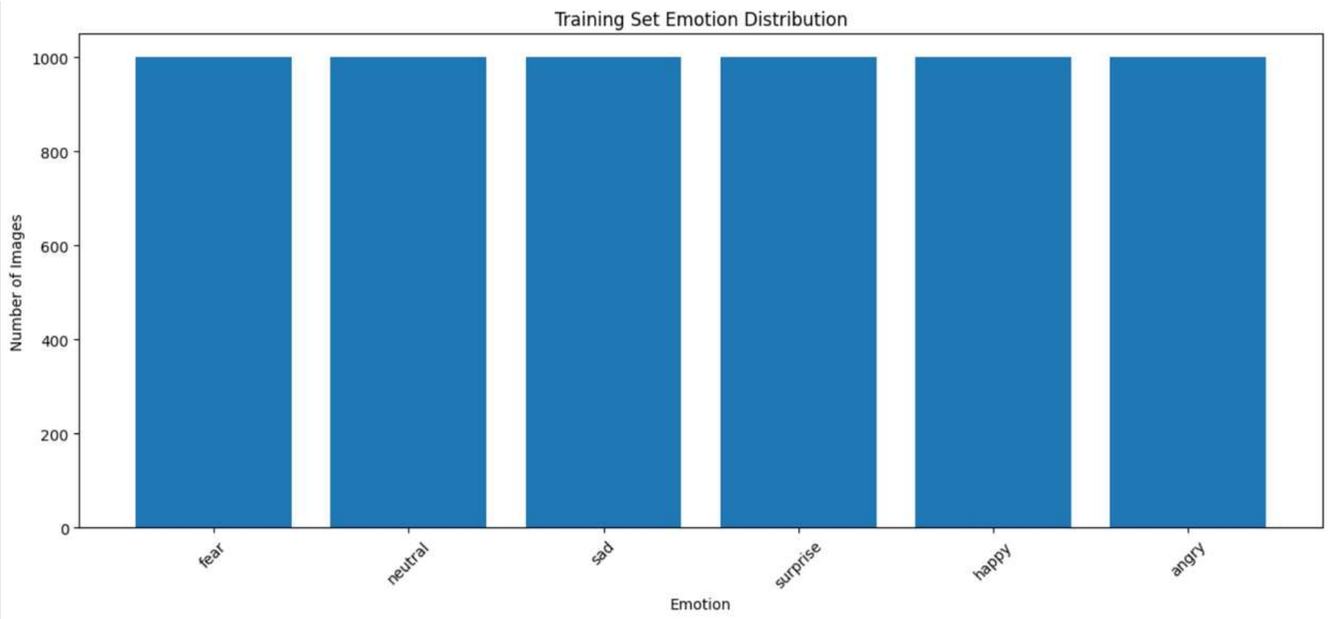
Emotion Recognition of Images

DATASET - FER 2013

FER 2013 (Facial Expression Recognition 2013) is a publicly available dataset used in facial expression recognition tasks. It consists of **6000 grayscale images** of faces, each **48 × 48 pixels in size,** and is classified into six emotion classes: anger, fear, happiness, sadness, surprise, and neutral. It is commonly used to train and evaluate machine learning models for detecting emotions from facial expressions.

PREPROCESSING

- We reduced the number of datasets to 1000 images from each class in order to train the model with high efficiency. We also removed the disgust class to reduce the classes and increase efficiency.




```
Reduced class 'fear' to 1000 images.
Removed class: disgust
Reduced class 'neutral' to 1000 images.
Reduced class 'sad' to 1000 images.
Reduced class 'surprise' to 1000 images.
Reduced class 'happy' to 1000 images.
Reduced class 'angry' to 1000 images.
```

PREPROCESSING

- **Augmentaion :**

Rotation

Width shift

Height shift

Zoom which includes zoom in and zoom out

Crop

Reflection (mirror)

use function :ImageDataGenerator

PREPROCESSING

- **normalizes data**

It normalizes pixel values to range[0,1]

- **split data**

Set aside 20% of the data for validation

PREPROCESSING

Resizes all images to 224x224 pixels. This is a common size for model VGG 16

The number of images that will be in each stage is 64.

the images are in RGB color format (3 color channels - red , green , blue). Alternatively, 'grayscale' could be used for single-channel images.

