

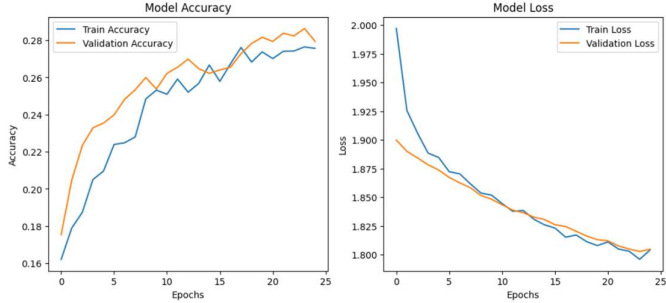
OUTPUT:

```
Found 6360 images belonging to 7 classes.
Found 2731 images belonging to 7 classes.
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5
58889256/58889256 ————— 2s 0us/step
Model: "functional"
```

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
global_average_pooling2d (GlobalAveragePooling2D)	(None, 512)	0
dense (Dense)	(None, 128)	65,664
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 7)	903

```
Total params: 14,781,255 (56.39 MB)
Trainable params: 66,567 (260.03 KB)
Non-trainable params: 14,714,688 (56.13 MB)
/usr/local/lib/python3.12/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:121: UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `self._warn_if_super_not_called()`
Epoch 1/25
199/199 ————— 0s 501ms/step - accuracy: 0.1559 - loss: 2.0339WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This file format is considered legacy.
199/199 ————— 157s 711ms/step - accuracy: 0.1559 - loss: 2.0337 - val_accuracy: 0.1754 - val_loss: 1.8999
Epoch 2/25
Epoch 3/25
Epoch 4/25
Epoch 5/25
Epoch 6/25
Epoch 7/25
Epoch 8/25
Epoch 9/25
Epoch 10/25
Epoch 11/25
Epoch 12/25
Epoch 13/25
Epoch 14/25
Epoch 15/25
Epoch 16/25
Epoch 17/25
Epoch 18/25
Epoch 19/25
Epoch 20/25
Epoch 21/25
Epoch 22/25
```

```
Epoch 20/25: 199/199 - 0s 431ms/step - accuracy: 0.2772 - loss: 1.7999WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
Epoch 21/25: 199/199 - 0s 431ms/step - accuracy: 0.2657 - loss: 1.8154WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
Epoch 22/25: 199/199 - 0s 429ms/step - accuracy: 0.2727 - loss: 1.8065WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
Epoch 23/25: 199/199 - 0s 429ms/step - accuracy: 0.2757 - loss: 1.8024WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
Epoch 24/25: 199/199 - 0s 434ms/step - accuracy: 0.2696 - loss: 1.7960WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
Epoch 25/25: 199/199 - 0s 434ms/step - accuracy: 0.2696 - loss: 1.7960WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
```



```
Epoch 1/5: 199/199 - 124s 569ms/step - accuracy: 0.2846 - loss: 1.7756 - val_accuracy: 0.3354 - val_loss: 1.7010
Epoch 2/5: 199/199 - 104s 525ms/step - accuracy: 0.3371 - loss: 1.7136 - val_accuracy: 0.3523 - val_loss: 1.6620
Epoch 3/5: 199/199 - 104s 523ms/step - accuracy: 0.3501 - loss: 1.6719 - val_accuracy: 0.3808 - val_loss: 1.6197
Epoch 4/5: 199/199 - 105s 527ms/step - accuracy: 0.3726 - loss: 1.6330 - val_accuracy: 0.3881 - val_loss: 1.5884
Epoch 5/5: 199/199 - 105s 536ms/step - accuracy: 0.3672 - loss: 1.6189 - val_accuracy: 0.3940 - val_loss: 1.5588
WARNING:absl:You are saving your model as an HDF5 file via 'model.save()' or 'keras.saving.save_model(model)'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save(format='h5')'.
```

Predicted Emotion: Angry

TEST IMAGE:

