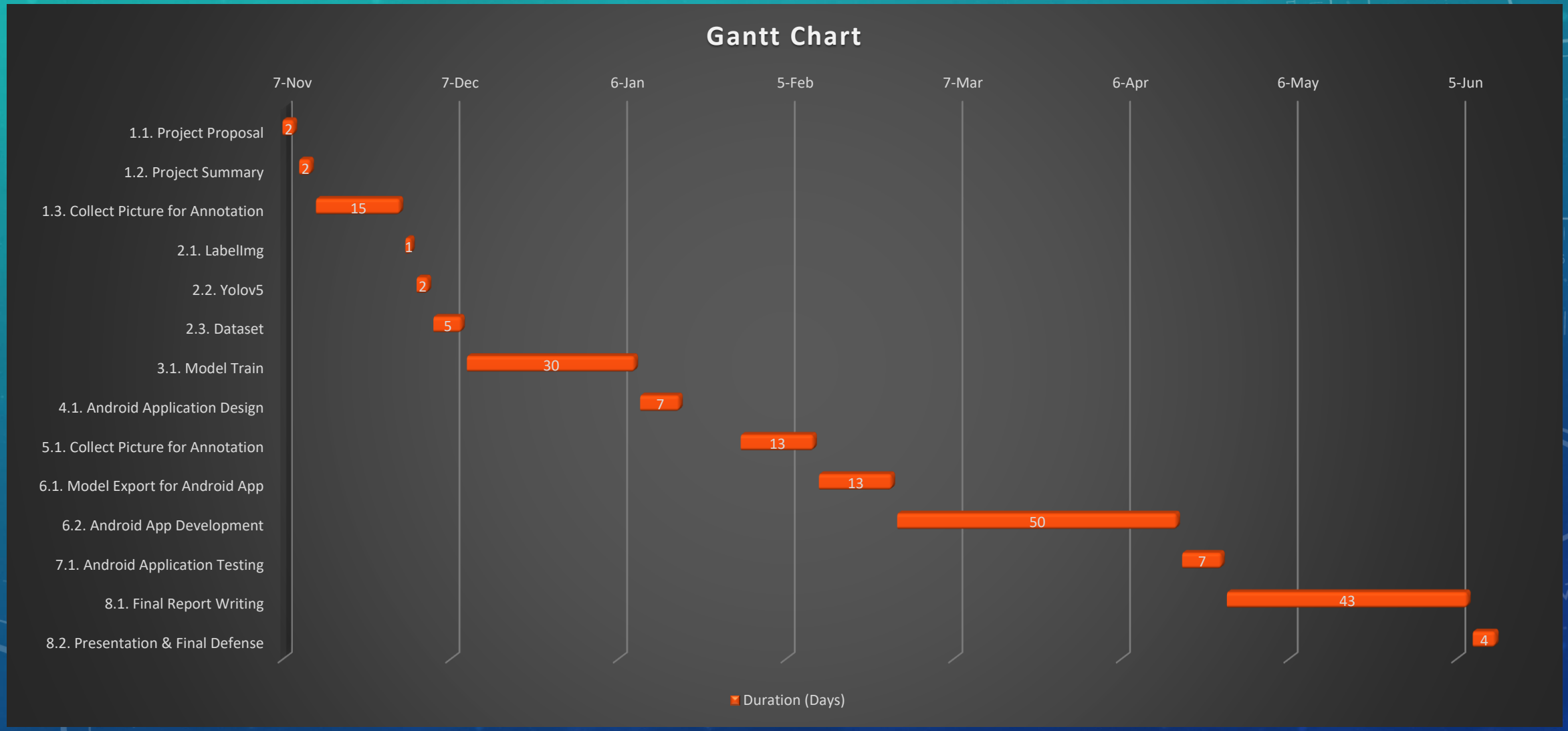
COST ANALYSIS

Types of Costs	Amount
Hardware Cost	
I. Acer/Dell/HP Laptop (GPU Included)	65,000 BDT
II. Android Smartphone with Minimum Android 11.0 Supported	17,000 BDT
Software Cost	
I. Anaconda Prompt	Free
II. Python	Free
III. Android Studio	Free
IV. Git Bash	Free
V. Kaggle Notebook	Free
VI. Adobe XD	Free
Tangible Costs	
I. Cost of Resources	Free
II. Cost of a Single Programmer	30,000 BDT
Intangible Costs	0 BDT
Total Amount	**1,12,000 BDT**

SCOPE FOR FUTURE WORK

- **Count** the numbers of a specific vehicles.
- Shows the **exact location** of a **“**Traffic Image**”.**
- Automatically **store** the detected images into the **cloud server.**

GANTT CHART


Gantt Chart
7-Nov
7-Dec
6-Jan
5-Feb
7-Mar
6-Apr
6-May
5-Jun
1.1. Project Proposal
1.2. Project Summary
1.3. Collect Picture for Annotation
2.1. LabelImg
2.2. Yolov5
2.3. Dataset
3.1. Model Train
4.1. Android Application Design
5.1. Collect Picture for Annotation
6.1. Model Export for Android App
6.2. Android App Development
7.1. Android Application Testing
8.1. Final Report Writing
8.2. Presentation & Final Defense
2
2
15
1
2
5
30
7
13
13
50
7
43
4
Duration (Days)

THANKS FOR WATCHING