GITA AUTONOMOUS

B.Tech Technical Project Presentation - 2023

Distracted Driver

By

Name: Ravi Ranjan Reg. No: 1901287258

Name: Suman Kumar Maity Reg. No: 1901287276

Under the guidance of

Prof. Lopamudra Das

GITA AUTONOMOUS

OUTLINES

- Introduction
- Project Overview
- Description
- Advantages
- Disadvantages
- Problem Statement
- Data Exploration
- Data Preprocessing
- Implementation
- Approach
- Conclusion
- Installation

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

INTRODUCTION

According to the CDC motor vehicle safety division, one in five car accidents is caused by a distracted driver. Sadly, this translates to 425,000 people injured and 3,000 people killed by distracted driving every year.

State Farm hopes to improve these alarming statistics, and better insure their customers, by testing whether dashboard cameras can automatically detect drivers engaging in distracted behaviors. Given a dataset of 2D dashboard camera images, State Farm is challenging Kagglers to classify each driver's behavior. Are they driving attentively, wearing their seatbelt, or taking a selfie with their friends in the backseat?

Name: Ravi Ranjan **Reg. No: 1901287258**
Name: Suman Kumar Maity **Reg. No: 1901287276**

GITA AUTONOMOUS

Project Overview

Distracted driving has become a growing concern and a major cause of accidents on roads globally. Distracted driving refers to any activity that diverts the driver's attention from the road, such as using a mobile phone, eating, or adjusting the radio. To address this issue, various technological solutions have been proposed, including the development of distracted driver detection systems. In this project, I have created and refined machine learning models to detect what the driver is doing in a car given driver images. This is done by predicting the likelihood of what the driver is doing in each picture.

Description

Distracted Driver Detection systems are designed to monitor the behavior and activities of drivers while they are operating a vehicle. These systems use various technologies, including cameras, sensors, and machine learning algorithms, to detect signs of distracted driving and alert the driver to refocus on the road. The systems can also be integrated with other safety features, such as collision avoidance systems, to provide a more comprehensive solution to reducing distracted driving incidents.

Advantages

1. **Improved safety**: Distracted Driver Detection systems can help reduce the number of accidents caused by distracted driving, making roads safer for everyone.
2. **Increased awareness**: By alerting drivers to their distractions, these systems can increase awareness and encourage safe driving habits.
3. **Reduced insurance costs**: By reducing the number of accidents caused by distracted driving, these systems can help reduce insurance costs for both individuals and insurance companies.
4. **Better compliance with laws**: In many jurisdictions, using a mobile phone while driving is illegal, and these systems can help drivers comply with these laws and avoid costly fines.

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Disadvantages

1. **Privacy concerns**: Some people may be concerned about the privacy implications of having cameras or sensors monitoring their behavior while driving.
2. **Technical limitations**: Some systems may be limited by their accuracy, and false alarms can distract drivers and potentially cause accidents.
3. **Cost**: Some Distracted Driver Detection systems can be expensive to install and maintain, making them inaccessible to some drivers.

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Problem Statement

Given a dataset of 2D dashboard camera images, an algorithm needs to be developed to classify each driver's behaviour and determine if they are driving attentively, wearing their seatbelt, or taking a selfie with their friends in the backseat etc..? This can then be used to automatically detect drivers engaging in distracted behaviours from dashboard cameras.

Following are needed tasks for the development of the algorithm:

1. Download and preprocess the driver images
2. Build and train the model to classify the driver images
3. Test the model and further improve the model using different techniques.

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Data Exploration

The provided data set has driver images, each taken in a car with a driver doing something in the car (texting, eating, talking on the phone, makeup, reaching behind, etc). This dataset is obtained from Kaggle(State Farm Distracted Driver Detection competition).

Following are the file descriptions and URLs from which the data can be obtained :

- imgs.zip - zipped folder of all (train/test) images
- sample_submission.csv - a sample submission file in the correct format
- driver_imgs_list.csv - a list of training images, their subject (driver) id, and
- class id
- driver_imgs_list.csv.zip
- sample_submission.csv.zip

