

BITI 1113 – Artificial Intelligence

HUMAN EMOTION

RECOGNITION

Machine Learning Project

Present to **Prof. Ts. Dr. Goh Ong Sing**



Tan Wei Yin

B032020029
1BITS S1G2



Pang Jia Mei

B031910436
2BITS S1G2



Mirza Sahid Afridi

B031920505
2BITS S1G1



Nur Afiqah Bt Raman

B031910100
2BITS S1G2

Group J

Acknowledgement

akmadan/
Emotion_Detection_CNN



1

Contributor

1

Issue

4

Stars

12

Forks



Medium

The 4 Convolutional Neural Network Models That Can Classify Your Fa...

Clothes shopping is a taxing experience. My eyes get bombarded with too much information. Sales, coupons, colors, toddlers, flashing...



YouTube

Misbah Mohammed

Emotion Detection -Python Project using Machine Learning & OpenCV -...



osgo88/**Artificial-Intelligence-Project**



1

Contributor

0

Issues

0

Stars

10

Forks

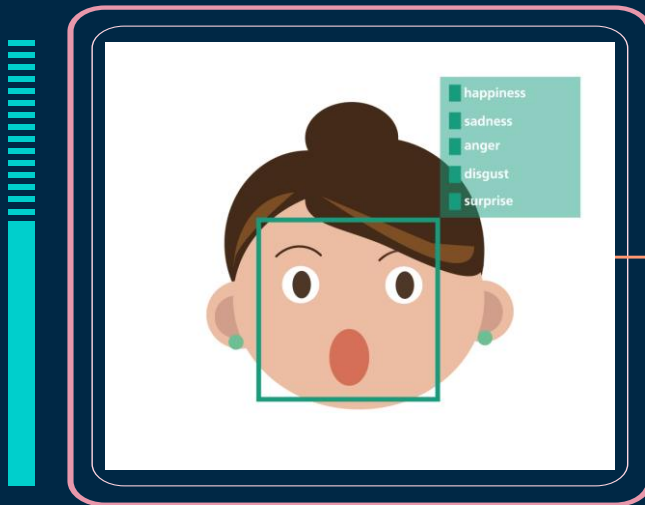


The background is a dark blue field decorated with a pattern of small, scattered squares in teal, orange, and pink. Thin, light-colored vertical lines of varying lengths are also scattered across the page, some intersecting with the squares.

PROJECT SUMMARY

OBJECTIVES

- To classify each face based on the emotion shown in the facial expression into one of six categories (Angry, Fear, Happy, Neutral, Sad, Surprise).
- To help people to see their present mental condition.
- To develop an algorithm that can detect facial expression in static image and live video.



SCOPE OF THE PROJECT :

Feeding the machine images of our facial expressions via the inbuilt camera.

Machine ability :
Able to identify six different type of emotional states in real-time

This project focuses on the concept of classification

Technique used :
4-Conv
Layered CNN Model



Convolutional Neural Network

Machine Learning

Computer Vision

Keras

The background is a dark blue field populated with various geometric elements. Thin white vertical lines of varying lengths are scattered across the frame. Small squares in teal, orange, and pink are also distributed throughout. The word 'ABSTRACT' is centered in a large, white, sans-serif font.

ABSTRACT

ABSTRACT

We give the machine lots of examples of data, demonstrating what we would like it to do so that it can figure out how to achieve a goal on its own.

The confidence value provides us with an indication of how sure the AI is of its classification.

Machine learning
is an application of
Artificial Intelligence.

The more varied the data we provide, the more likely the AI will correctly classify the input as the appropriate emotion.

Classification is a learning technique used to group data based on attributes or features.


Project Structure

```
|— misc (8 entries)                                # Miscellaneous information
|   |— GroupJ_Slides.pdf
|   |— Group_J.jpg
|   |— OriDataSet.jpg
|   |— SelectedDataSet.jpg
|   |— TestResult.jpg
|   |— confusionmatrix.jpg
|   |— dataset.jpg
|   |— emotionimg.jpg
|   |— livedemo.jpg
|   |— training_loss_accuracy.png
|— src (5 entries)
|   |— haarcascade_frontalface_default.xml          # To detect the face of individuals
|   |— HumanRecognitionTrainModel.ipynb            # Training script
|   |— Main.ipynb                                   # Entry point of webcam live demo
|   |— Model.h5                                     # Load Keras Model
|   |— PlotConfussionMatrix.ipynb                  # Confusion matrix visualization
|— .gitignore
|— README.md
2 categories, 15 files
```

Dataset

171

Face expression recognition dataset

 Jonathan Oheix • updated 2 years ago (Version 1)

[Data](#) [Tasks](#) [Code \(24\)](#) [Discussion \(1\)](#) [Activity](#) [Metadata](#)

[Download \(54 MB\)](#) [New Notebook](#)

Usability 1.9


Tags No tags yet

Data Explorer


53.89 MB

- Images
 - train
 - validation

< images (2 directories)



train
7 directories



validation
7 directories

Summary

- 35.9k files

Original Dataset

	ANGRY	SAD	NEUTRAL	SURPRISE	HAPPY	DISGUST	FEAR
TRAINING	3993	4938	4982	3205	7164	436	4103
VALIDATION	960	1139	1216	797	1825	111	1018

Training : 28,821 images

Validation : 7,066 images

Selected Dataset

ANGRY

SAD

NEUTRAL

SURPRISE

HAPPY

FEAR

Original Dataset

TRAINING

3005

3015

3018

3006

3003

3008

VALIDATION

501

509

504

502

504

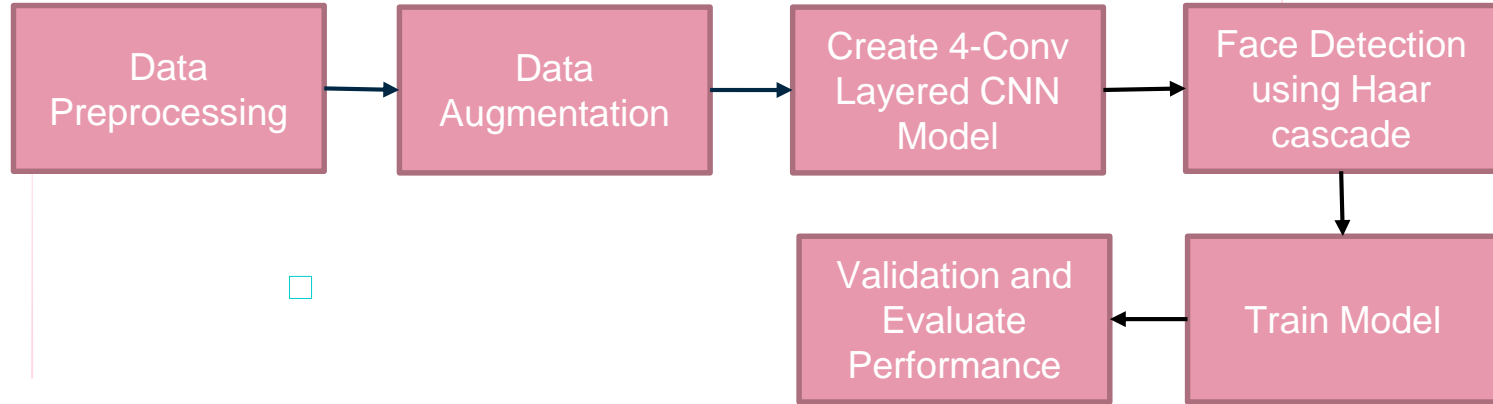
5005

Training : 18,055 images

Validation : 3,025 images

Project Phase

Phase #1 : Training the Model



Phase #2 : Deployment

Use Model to
Recognize Facial
Expressions in
Videos

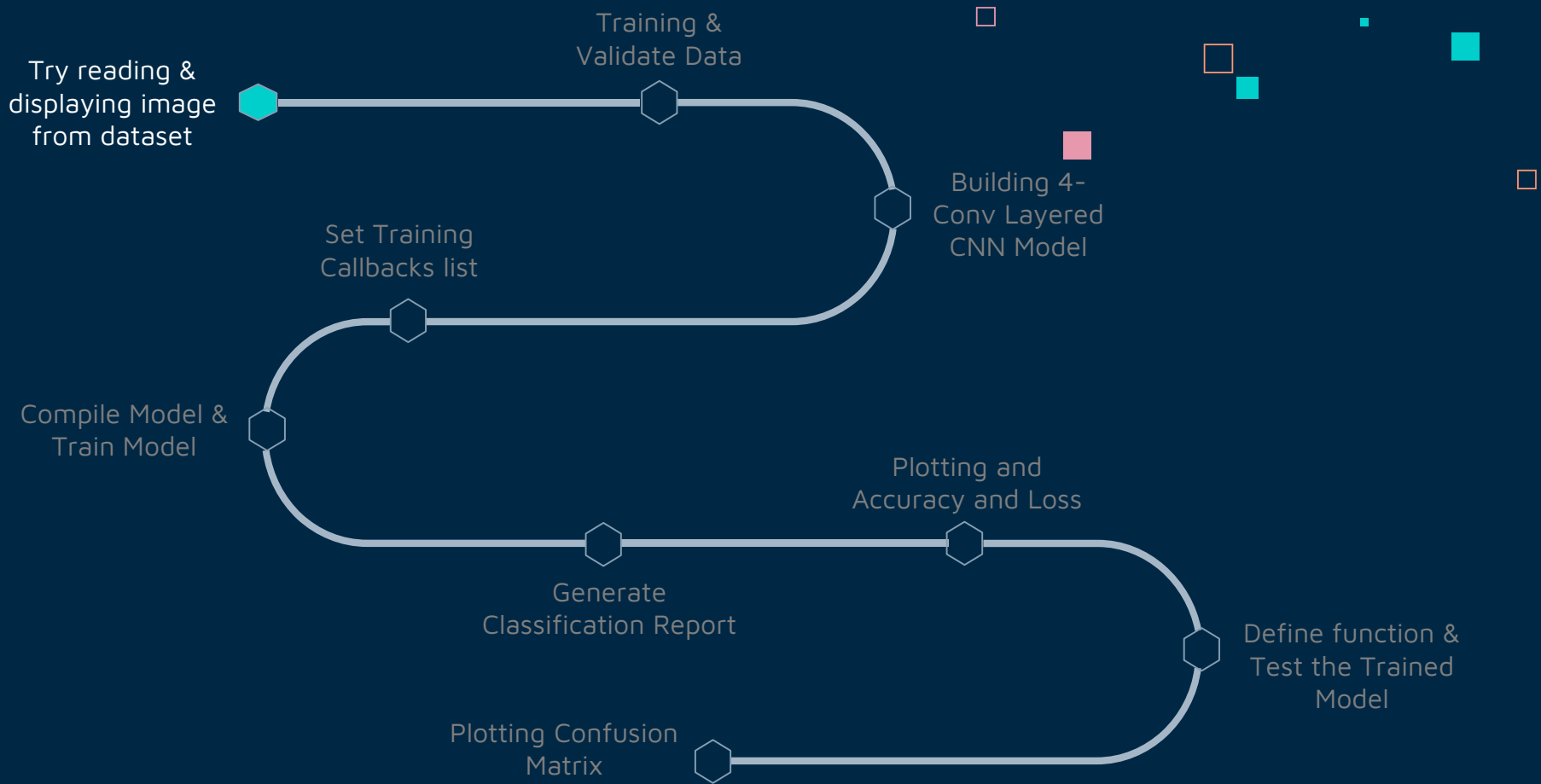
Testing on
Static Images

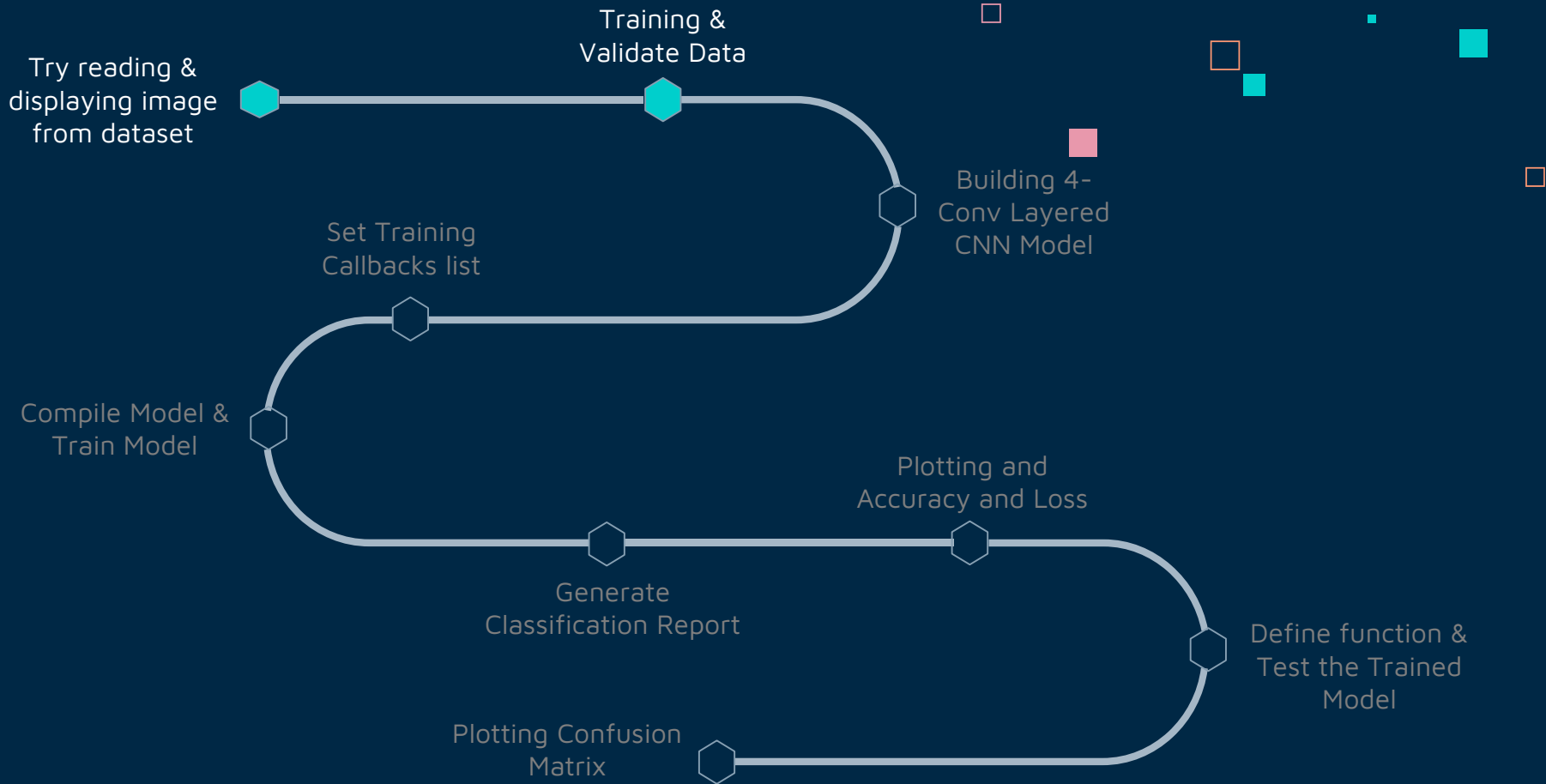
Show
Significant
Results

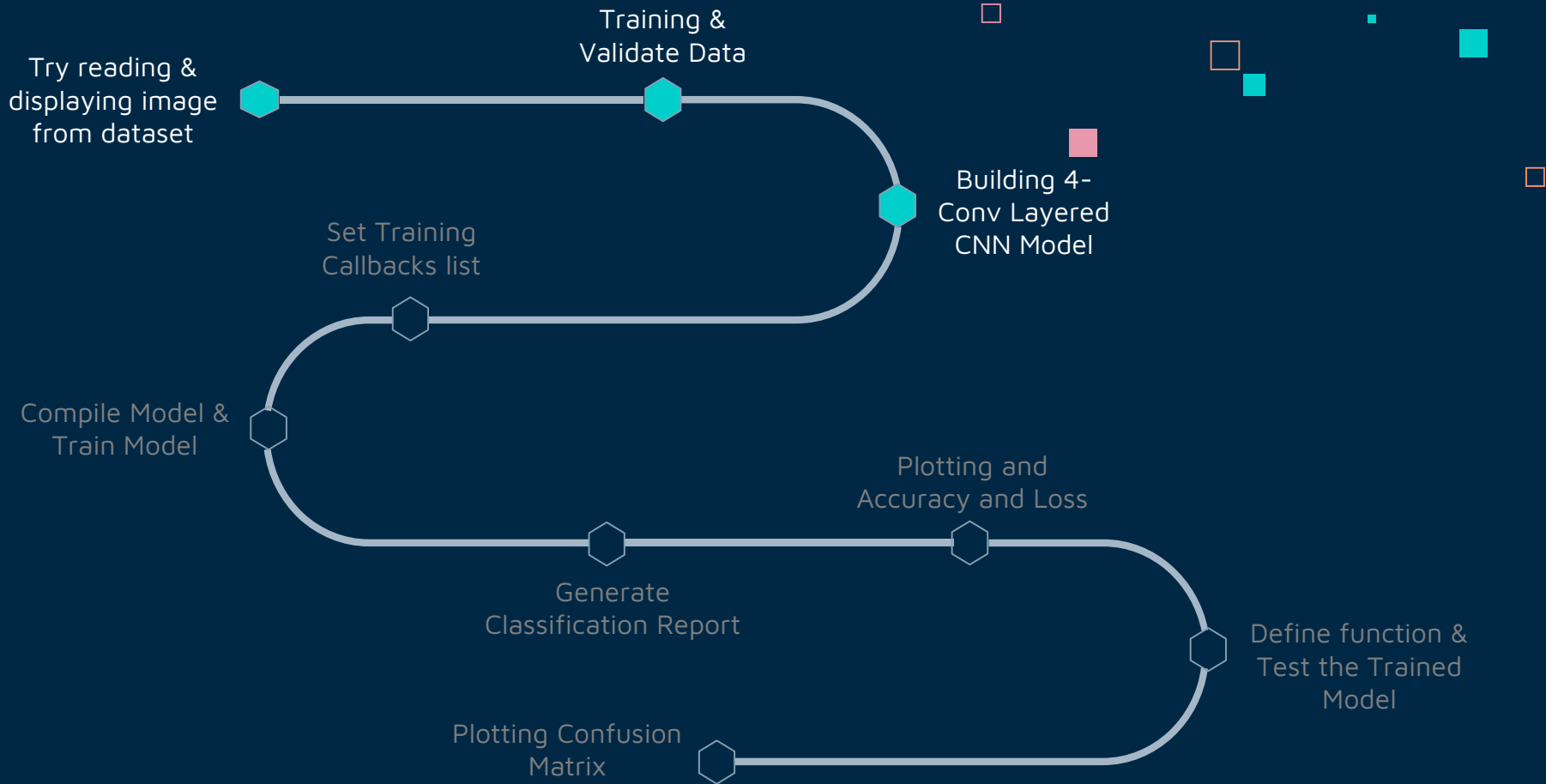


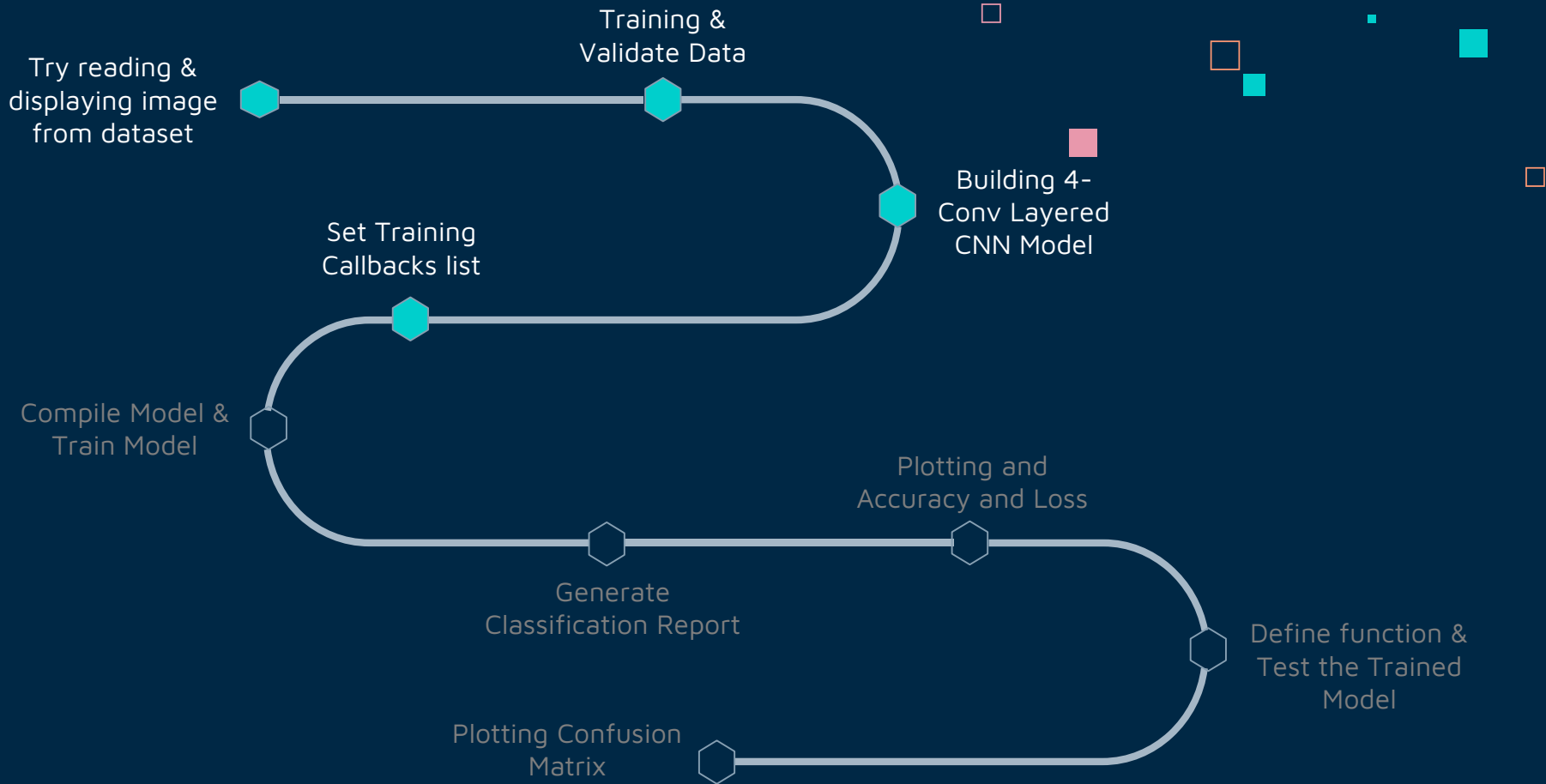


TRAINING THE HUMAN EMOTION RECOGNITION









Try reading & displaying image from dataset

Training & Validate Data

Building 4-Conv Layered

Compile Model & Train Model

```
Epoch 00065: val_acc did not improve from 0.75532
Epoch 66/70
564/564 [=====] - 83s 147ms/step - loss: 0.5145 - acc: 0.8121 - val_loss: 0.8691 - val_acc: 0.7550
Epoch 00066: val_acc did not improve from 0.75532
Epoch 67/70
564/564 [=====] - 83s 148ms/step - loss: 0.5130 - acc: 0.8136 - val_loss: 0.8622 - val_acc: 0.7560
Epoch 00067: val_acc improved from 0.75532 to 0.75598, saving model to .\model.h5
Epoch 68/70
564/564 [=====] - 83s 146ms/step - loss: 0.5006 - acc: 0.8165 - val_loss: 0.8530 - val_acc: 0.7517
Epoch 00068: val_acc did not improve from 0.75598
Epoch 69/70
564/564 [=====] - 83s 147ms/step - loss: 0.4934 - acc: 0.8214 - val_loss: 0.8744 - val_acc: 0.7563
Epoch 00069: val_acc improved from 0.75598 to 0.75632, saving model to .\model.h5
Epoch 70/70
564/564 [=====] - 83s 148ms/step - loss: 0.4864 - acc: 0.8205 - val_loss: 0.9028 - val_acc: 0.7566
Epoch 00070: val_acc improved from 0.75632 to 0.75665, saving model to .\model.h5
```

