

Model Development Phase Template

Date	11 July 2024
Team ID	SWTID1720537811
Project Title	Dog Breed Identification using Transfer Learning
Maximum Marks	10 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

Initial Model Training Code (5 marks):

```
vgg16.fit(train_generator,validation_data = test_generator,epochs=10 )
```

```
resnet.fit(train_generator,validation_data = test_generator,epochs=10 )
```

```
inception.fit(train_generator,validation_data = test_generator,epochs=10 )
```

```
xception.fit(train_generator,validation_data = test_generator,epochs=10 )
```

Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics																																																																		
Model 1	<p>Model: "model"</p> <table> <tr> <th>Layer (type)</th><th>Output Shape</th><th>Param #</th></tr> <tr> <td>input_1 (InputLayer)</td><td>[(None, 224, 224, 3)]</td><td>0</td></tr> <tr> <td>block1_conv1 (Conv2D)</td><td>(None, 224, 224, 64)</td><td>1792</td></tr> <tr> <td>block1_conv2 (Conv2D)</td><td>(None, 224, 224, 64)</td><td>36928</td></tr> <tr> <td>block1_pool (MaxPooling2D)</td><td>(None, 112, 112, 64)</td><td>0</td></tr> <tr> <td>block2_conv1 (Conv2D)</td><td>(None, 112, 112, 128)</td><td>73856</td></tr> <tr> <td>block2_conv2 (Conv2D)</td><td>(None, 112, 112, 128)</td><td>147584</td></tr> <tr> <td>block2_pool (MaxPooling2D)</td><td>(None, 56, 56, 128)</td><td>0</td></tr> <tr> <td>block3_conv1 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>295168</td></tr> <tr> <td>block3_conv2 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>590080</td></tr> <tr> <td>block3_conv3 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>590080</td></tr> <tr> <td>block3_pool (MaxPooling2D)</td><td>(None, 28, 28, 256)</td><td>0</td></tr> <tr> <td>block4_conv1 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>1180160</td></tr> <tr> <td>block4_conv2 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>2359808</td></tr> <tr> <td>block4_conv3 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>2359808</td></tr> <tr> <td>block4_pool (MaxPooling2D)</td><td>(None, 14, 14, 512)</td><td>0</td></tr> <tr> <td>block5_conv1 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2359808</td></tr> <tr> <td>block5_conv2 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2359808</td></tr> <tr> <td>block5_conv3 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2359808</td></tr> <tr> <td>block5_pool (MaxPooling2D)</td><td>(None, 7, 7, 512)</td><td>0</td></tr> <tr> <td>flatten (Flatten)</td><td>(None, 25088)</td><td>0</td></tr> <tr> <td>dense (Dense)</td><td>(None, 8)</td><td>200712</td></tr> </table> <p>===== Total params: 14915400 (56.90 MB) Trainable params: 200712 (784.03 KB) Non-trainable params: 14714688 (56.13 MB)</p>	Layer (type)	Output Shape	Param #	input_1 (InputLayer)	[(None, 224, 224, 3)]	0	block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792	block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928	block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0	block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856	block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584	block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0	block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168	block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080	block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080	block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0	block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160	block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808	block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808	block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0	block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808	block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808	block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808	block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0	flatten (Flatten)	(None, 25088)	0	dense (Dense)	(None, 8)	200712	<pre>sgm1.fit(train_generator, validation_data = test_generator, epochs=10)</pre> <pre>Epoch 1/10 12/12 [=====] - 27s 1s/step - loss: 2.2112 - accuracy: 0.2580 - val_loss: 1.4055 - val_accuracy: 0.4286 Epoch 2/10 12/12 [=====] - 9s 747ms/step - loss: 1.2505 - accuracy: 0.5790 - val_loss: 0.9771 - val_accuracy: 0.7143 Epoch 3/10 12/12 [=====] - 8s 679ms/step - loss: 0.8296 - accuracy: 0.7181 - val_loss: 0.9401 - val_accuracy: 0.7143 Epoch 4/10 12/12 [=====] - 10s 838ms/step - loss: 0.6643 - accuracy: 0.7686 - val_loss: 0.6389 - val_accuracy: 0.8571 Epoch 5/10 12/12 [=====] - 10s 845ms/step - loss: 0.5785 - accuracy: 0.8191 - val_loss: 0.5487 - val_accuracy: 0.8571 Epoch 6/10 12/12 [=====] - 9s 718ms/step - loss: 0.4333 - accuracy: 0.8777 - val_loss: 0.2663 - val_accuracy: 0.8571 Epoch 7/10 12/12 [=====] - 10s 803ms/step - loss: 0.4174 - accuracy: 0.9043 - val_loss: 0.2756 - val_accuracy: 1.0000 Epoch 8/10 12/12 [=====] - 10s 838ms/step - loss: 0.3417 - accuracy: 0.8989 - val_loss: 0.2725 - val_accuracy: 0.8571 Epoch 9/10 12/12 [=====] - 8s 682ms/step - loss: 0.3409 - accuracy: 0.9069 - val_loss: 0.3049 - val_accuracy: 1.0000 Epoch 10/10 12/12 [=====] - 9s 766ms/step - loss: 0.2907 - accuracy: 0.9202 - val_loss: 0.3160 - val_accuracy: 1.0000 tf.keras.callbacks.History at 0x7ed558f6e900</pre>
