KLE TECH

KLE Technological University

Creating Value
Leveraging Knowledge

Earlier Known as
B.V.B. College of Engineering and Technology, Vidyanagar, Hubballi -580031

Presentation on

Detection and classification of vehicles using Deep learning

Presented by,

Dinesh Dhotrad
Sanath Malagi
Prajwal Hebbar
Shivakumar M
Namrata Nyamagoudar

KLE Technological University,
Hubballi-580031, Karnataka, INDIA

Content

- Problem Statement
- Objectives
- Walkthrough
- Results

Problem Statement

Detection and classification of vehicles using Deep learning

Objectives

1. Selection of the model
2. Preparation of data set
3. Retrain the model using transfer learning with annotated data
4. Get the model’s lite version with TFLite
5. Create an application in android that runs the model

Selection of the model

1. Since we need to implement our project in android or any edge device we need to have **TFLite version** of model to implement it and TFLite doesn't support RCNN models.
2. So we chose **SSD-MobileNet** as our model
3. i.e: ”**SSD Mobilenet V2 COCO**” pre-trained model

Preparation of data set

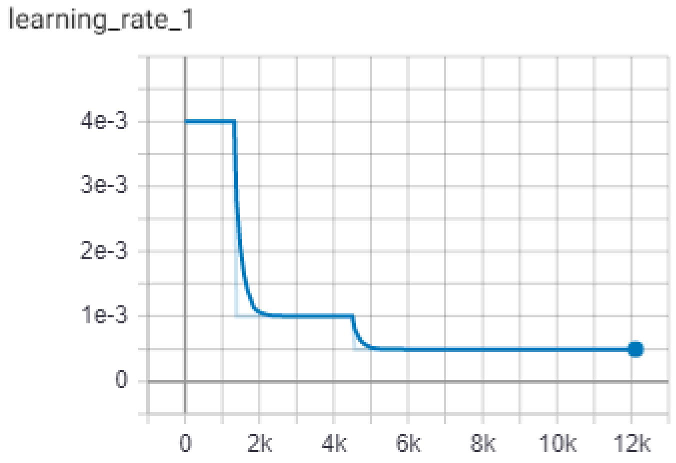
1. The dataset that we need should have annotated data
2. We chose to download the annotated data instead of annotating for ourselves
3. The dataset was provided by **Udacity**

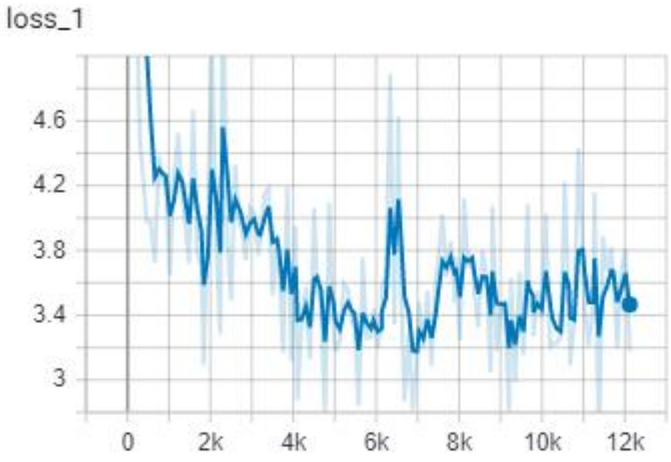
   “https://github.com/udacity/self-driving-car/tree/master/annotations”

4. The dataset consisted of 3 class namely : “**Car**”, “**Truck**”, “**Pedestrians**”
5. And the dataset was captured using **Dashboard Camera**.

