Detect the driver state by using Deep learning classification model .

By: Nisreen Alsayegh,
Afnan Alsirhani

OUTLINE:

Introduction:

every day about 13 people in the Saudi Arabia are killed in crashes.

People who use their cell phones to talk or text while driving are by far the most common reason for distracted driving accidents.

In this project our task is to get a dataset to build a deep learning classification model.

We chose to analyze a dataset obtained from kaggle that collects an images for the driver state The dataset contains 10 classes each class contains around 2000 images in the training set and 2400 in the testing set .

https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

Goals:

Build Deep learning classification model to predict the state of the driver .

Tools:

Numpy

Jupyter

Tensorflow

Seaborn

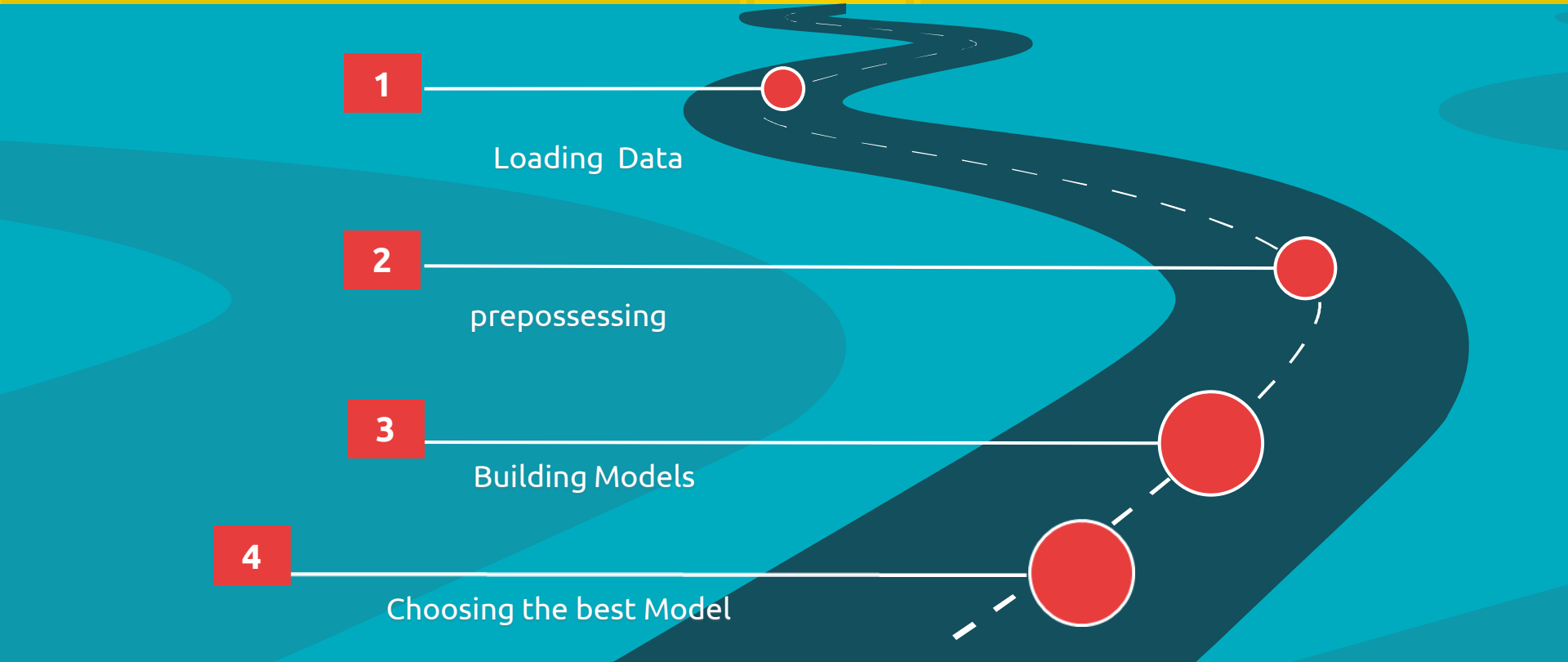
Pandas

keras

Matplotlib

Scikit-Learn

Workflow


1
Loading Data
2
prepossessing
3
Building Models
4
Choosing the best Model

—Prepossessing:

Dataset Before:
Training dataset: 10 classes
22,024 images

Test dataset:
2400 unlabelled images

Dataset After:
Training dataset: 3 classes
19624 images
Validation dataset:
3 classes 2400 images
Test dataset:
3 classes 2400 images

Image size:150*150
Batch size = 32
Reshape

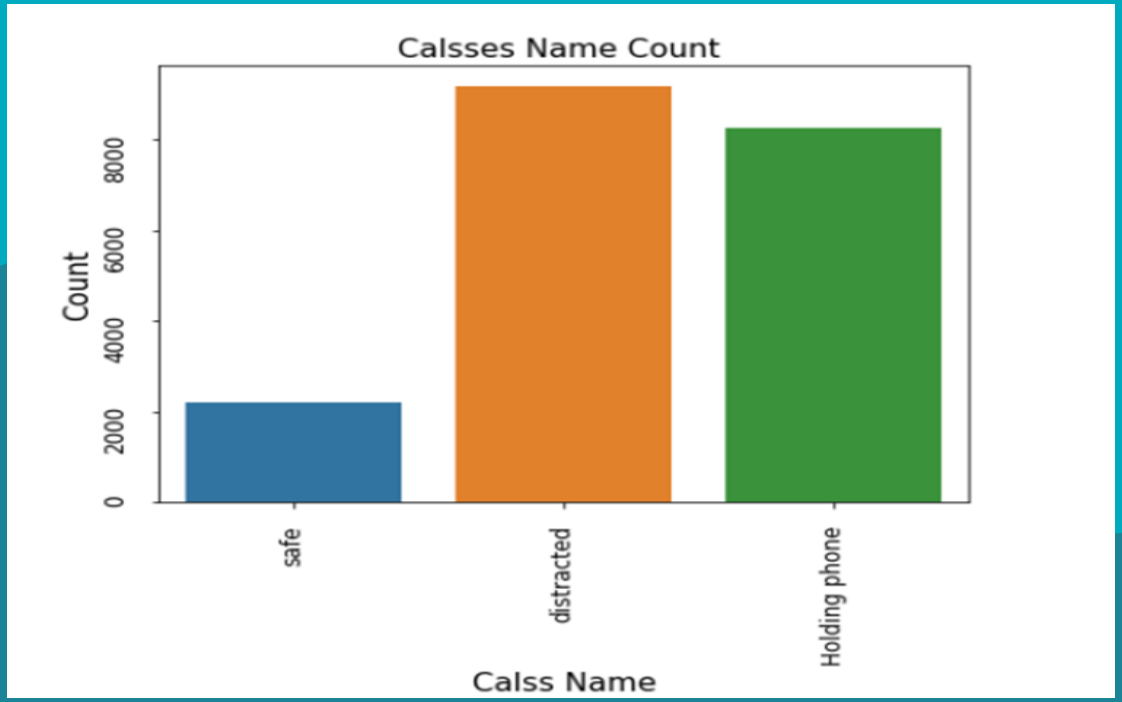
Prepossessing:

Safe

Holding phone

Distracted

EDA (Exploratory Data Analysis):



Modelling:

Base models :

	Base model1	Base model2-NN	Base model3-NN
	Logistic regression	1 hidden layer	6 hidden layer
Train	.50	1	1
Validation	.50	.998	.999

Modelling:

CNN models :

	model1	model2	model3	model4
	MaxPooling2D, activation= relu ,no padding, small size unit, Hidden layers=4, epochs=10	MaxPooling2D, activation= relu ,no padding, large size unit, Hidden layers=4, epochs=10	MaxPooling2D, activation=tanh padding=same, small size unit , Hidden layers=4, epochs=10	MaxPooling2D, activation= relu , padding=same, small size unit, Hidden layers=4 , epochs=10
Train	1	.98	.43	.999
Validation	.997	.97	.53	.995

Modelling:

Transfer Learning Models :

	VGG16	MobileNet (V2)
	activation= relu small size unit, Hidden layers=2, epochs=10	activation= relu small size unit, Hidden layers=2, epochs=10
Tarin	.999	1
Validation	.994	.996

Modelling:

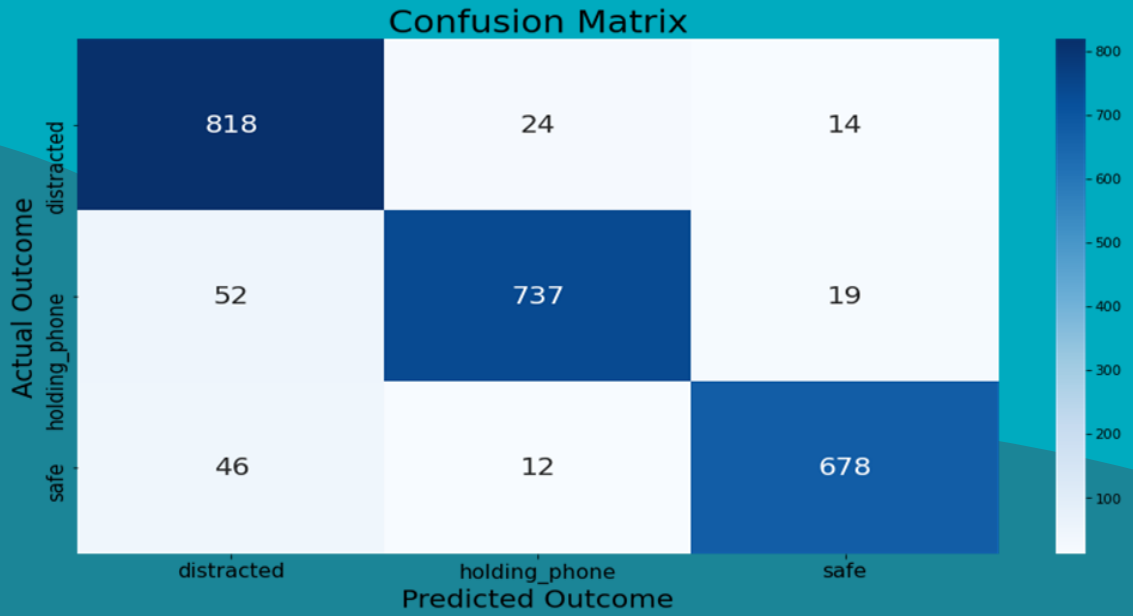
Our Best Models :

	VGG16	MobileNet (V2)	model1
	activation= relu small size unit, Hidden layers=2, epochs=10	activation= relu small size unit, Hidden layers=2, epochs=10	MaxPooling2D, activation= relu ,no padding, small size unit, Hidden layers=4 , epochs=10
Tarin	.999	1	1
Val	.994	.996	.997
Tarin+val	.996	1	.998
Test	.93	.89	.87

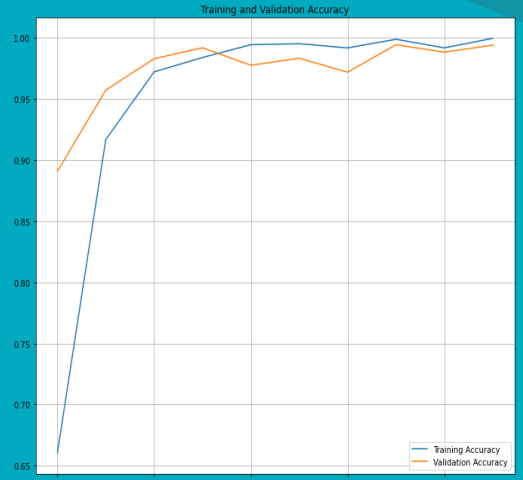
Best Model (VGG16)

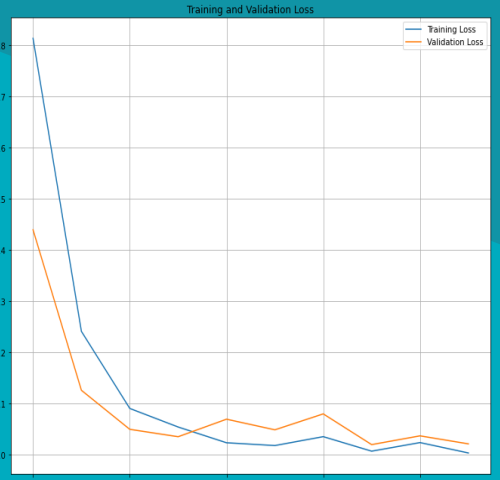
Results:

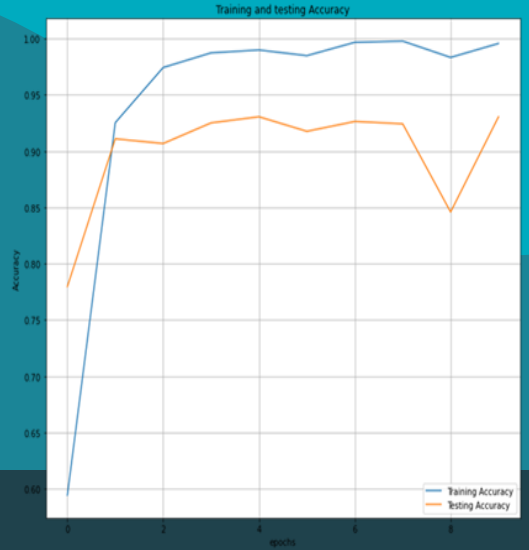
Train Accuracy: 0.998618
Train Precision: 0.995899
Train Recall: 0.998999
Train F1 score: 0.997436
Test Accuracy: 0.930417
Test Precision: 0.933343
Test Recall: 0.929644
Test F1 score: 0.930894

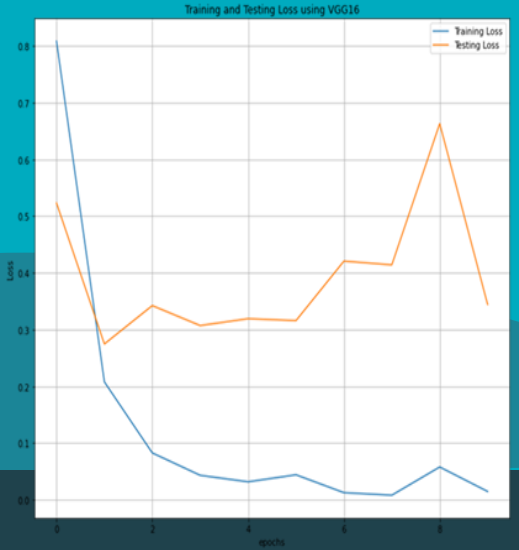


Our best model: VGG16





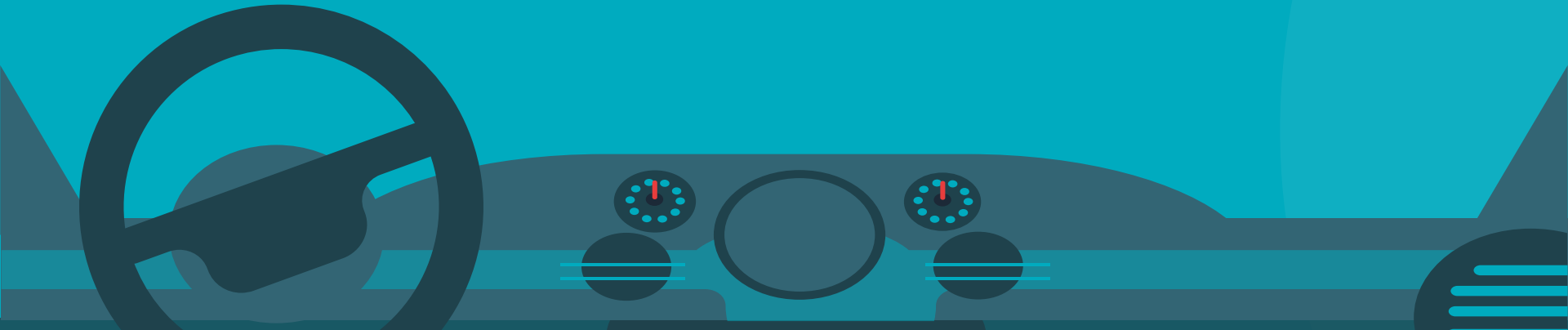




Conclusion:

Most models give us a good results in the both training and validation set and no overfitting .

VGG16 shows best result in the test set .



Future work::

- Collect more images from different angle.
- Do more Tuning the models by try different number of units and layers change the values of the hyperparameter .
- Build an app that notify the driver when they git distracted.

Thank You