Test the Trained Model

Plotting Confusion Matrix

Try reading & displaying image from dataset

Training & Validate Data

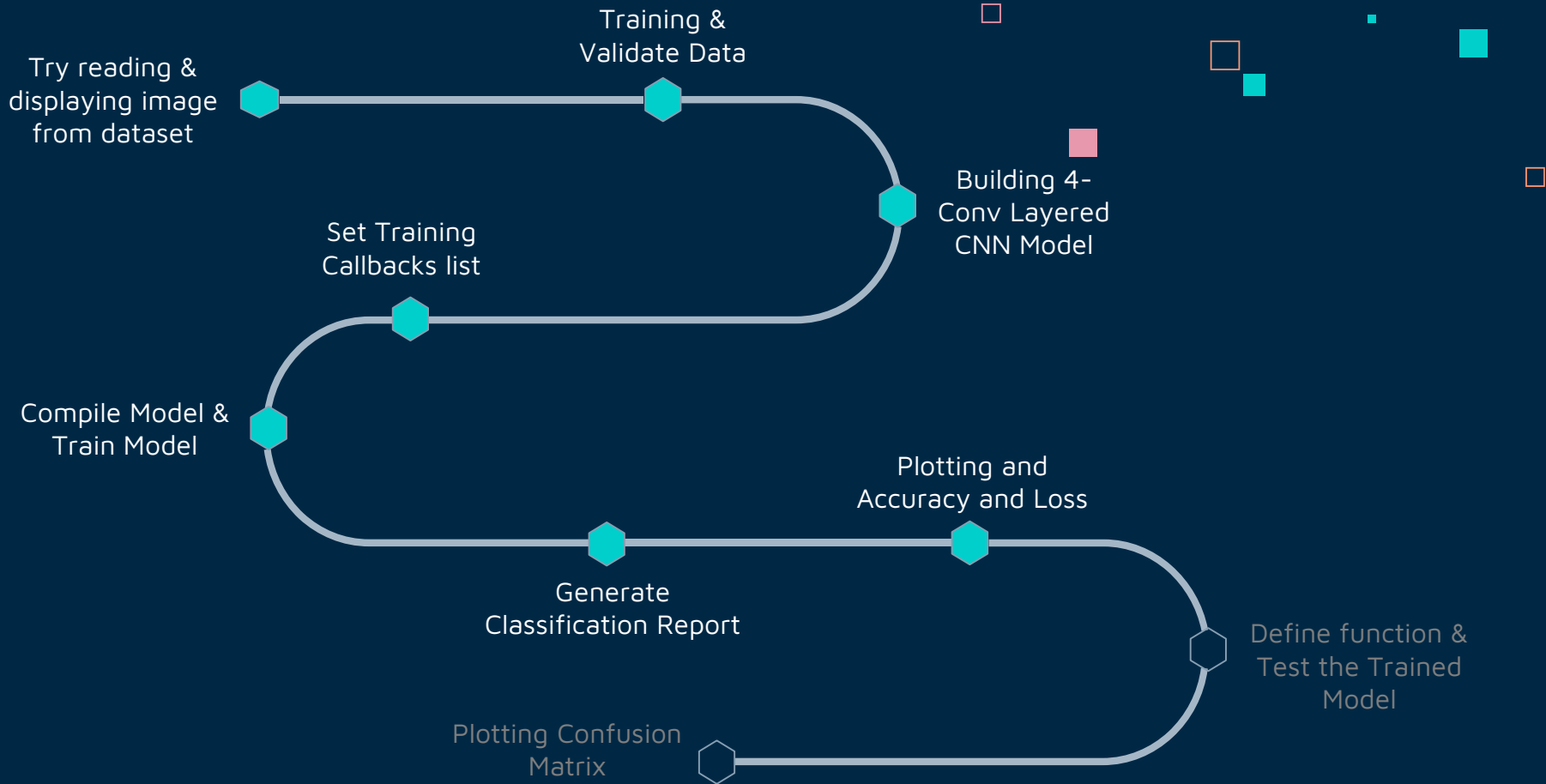
Set Training
Callbacks list

Compile Model &
Train Model

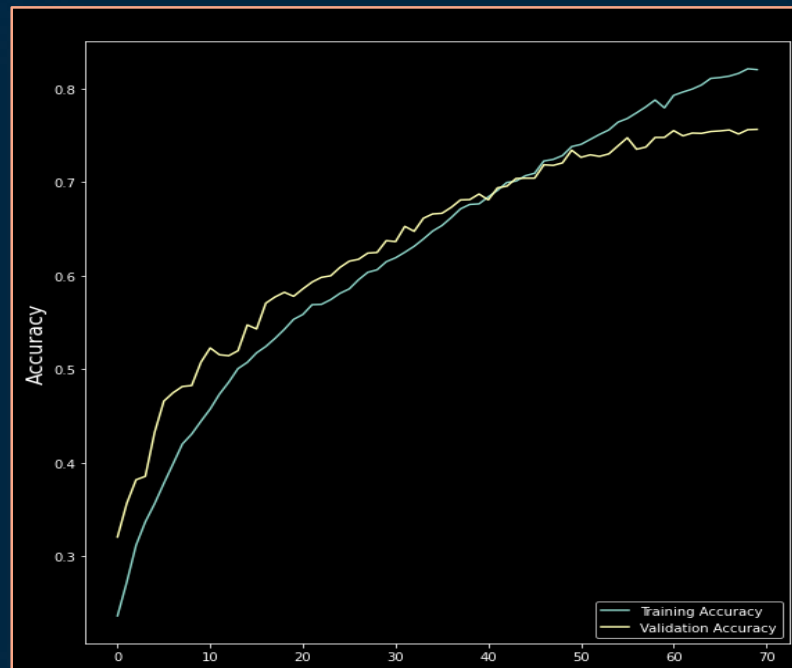
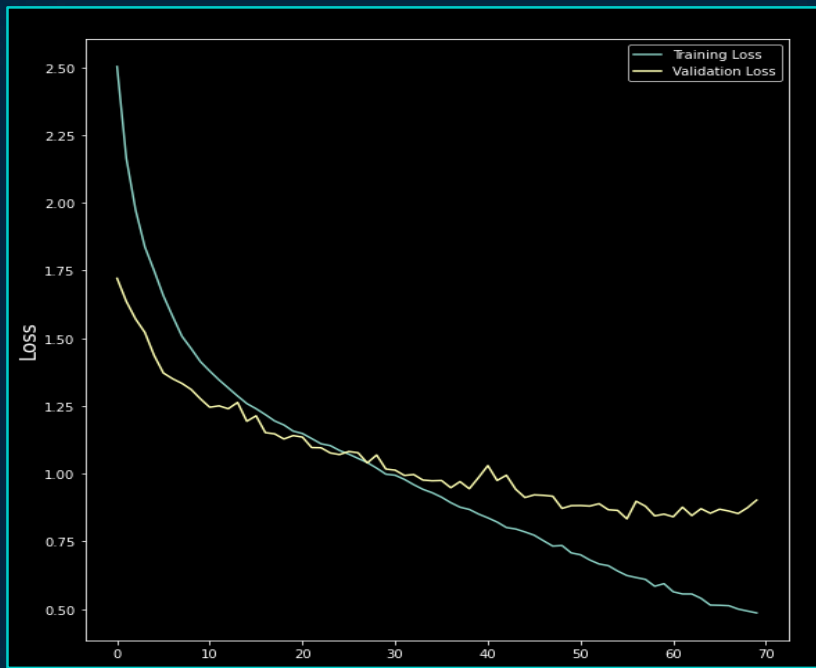
Generate
Classification Report

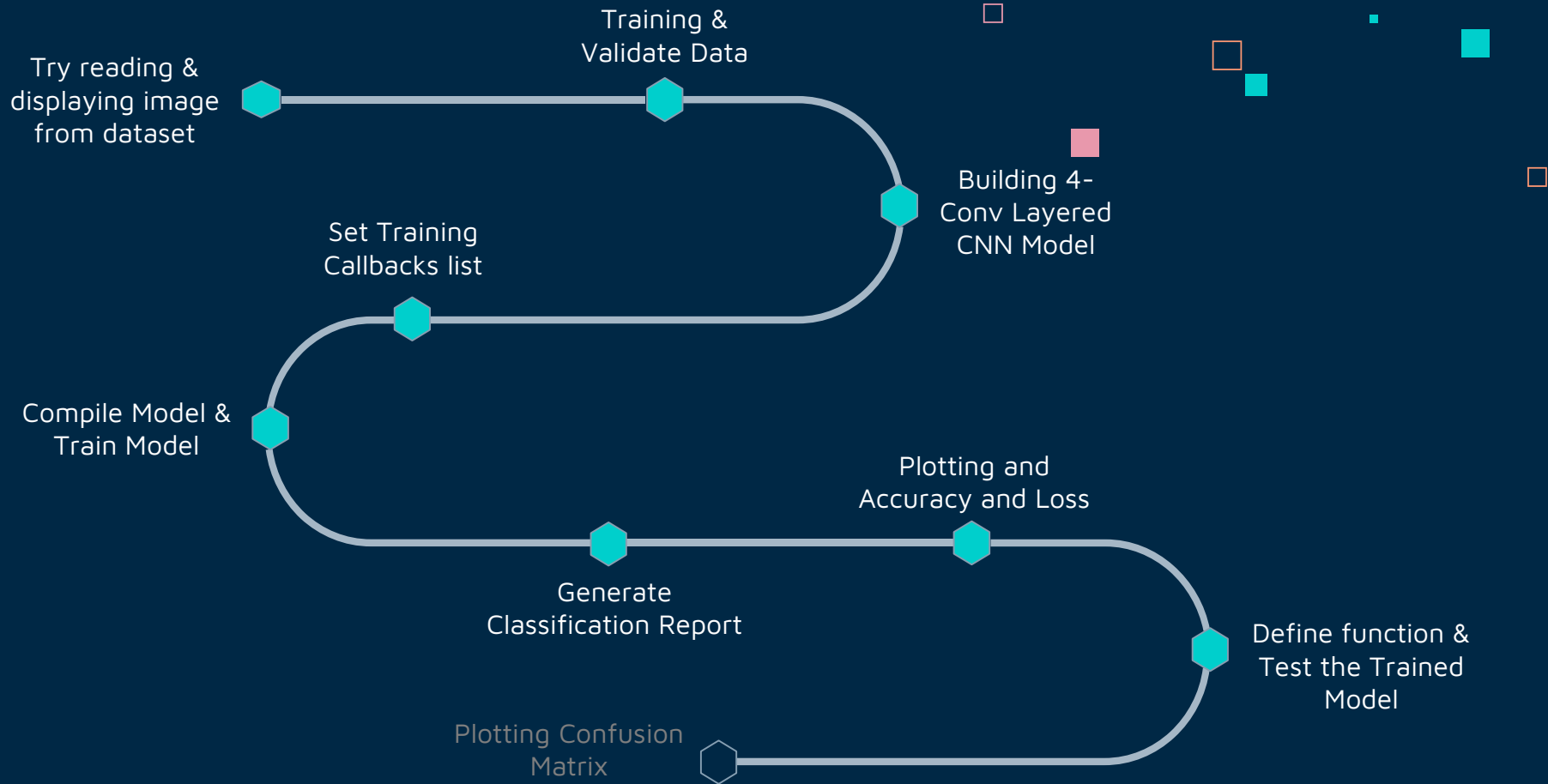
Plotting Confusion
Matrix

	precision	recall	f1-score	support
angry	0.69	0.75	0.72	501
fear	0.78	0.64	0.70	505
happy	0.81	0.84	0.83	504
neutral	0.71	0.79	0.74	504
sad	0.74	0.59	0.66	509
surprise	0.81	0.94	0.87	502
accuracy			0.76	3025
macro avg	0.76	0.76	0.75	3025
weighted avg	0.76	0.76	0.75	3025