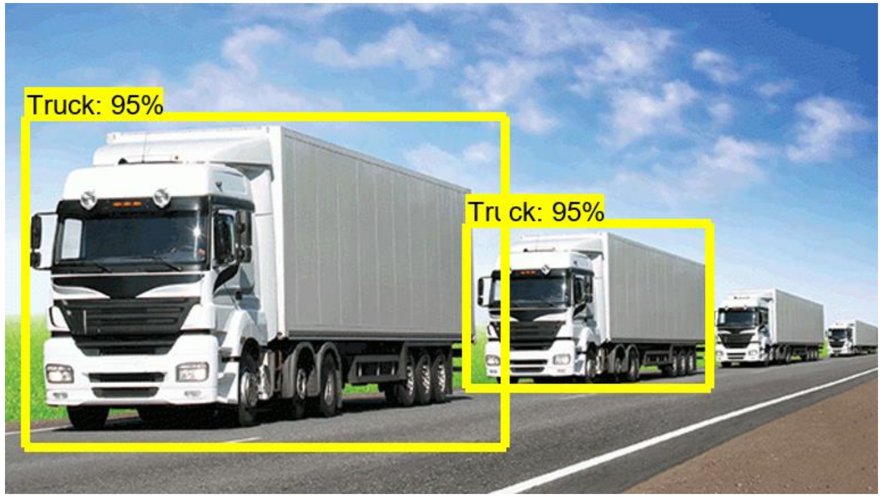
Retrain the model using transfer learning with annotated data

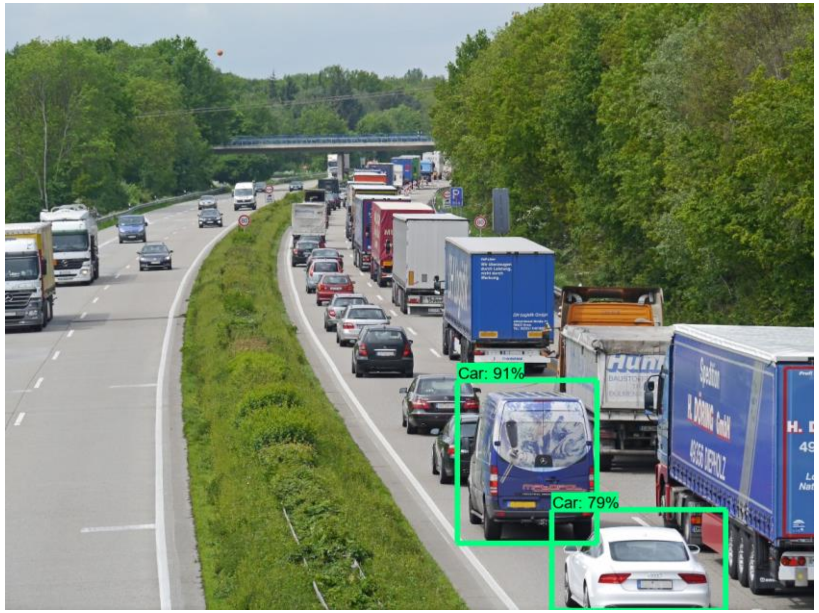
1. Configured ssd_mobilenet_v2_quantized_300x300_coco.config file for variable learning rate of [0.004,0.001,0.0005] and bath size of 24.
2. Retrained “ssd_mobilenet_v2_quantized_300x300_coco” model using transfer learning
3. Trained the model until steps reached 13k
4. The loss was averaging around 3.4