PREPROCESSING

the size of data after augmentation

Training Dataset

```
Found 4800 images belonging to 6 classes.
```

validation Dataset

```
Found 1200 images belonging to 6 classes.
```

Testing Dataset

```
Found 7067 images belonging to 6 classes.
```

MODDLING - VGG 16

The basic structure of the model after training it on Image Net Dataset

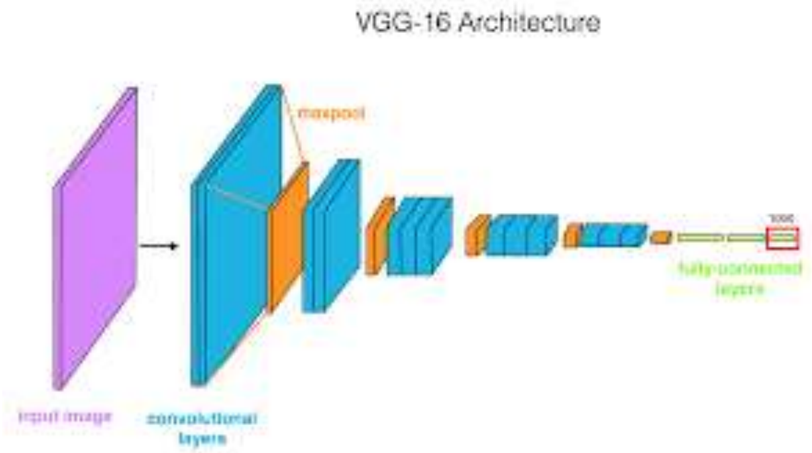
Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
fc1 (Dense)	(None, 4096)	102,764,544
fc2 (Dense)	(None, 4096)	16,781,312
predictions (Dense)	(None, 1000)	4,097,000

IMAGENET

```
Total params: 138,357,544 (527.79 MB)

Trainable params: 138,357,544 (527.79 MB)

Non-trainable params: 0 (0.00 B)
```



FREEZE MODEL

MODDLING -VGG16

Add custom classification layers on top

```
x = vgg16_base.output

x = layers.Flatten()(x)

x = layers.Dense(256, activation='relu')(x)

x = layers.Dropout(0.5)(x)

x = layers.Dense(6, activation='softmax')(x)
```

MODDLING -VGG16

Compile the model

```
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

MODDLING -VGG16

The basic structure of the model after adding our layers.

Layer (type)	Output Shape	Param #
input_layer_1 (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 256)	6,422,784
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 6)	1,542

```
Total params: 21,139,014 (80.64 MB)

Trainable params: 6,424,326 (24.51 MB)

Non-trainable params: 14,714,688 (56.13 MB)
```

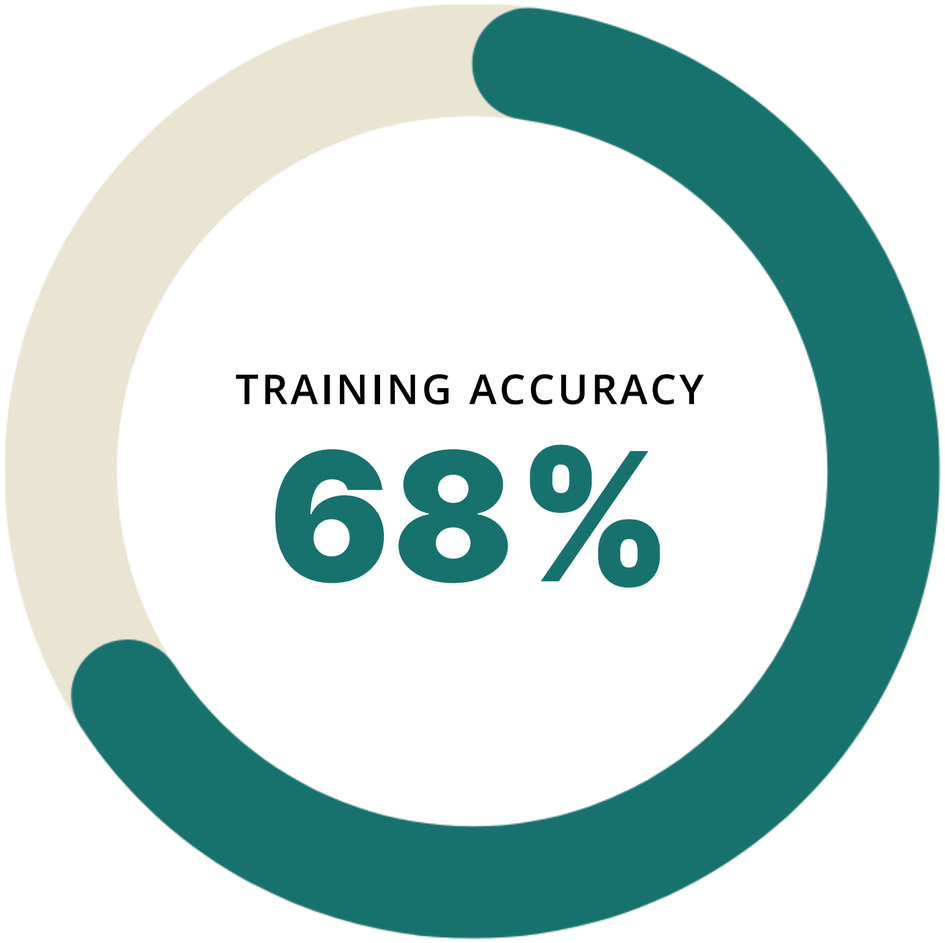
MODDLING -VGG16

Unfreeze the top layers of the convolutional base

Use EarlyStopping in 10 steps

RESULTS

METRIC: ACCURACY

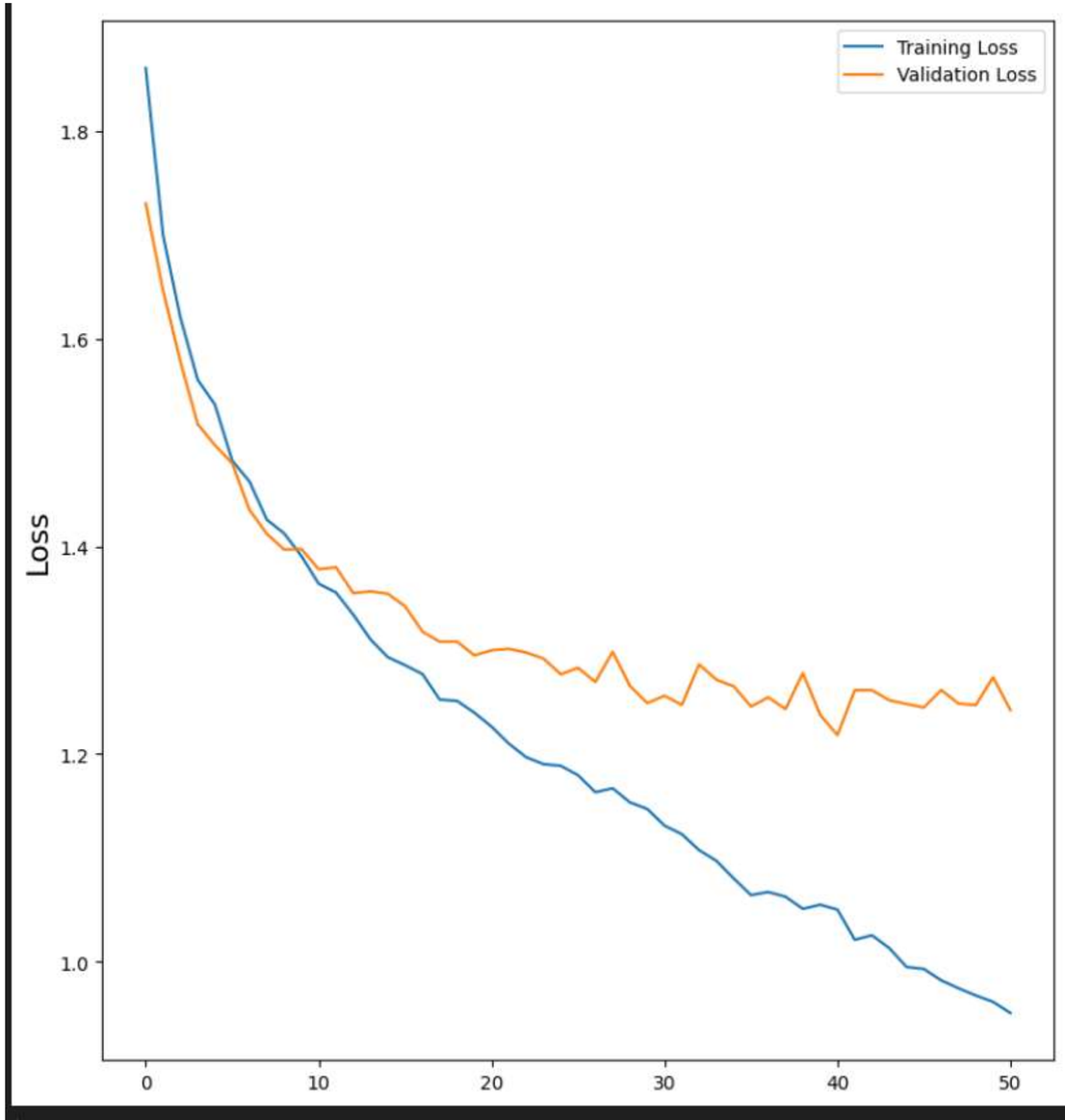


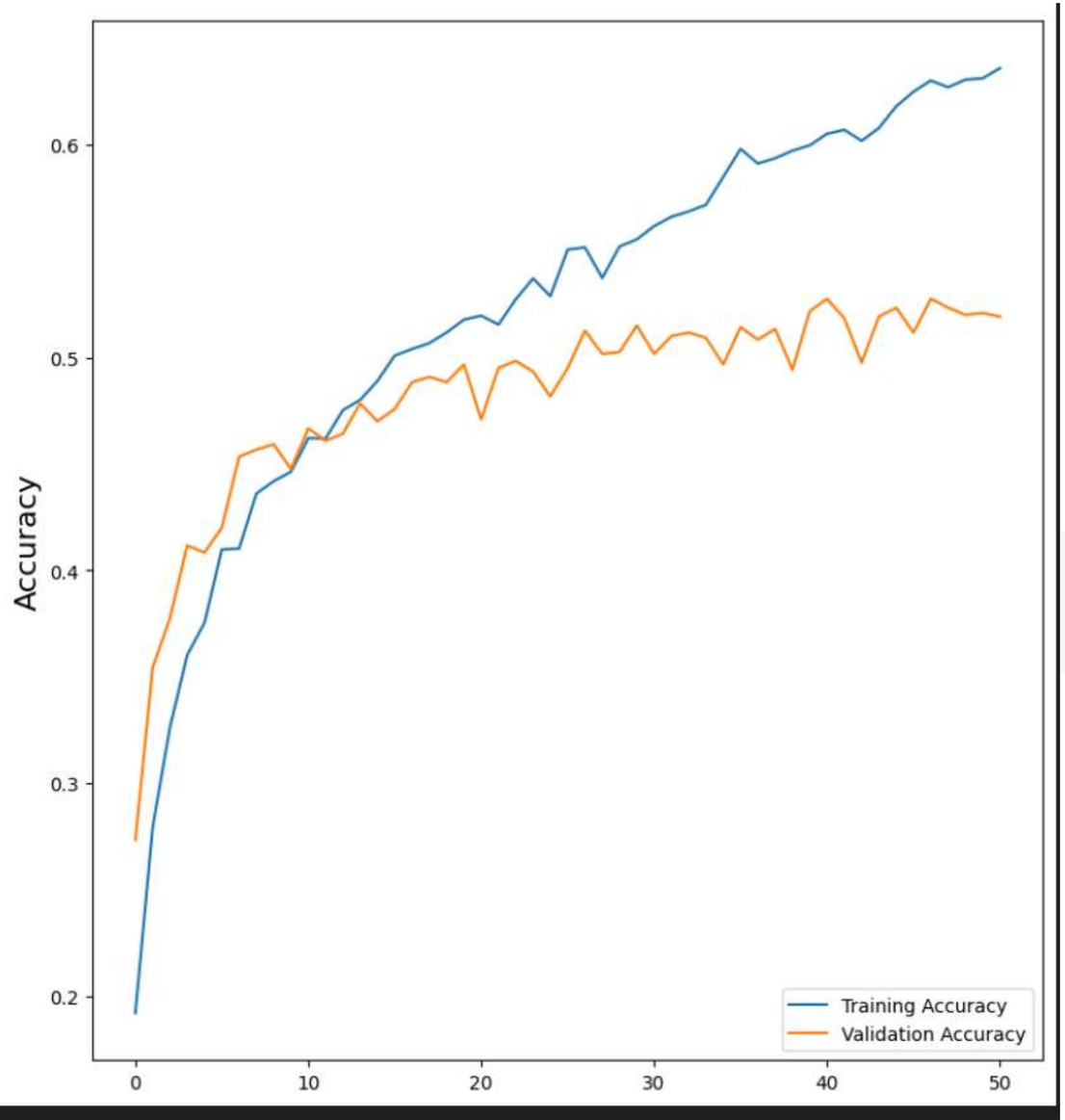
52%	Test Accuracy
1.20	Loss Function