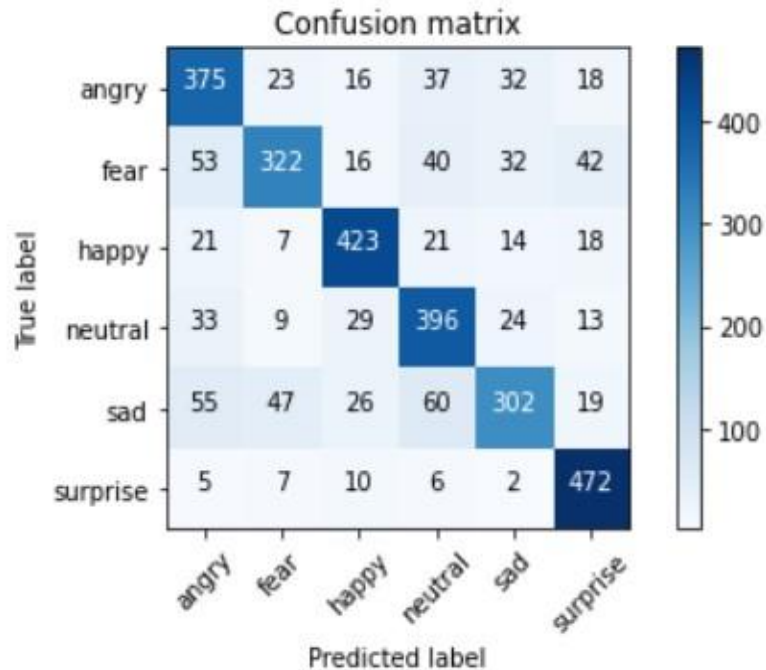
Plotting Accuracy and Loss





Training &
Validate Data

Try reading &
displaying image
from dataset



Building 4-
Conv Layered
CNN Model

Compile Model
Train Model

Plotting and
Accuracy and Loss

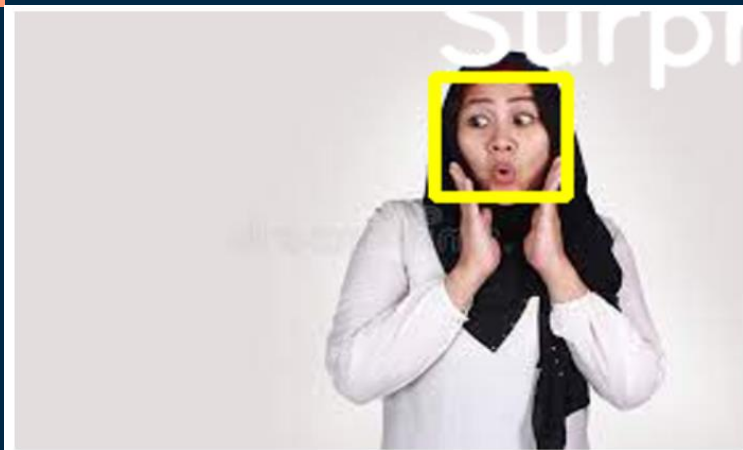
Plotting Confusion
Matrix

Define function &
Test the Trained
Model

RESULT AND CONCLUSION

The background is a dark blue gradient. It is decorated with various geometric elements: small squares in teal, orange, and pink, and thin white vertical lines of varying lengths. Some squares are solid, while others are outlined. The lines are positioned at different heights, creating a sense of depth and movement.

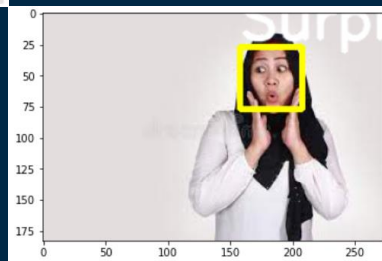
Testing on Static Images

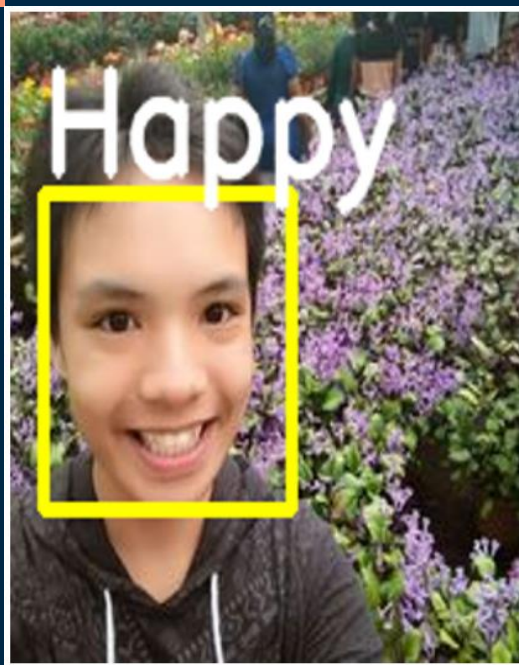


Larger Image

Surprise

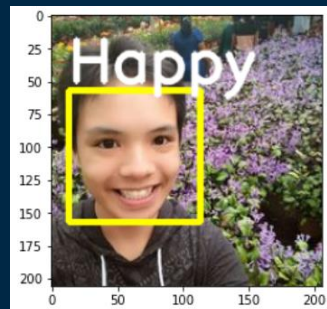
Smaller Image





Larger Image

Happy



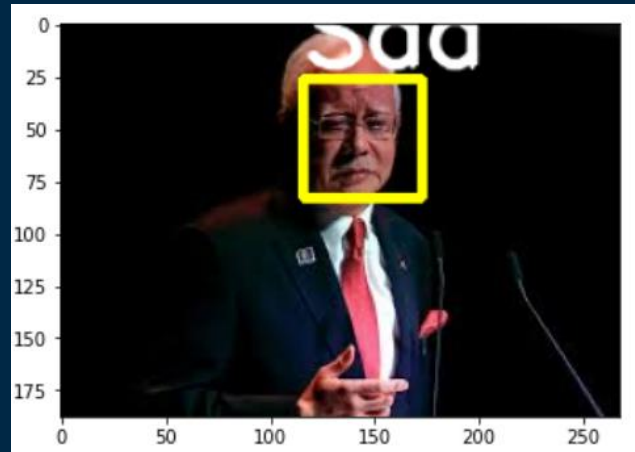
Smaller Image

Testing on Static Images (cont.)