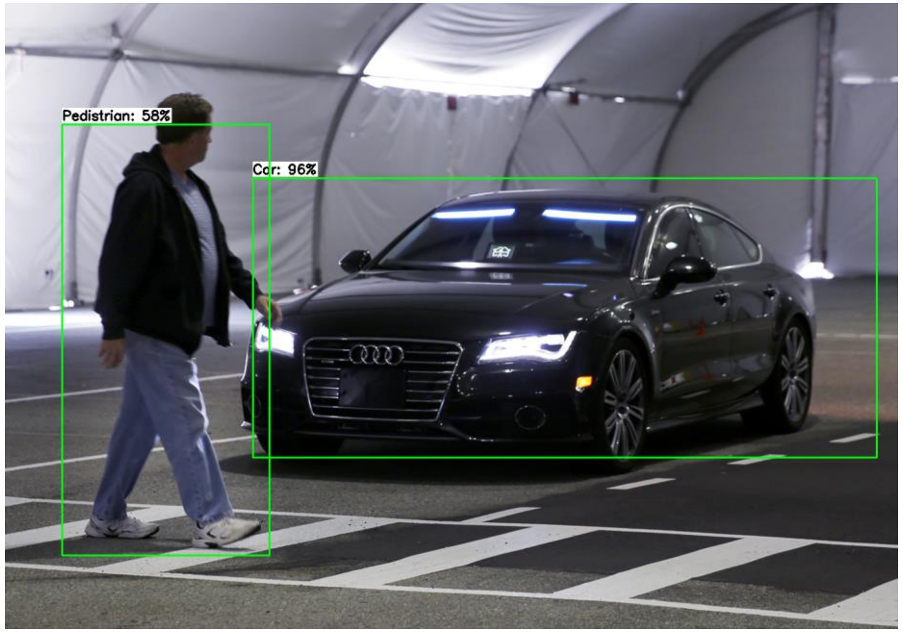


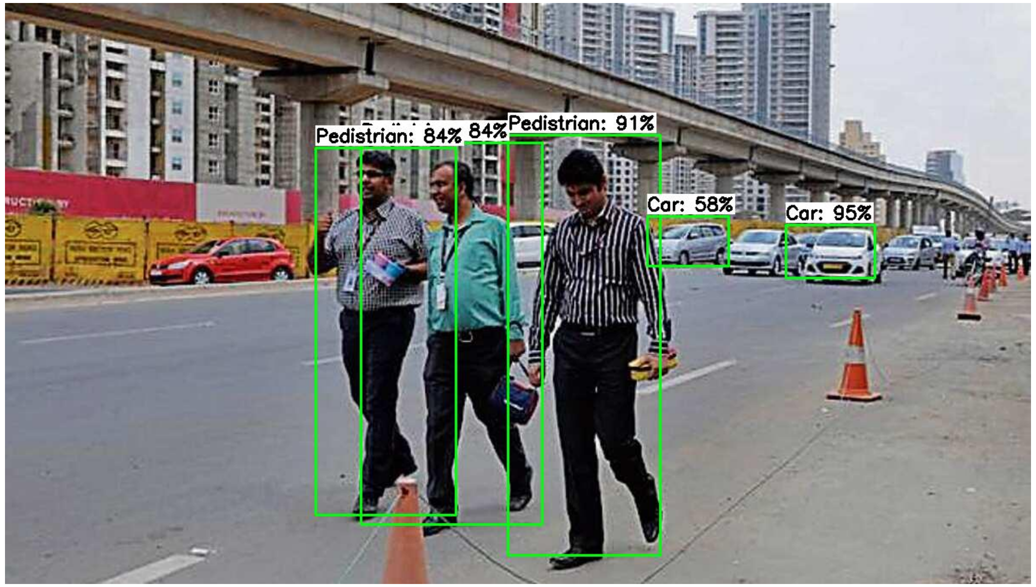


Retrained Results







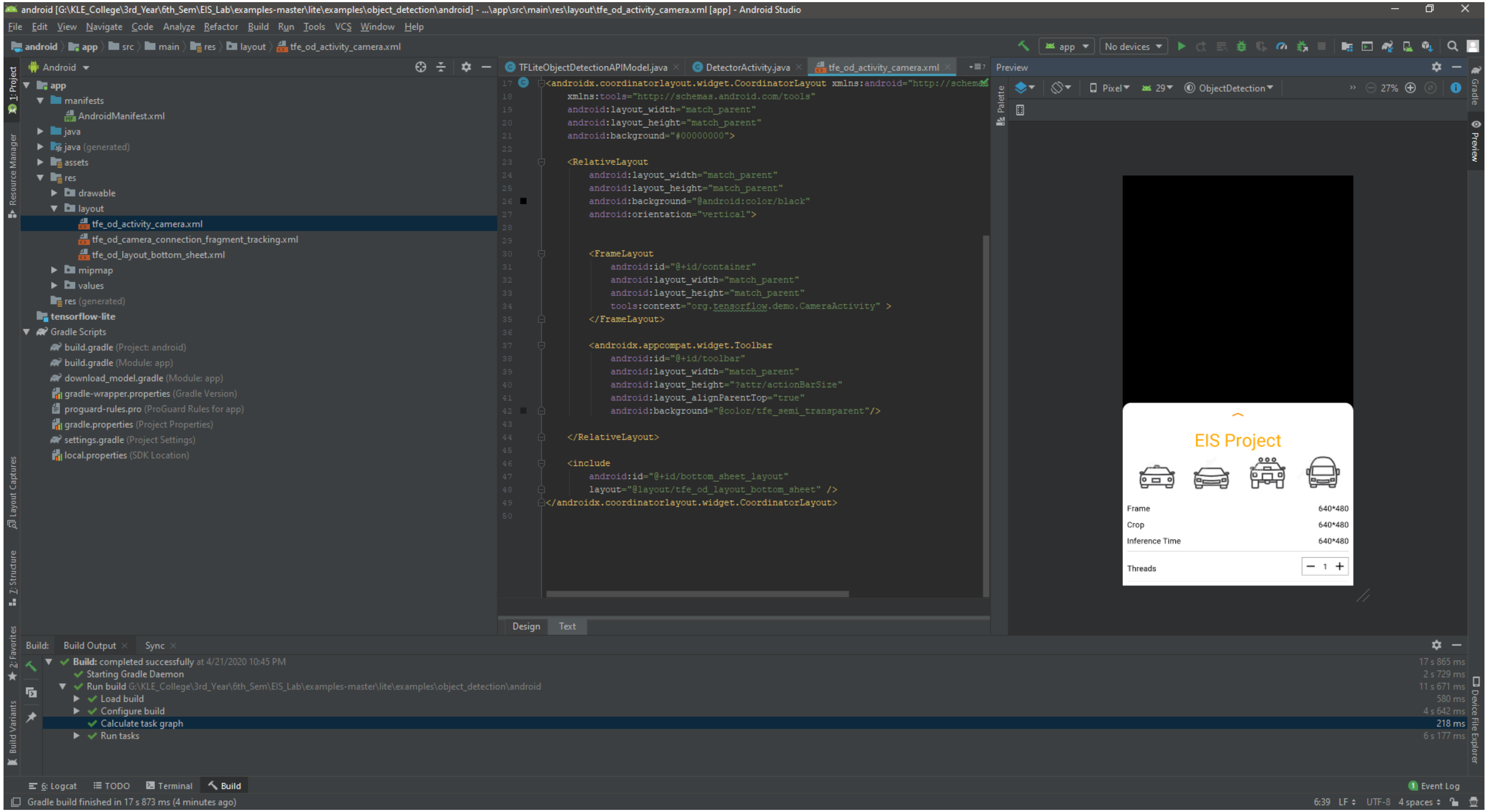


Get the model's lite version with TFLite

1. In order to convert the graph from **Tensorflow** to **TFLite** we need **BAZEL**
2. bazel v0.26.1 was configured into the system
3. Upon running bazel build and using TOCO converter we created optimized TFLite Model
4. We also needed to create labelmap.txt which was compatable with TFLite model.