```
test_loss, test_accuracy = model.evaluate(testgen,verbose=1)
```

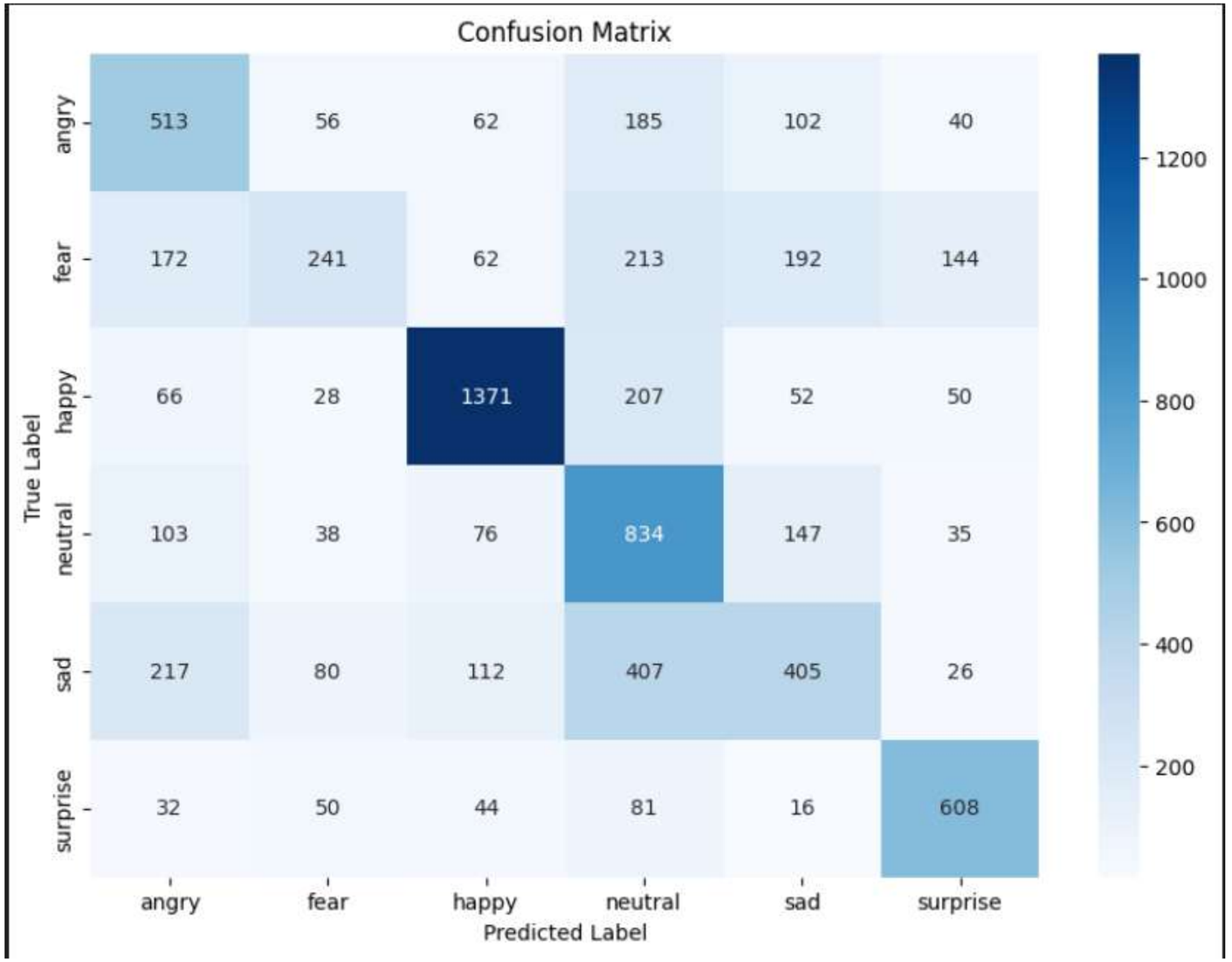
```
154/154 ━━━━━━━━━━━━━━━━━━━━ 58s 266ms/step - accuracy: 0.5265 - loss: 1.2075
```

MODDLING -VGG16





MODDLING -VGG16



MODDLING -VGG16

Streamlit

Facial Emotion Recognition

Upload an image

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files

im122.png 1.6KB

Uploaded Image

Detect Emotion

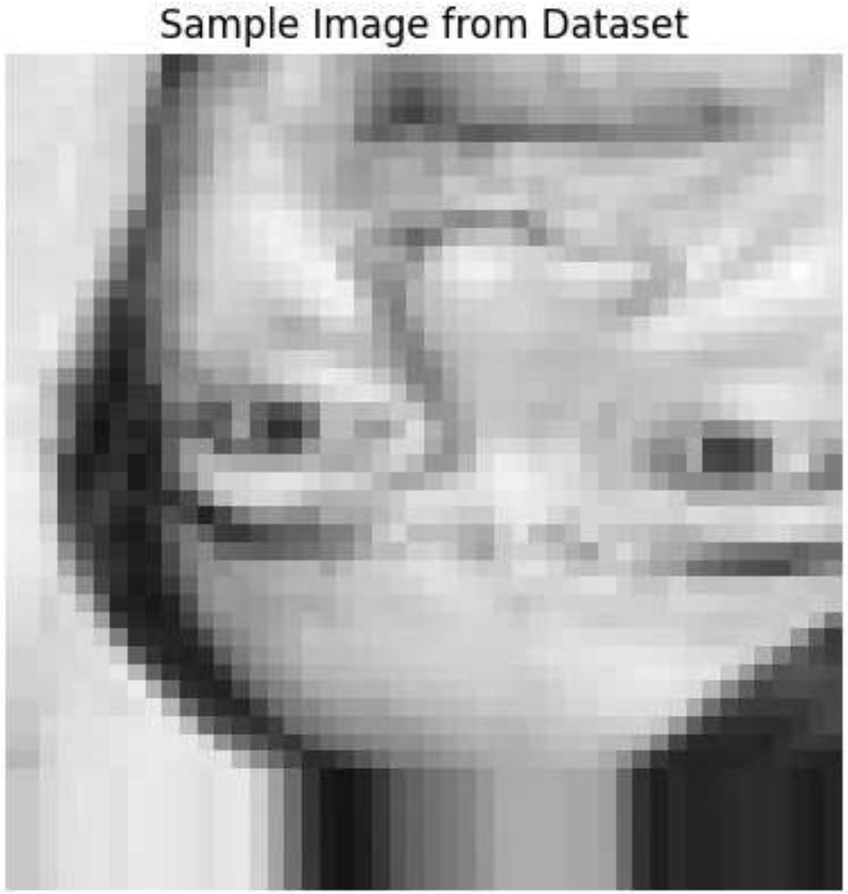
Detected Emotion: Angry

Confidence: 100.00%

CHALLENGES

THAT OVERWHELMED AT WORK

1 Dataset and Quality of images



CHALLENGES

THAT OVERWHELMED AT WORK

Neutrality class

sad

neutra

l

CHALLENGES

THAT OVERWHELMED AT WORK

model

The model VGG-16 must have color data and our data set is black and white.

CHALLENGES

THAT OVERWHELMED AT WORK

4 Dataset

The number of datasets is huge, so we had to take only a part of it.

CHALLENGES

THAT OVERWHELMED AT WORK

4 AI photo

We noticed that there are images generated by artificial intelligence, which means that they are not real.

SOLUTIONS

SOLUTIONS OF THE PROBLEMS

1- Improving image quality

2- Using a better model than VGG-16

3- Creating a model to classify images if they are generated by artificial intelligence, it deletes them from the dataset, and if they are real, they remain

4- Training the model on a small part of the data and then training it on the rest of the existing data

THANK YOU !

Team members

Aman Faisal - leader of team

Ala'a Motawe

Huda Abu Safa

Roaa iashash

Tasneem Al-Houwari