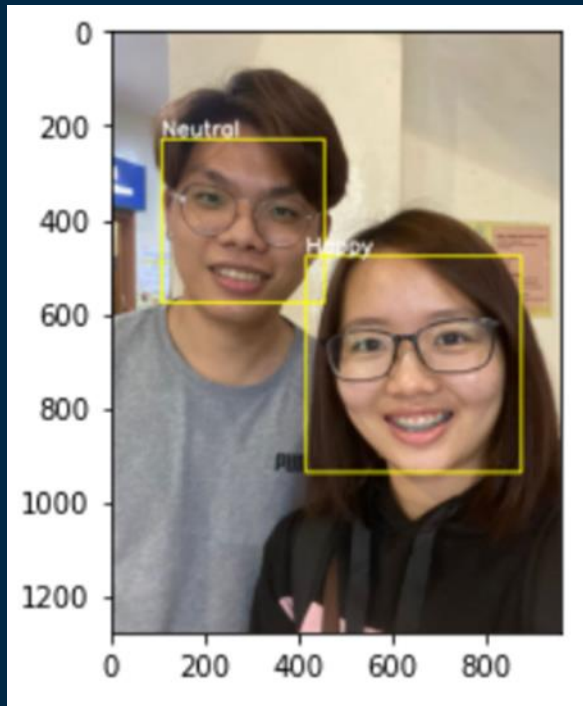


Angry

Sad



Testing on Static Images (cont.)



CONCLUSION



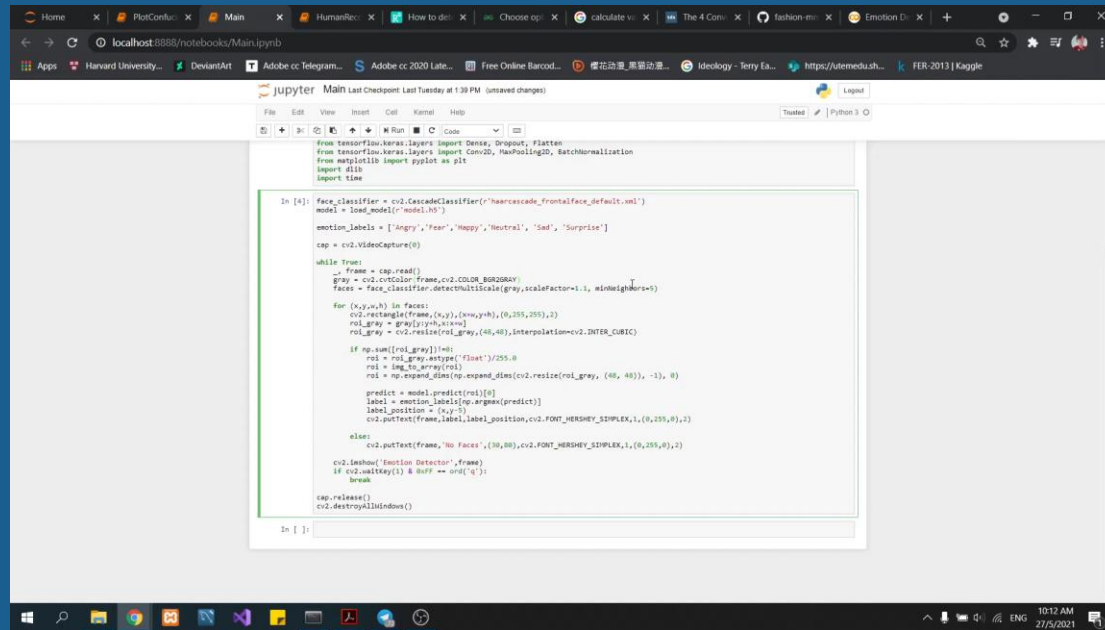
The AI recognizes emotion model can get around **75%** accuracy. There is still room for improving accuracy and efficiency. We hope that we can train the model from scratch by collecting more comprehensive dataset in future.



Sometimes we hide our emotions. What others see on our outside is not always how we are feeling on the inside. This AI still not able to recognize our emotions if we were hiding them. This AI just able to detect basic emotion so, we need to gather more data set to spot the human's micro expression.

LIVE DEMO

1. Download all src files from GitHub.
2. Click run to execute the main.ipynb script.
3. Place your face in front of the webcam.
4. The facial emotion recognizer is ready to use.



The screenshot shows a Jupyter Notebook interface with a browser window at the top displaying the URL `localhost:8888/notebooks/Main.ipynb`. The notebook is titled "Main" and shows the following code:

```
from tensorflow.keras.layers import Dense, Dropout, Flatten
from tensorflow.keras.layers import Conv2D, MaxPooling2D, BatchNormalization
from matplotlib import pyplot as plt
import dlib
import time

In [4]: face_classifier = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
model = load_model('model.h5')
emotion_labels = ['Angry', 'Fear', 'Happy', 'Neutral', 'Sad', 'Surprise']
cap = cv2.VideoCapture(0)

while True:
    frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)

    for (x,y,w,h) in faces:
        cv2.rectangle(frame, (x,y), (x+w,y+h), (0,255,255), 2)
        roi_gray = gray[y:y+h, x:x+w]
        roi_gray = cv2.resize(roi_gray, (40,40), interpolation=cv2.INTER_CUBIC)

        if np.sum([roi_gray])>0:
            roi = roi_gray.astype('float')/255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(np.expand_dims(roi, -1), -1)

            predict = model.predict(roi)[0]
            label = emotion_labels[np.argmax(predict)]
            label_position = (x,y-10)
            cv2.putText(frame, label, label_position, cv2.FONT_HERSHEY_SIMPLEX, 1, (0,255,0), 2)

        else:
            cv2.putText(frame, 'No Faces', (10,80), cv2.FONT_HERSHEY_SIMPLEX, 1, (0,255,0), 2)

    cv2.imshow('Emotion Detector', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

The bottom of the screenshot shows a Windows taskbar with the time 10:12 AM and date 27/5/2021.

For more info 🖐️



TanWeiYin/**Group_J_AI_Project**



youtu.be/UgMNSuaAdL4