Android application that runs the model

Using **detect.tflite** model and **labelmap.txt** for labels



android [G:\KLE_College\3rd_Year\6th_Sem\EIS_Lab\examples-master\lite\examples\object_detection\android] - ...\app\src\main\res\layout\tfe_od_activity_camera.xml [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
android › app › src › main › res › layout › tfe_od_activity_camera.xml
app
No devices
Android
app
manifests
AndroidManifest.xml
java
java (generated)
assets
res
drawable
layout
tfe_od_activity_camera.xml
tfe_od_camera_connection_fragment_tracking.xml
tfe_od_layout_bottom_sheet.xml
mipmap
values
res (generated)
tensorflow-lite
Gradle Scripts
build.gradle (Project: android)
build.gradle (Module: app)
download_model.gradle (Module: app)
gradle-wrapper.properties (Gradle Version)
proguard-rules.pro (ProGuard Rules for app)
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
TFLiteObjectDetectionAPIModel.java
DetectorActivity.java
tfe_od_activity_camera.xml
Preview
Pixel
29
ObjectDetection
27%
Palette
Design
Text
1: Project
Resource Manager
Layout Captures
Z: Structure
2: Favorites
Build Variants
Gradle
Preview
Device File Explorer

```xml
<androidx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#00000000">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@android:color/black"
        android:orientation="vertical">


        <FrameLayout
            android:id="@+id/container"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            tools:context="org.tensorflow.demo.CameraActivity" >
        </FrameLayout>

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:layout_alignParentTop="true"
            android:background="@color/tfe_semi_transparent"/>

    </RelativeLayout>

    <include
        android:id="@+id/bottom_sheet_layout"
        layout="@layout/tfe_od_layout_bottom_sheet" />
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

EIS Project

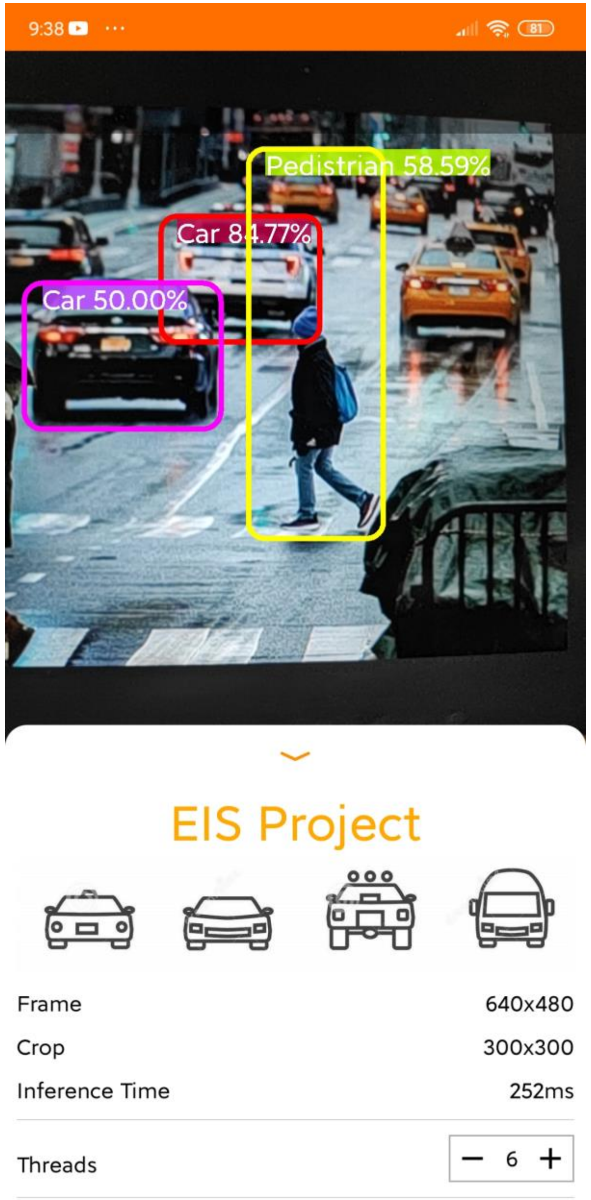
```
Frame              640*480
Crop               640*480
Inference Time     640*480