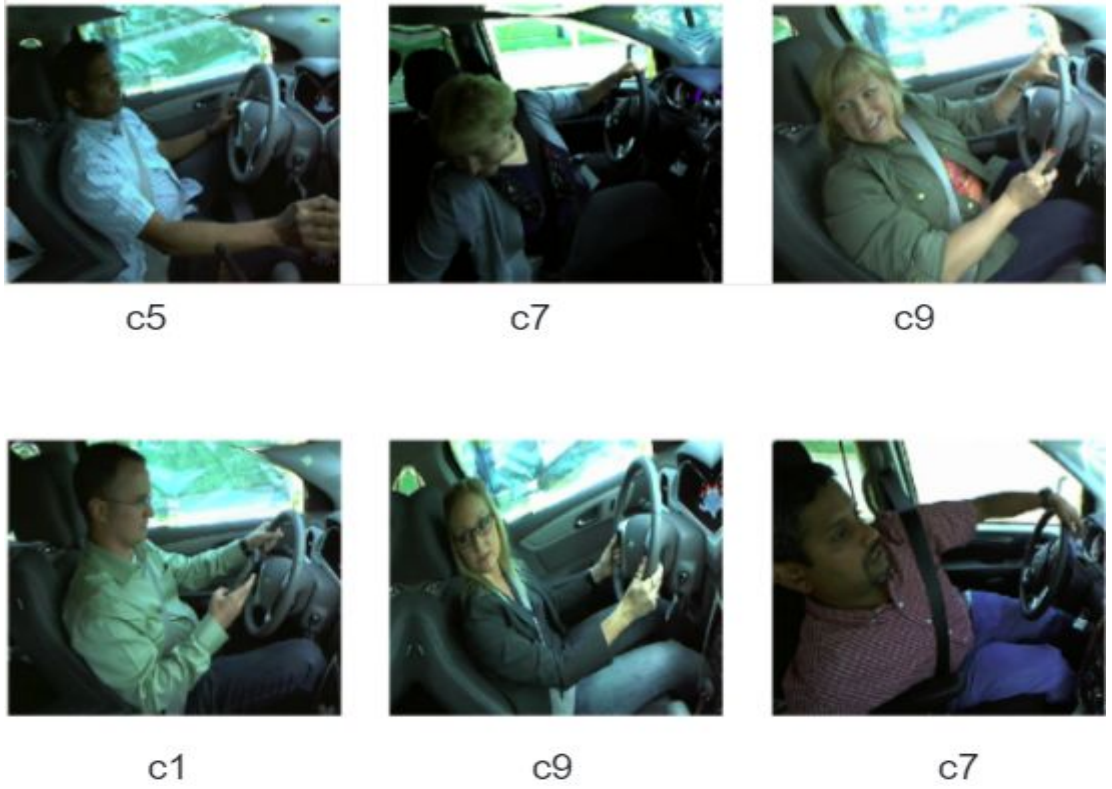
The 10 classes to predict are:

- c0: safe driving
- c1: texting - right
- c2: talking on the phone - right
- c3: texting - left
- c4: talking on the phone - left
- c5: operating the radio
- c6: drinking
- c7: reaching behind
- c8: hair and makeup
- c9: talking to passenger

There are 102150 total images. Of these 17939 are training images,4485 are validation images and 79726 are training images. All the training, validation images belong to the 10 categories shown above. The images are coloured and have 640 x 480 pixels each as shown below

Name: Ravi Ranjan **Reg. No: 1901287258**
Name: Suman Kumar Maity **Reg. No: 1901287276**

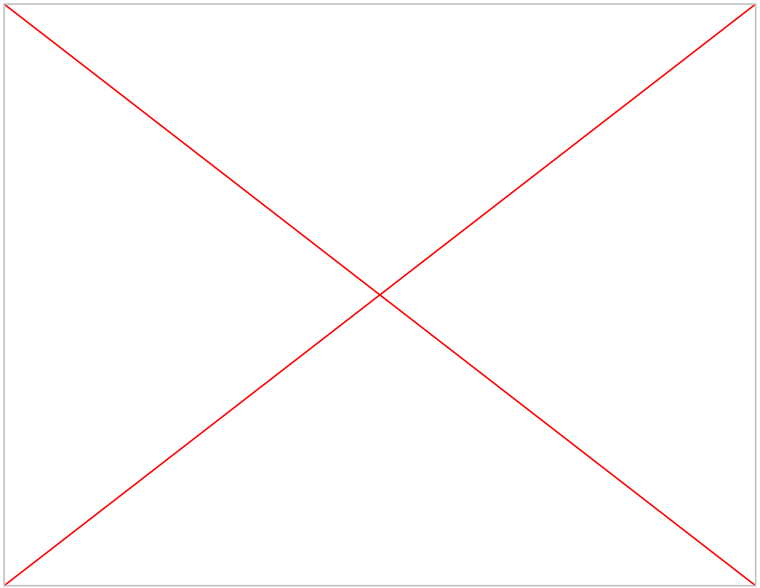
GITA AUTONOMOUS



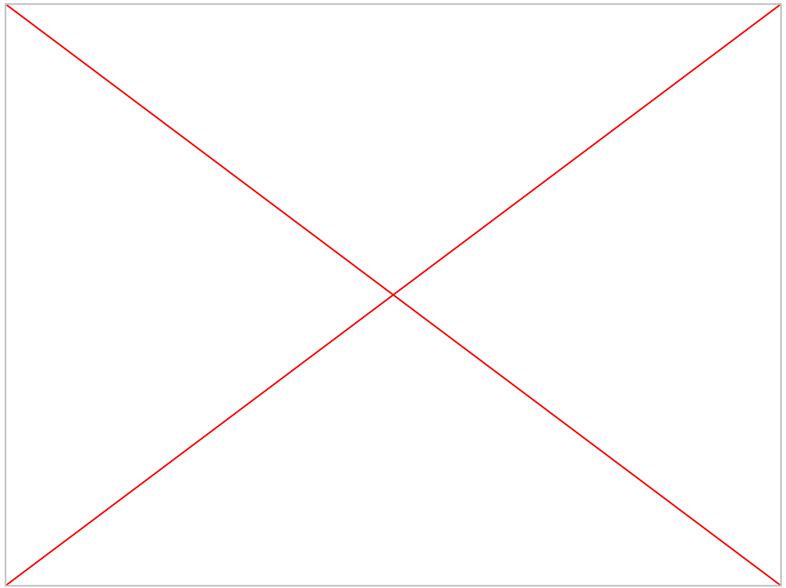
Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

Input Video



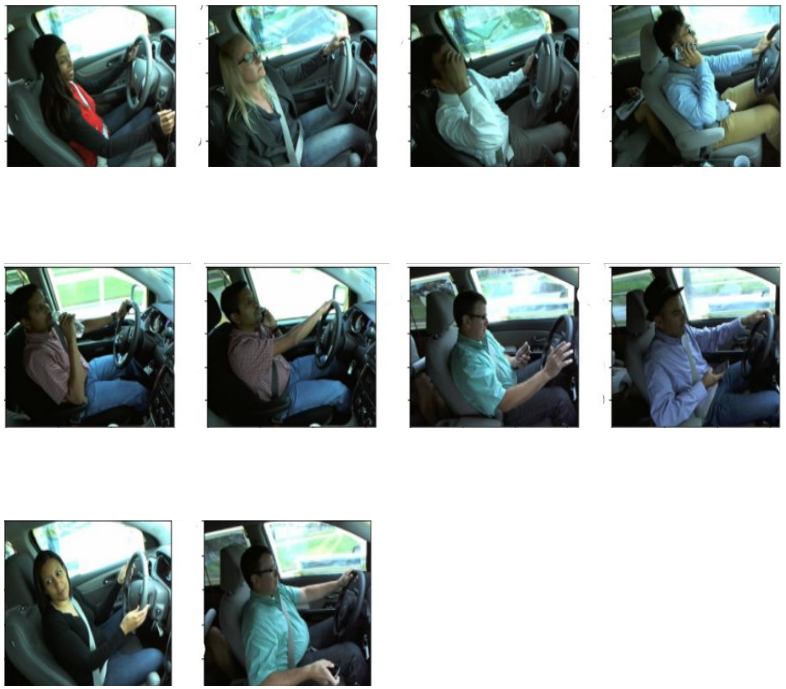
Output Video



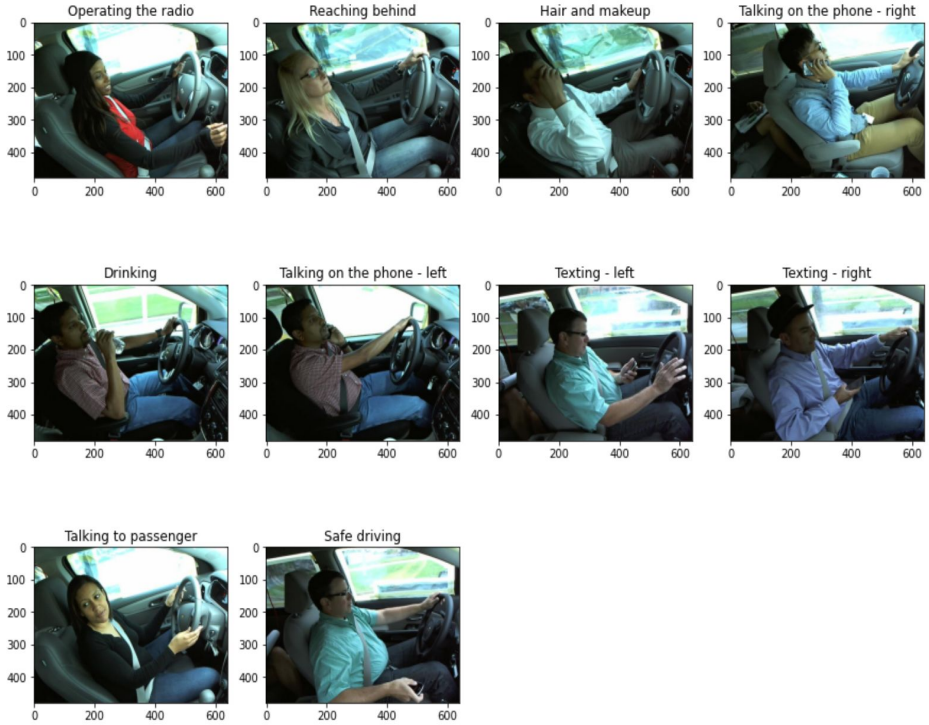
Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

Input Images



Output Images



Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Data Preprocessing

Preprocessing of data is carried out before model is built and training process is executed. Following are the steps carried out during preprocessing.Initially the images are divided into training and validation sets.

- The images are resized to a square images i.e. (100 x 100) pixels.
- Only one channels were used during training process as these are converted into gray images.
- The images are normalised by dividing every pixel in every image by 255.

```
def preprocessing(img):
    img = cv2.equalizeHist(img)
    img = img.reshape(100, 100, 1)
    img = img/255
    return img
```

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

Implementation

Conventional Neural Networks

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Approach

I followed the following four approaches one by one for improving the model results

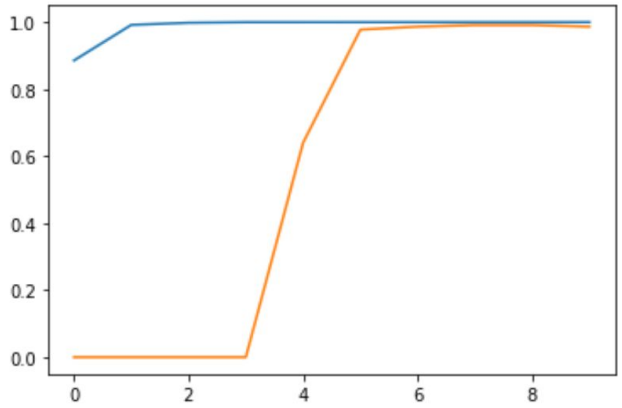
- Making a CNN architecture.

```
model = Sequential()

model.add(Conv2D(32, (3, 3), padding="same",input_shape = (100,100,1)))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=1))
model.add(MaxPooling2D(pool_size=(3, 3)))

model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=1))

model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=1))

model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())

model.add(Dense(1024))
model.add(Activation("relu"))
model.add(BatchNormalization())
model.add(Dense(10))
model.add(Activation("softmax"))
model.build((0,100,100,1))
```

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

Accuracy Plot

```
plt.plot(h.history['accuracy'])
plt.plot(h.history['val_accuracy'])
plt.show()
```



Accuracy

```
loss: 2.5471e-04 - accuracy: 1.0000 - val_loss: 0.0364 - val_accuracy: 0.9911

loss: 2.1247e-04 - accuracy: 1.0000 - val_loss: 0.0319 - val_accuracy: 0.9911

loss: 1.8572e-04 - accuracy: 1.0000 - val_loss: 0.0351 - val_accuracy: 0.9867
```

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

Conclusion

Distracted Driver Detection systems are an important step in reducing the number of accidents caused by distracted driving. These systems can provide a more comprehensive solution to the problem of distracted driving, incorporating various technologies to monitor driver behavior and alert drivers to distractions. Despite some limitations and privacy concerns, the advantages of these systems, including increased safety and reduced insurance costs, make them a valuable investment for individuals and society as a whole.

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

Installation

1. Install `Python 3.10` on your Local Machine
2. Create Python Virtual Environment in root folder by opening terminal and executing
   * `pip install virtualenv`
   * `virtualenv distracted_env`

   * `source distracted_env/bin/activate`

3. Install required Python Libraries by

   `pip install -r requirements.txt`

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

THANK YOU

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276

GITA AUTONOMOUS

https://docs.google.com/presentation/d/1-y2ePvyZvTKSrKftbshTz4VTVM0w0sNqQHcb50tjDy4/edit?usp=sharing

ANY QUESTION

Name: Ravi Ranjan Reg. No: 1901287258
Name: Suman Kumar Maity Reg. No: 1901287276