Threads          − 1 +
```

Build: Build Output × Sync ×

```
▼ ✓ Build: completed successfully at 4/21/2020 10:45 PM                                   17 s 865 ms
    ✓ Starting Gradle Daemon                                                              2 s 729 ms
  ▼ ✓ Run build G:\KLE_College\3rd_Year\6th_Sem\EIS_Lab\examples-master\lite\examples\object_detection\android   11 s 671 ms
    ▶ ✓ Load build                                                                        580 ms
    ▶ ✓ Configure build                                                                   4 s 642 ms
      ✓ Calculate task graph                                                              218 ms
    ▶ ✓ Run tasks                                                                         6 s 177 ms
```

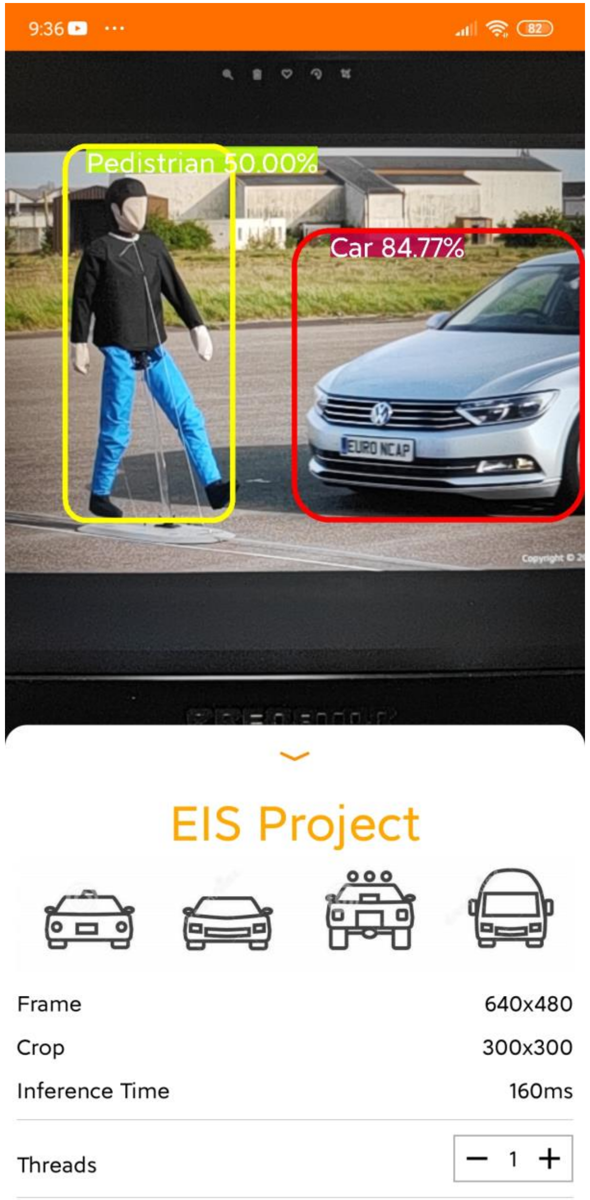
Gradle build finished in 17 s 873 ms (4 minutes ago)

Android application Results



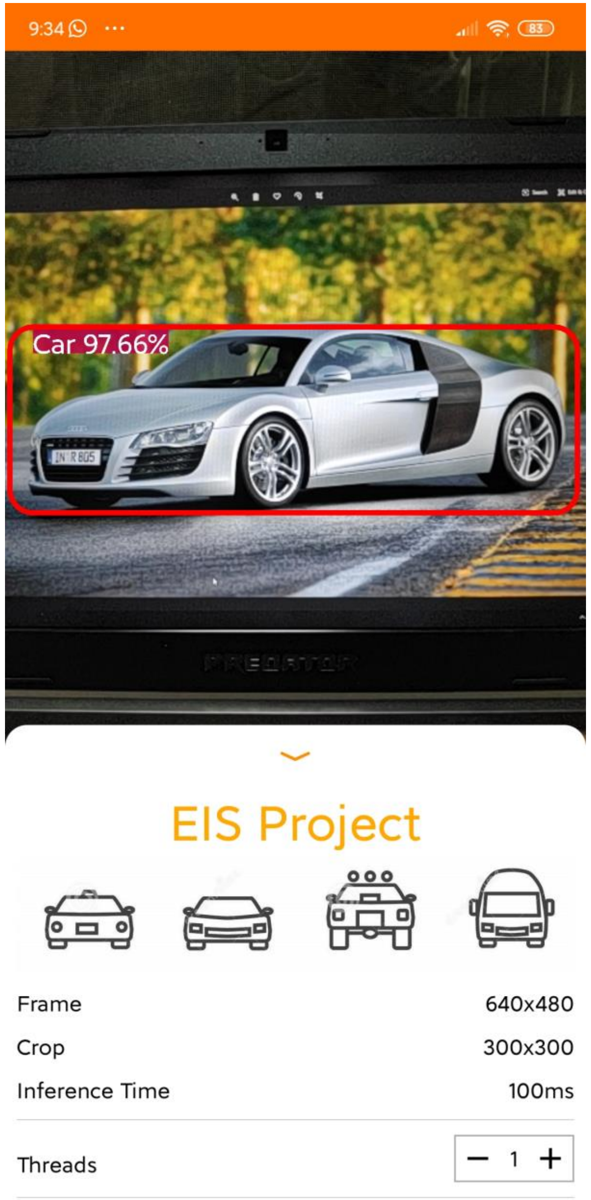
EIS Project

Frame 640x480
Crop 300x300
Inference Time 252ms
Threads − 6 +



EIS Project

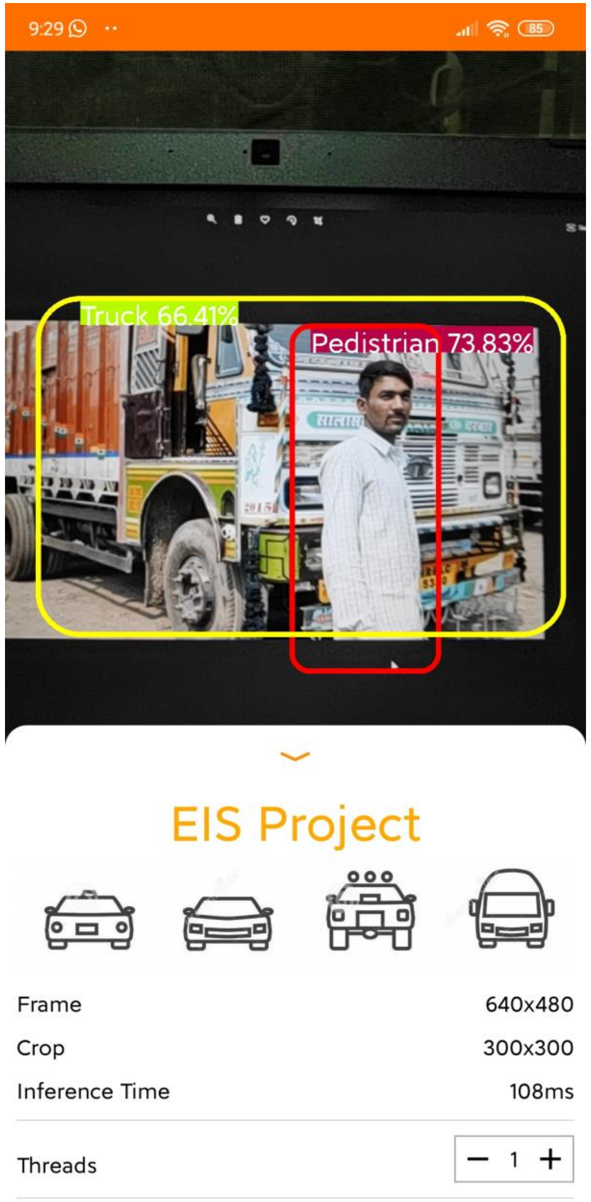
Frame 640x480
Crop 300x300
Inference Time 160ms
Threads − 1 +



EIS Project

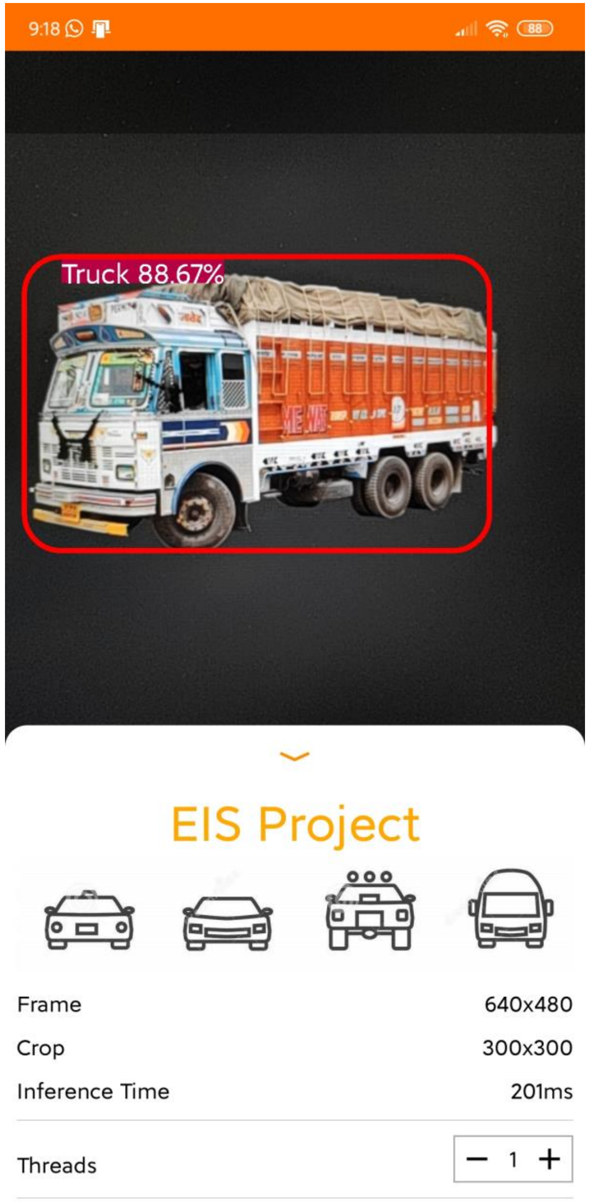
Frame 640x480
Crop 300x300
Inference Time 100ms
Threads − 1 +

Android application Results



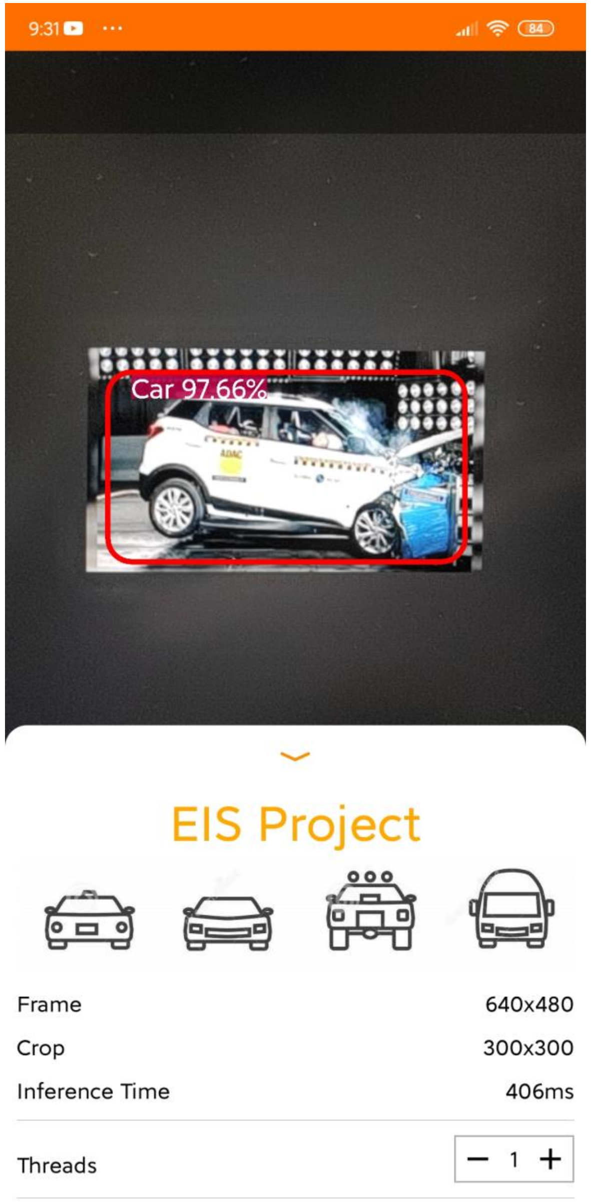
EIS Project

Frame	640x480
Crop	300x300
Inference Time	108ms
Threads	− 1 +



EIS Project

Frame	640x480
Crop	300x300
Inference Time	201ms
Threads	− 1 +



EIS Project

Frame	640x480
Crop	300x300
Inference Time	406ms
Threads	− 1 +