Layer (type)	Output Shape	Param #																																																																		
input_1 (InputLayer)	[(None, 224, 224, 3)]	0																																																																		
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792																																																																		
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928																																																																		
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0																																																																		
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856																																																																		
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584																																																																		
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0																																																																		
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168																																																																		
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080																																																																		
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080																																																																		
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0																																																																		
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160																																																																		
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808																																																																		
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808																																																																		
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0																																																																		
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808																																																																		
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808																																																																		
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808																																																																		
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0																																																																		
flatten (Flatten)	(None, 25088)	0																																																																		
dense (Dense)	(None, 8)	200712																																																																		
Model 2	<table> <tr> <td>conv5_block2_2_conv (Conv2 (None, 7, 7, 512)</td><td>2359808</td><td>['conv5_block2_1_relu@0][0]</td></tr> <tr> <td>conv5_block2_3_bn (BatchNorm (None, 7, 7, 512)</td><td>2048</td><td>['conv5_block2_3_conv@0][0]</td></tr> <tr> <td>conv5_block2_3_relu (Activ (None, 7, 7, 512)</td><td>0</td><td>['conv5_block2_3_bn@0][0]</td></tr> <tr> <td>conv5_block2_3_conv (Conv2 (None, 7, 7, 2048)</td><td>1858624</td><td>['conv5_block2_3_relu@0][0]</td></tr> <tr> <td>conv5_block2_3_bn (BatchNorm (None, 7, 7, 2048)</td><td>8192</td><td>['conv5_block2_3_conv@0][0]</td></tr> <tr> <td>conv5_block2_add (Add) (None, 7, 7, 2048)</td><td>0</td><td>['conv5_block2_out@0][0], ['conv5_block2_3_bn@0][0]</td></tr> <tr> <td>conv5_block2_out (Activati (None, 7, 7, 2048)</td><td>0</td><td>['conv5_block2_add@0][0]</td></tr> <tr> <td>conv5_block3_1_conv (Conv2 (None, 7, 7, 512)</td><td>1849808</td><td>['conv5_block2_out@0][0]</td></tr> <tr> <td>conv5_block3_1_bn (BatchNorm (None, 7, 7, 512)</td><td>2048</td><td>['conv5_block3_1_conv@0][0]</td></tr> <tr> <td>conv5_block3_1_relu (Activ (None, 7, 7, 512)</td><td>0</td><td>['conv5_block3_1_bn@0][0]</td></tr> <tr> <td>conv5_block3_2_conv (Conv2 (None, 7, 7, 512)</td><td>2359808</td><td>['conv5_block3_1_relu@0][0]</td></tr> <tr> <td>conv5_block3_3_bn (BatchNorm (None, 7, 7, 512)</td><td>2048</td><td>['conv5_block3_2_conv@0][0]</td></tr> <tr> <td>conv5_block3_3_relu (Activ (None, 7, 7, 512)</td><td>0</td><td>['conv5_block3_3_bn@0][0]</td></tr> <tr> <td>conv5_block3_3_conv (Conv2 (None, 7, 7, 2048)</td><td>1858624</td><td>['conv5_block3_3_relu@0][0]</td></tr> <tr> <td>conv5_block3_3_bn (BatchNorm (None, 7, 7, 2048)</td><td>8192</td><td>['conv5_block3_3_conv@0][0]</td></tr> <tr> <td>conv5_block3_add (Add) (None, 7, 7, 2048)</td><td>0</td><td>['conv5_block3_out@0][0], ['conv5_block3_3_bn@0][0]</td></tr> <tr> <td>conv5_block3_out (Activati (None, 7, 7, 2048)</td><td>0</td><td>['conv5_block3_add@0][0]</td></tr> <tr> <td>flatten_1 (Flatten) (None, 100352)</td><td>0</td><td>['conv5_block3_out@0][0]</td></tr> <tr> <td>dense_1 (Dense) (None, 8)</td><td>802824</td><td>['flatten_1@0][0]</td></tr> </table> <p>===== Total params: 24300536 (53.04 MB) Trainable params: 803224 (1.06 MB)</p>	conv5_block2_2_conv (Conv2 (None, 7, 7, 512)	2359808	['conv5_block2_1_relu@0][0]	conv5_block2_3_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block2_3_conv@0][0]	conv5_block2_3_relu (Activ (None, 7, 7, 512)	0	['conv5_block2_3_bn@0][0]	conv5_block2_3_conv (Conv2 (None, 7, 7, 2048)	1858624	['conv5_block2_3_relu@0][0]	conv5_block2_3_bn (BatchNorm (None, 7, 7, 2048)	8192	['conv5_block2_3_conv@0][0]	conv5_block2_add (Add) (None, 7, 7, 2048)	0	['conv5_block2_out@0][0], ['conv5_block2_3_bn@0][0]	conv5_block2_out (Activati (None, 7, 7, 2048)	0	['conv5_block2_add@0][0]	conv5_block3_1_conv (Conv2 (None, 7, 7, 512)	1849808	['conv5_block2_out@0][0]	conv5_block3_1_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block3_1_conv@0][0]	conv5_block3_1_relu (Activ (None, 7, 7, 512)	0	['conv5_block3_1_bn@0][0]	conv5_block3_2_conv (Conv2 (None, 7, 7, 512)	2359808	['conv5_block3_1_relu@0][0]	conv5_block3_3_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block3_2_conv@0][0]	conv5_block3_3_relu (Activ (None, 7, 7, 512)	0	['conv5_block3_3_bn@0][0]	conv5_block3_3_conv (Conv2 (None, 7, 7, 2048)	1858624	['conv5_block3_3_relu@0][0]	conv5_block3_3_bn (BatchNorm (None, 7, 7, 2048)	8192	['conv5_block3_3_conv@0][0]	conv5_block3_add (Add) (None, 7, 7, 2048)	0	['conv5_block3_out@0][0], ['conv5_block3_3_bn@0][0]	conv5_block3_out (Activati (None, 7, 7, 2048)	0	['conv5_block3_add@0][0]	flatten_1 (Flatten) (None, 100352)	0	['conv5_block3_out@0][0]	dense_1 (Dense) (None, 8)	802824	['flatten_1@0][0]	<pre>sgm2.fit(train_generator, validation_data = test_generator, epochs=10)</pre> <pre>Epoch 1/10 12/12 [=====] - 17s 941ms/step - loss: 8.6675 - accuracy: 0.1416 - val_loss: 6.9575 - val_accuracy: 0.1429 Epoch 2/10 12/12 [=====] - 8s 643ms/step - loss: 6.9999 - accuracy: 0.1356 - val_loss: 3.4116 - val_accuracy: 0.4286 Epoch 3/10 12/12 [=====] - 8s 636ms/step - loss: 4.2426 - accuracy: 0.1862 - val_loss: 2.2663 - val_accuracy: 0.1429 Epoch 4/10 12/12 [=====] - 9s 778ms/step - loss: 2.8612 - accuracy: 0.2207 - val_loss: 2.5258 - val_accuracy: 0.2857 Epoch 5/10 12/12 [=====] - 9s 750ms/step - loss: 2.5577 - accuracy: 0.2340 - val_loss: 1.5863 - val_accuracy: 0.2857 Epoch 6/10 12/12 [=====] - 8s 652ms/step - loss: 2.2711 - accuracy: 0.2048 - val_loss: 1.5730 - val_accuracy: 0.5714 Epoch 7/10 12/12 [=====] - 9s 774ms/step - loss: 2.2721 - accuracy: 0.2500 - val_loss: 1.4668 - val_accuracy: 0.4286 Epoch 8/10 12/12 [=====] - 9s 776ms/step - loss: 2.0657 - accuracy: 0.2608 - val_loss: 1.5892 - val_accuracy: 0.4286 Epoch 9/10 12/12 [=====] - 8s 646ms/step - loss: 1.9804 - accuracy: 0.2926 - val_loss: 1.5387 - val_accuracy: 0.4286 Epoch 10/10 12/12 [=====] - 9s 766ms/step - loss: 2.7614 - accuracy: 0.1835 - val_loss: 2.1361 - val_accuracy: 0.4286 tf.keras.callbacks.History at 0x7ed558f6e900</pre>									
conv5_block2_2_conv (Conv2 (None, 7, 7, 512)	2359808	['conv5_block2_1_relu@0][0]																																																																		
conv5_block2_3_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block2_3_conv@0][0]																																																																		
conv5_block2_3_relu (Activ (None, 7, 7, 512)	0	['conv5_block2_3_bn@0][0]																																																																		
conv5_block2_3_conv (Conv2 (None, 7, 7, 2048)	1858624	['conv5_block2_3_relu@0][0]																																																																		
conv5_block2_3_bn (BatchNorm (None, 7, 7, 2048)	8192	['conv5_block2_3_conv@0][0]																																																																		
conv5_block2_add (Add) (None, 7, 7, 2048)	0	['conv5_block2_out@0][0], ['conv5_block2_3_bn@0][0]																																																																		
conv5_block2_out (Activati (None, 7, 7, 2048)	0	['conv5_block2_add@0][0]																																																																		
conv5_block3_1_conv (Conv2 (None, 7, 7, 512)	1849808	['conv5_block2_out@0][0]																																																																		
conv5_block3_1_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block3_1_conv@0][0]																																																																		
conv5_block3_1_relu (Activ (None, 7, 7, 512)	0	['conv5_block3_1_bn@0][0]																																																																		
conv5_block3_2_conv (Conv2 (None, 7, 7, 512)	2359808	['conv5_block3_1_relu@0][0]																																																																		
conv5_block3_3_bn (BatchNorm (None, 7, 7, 512)	2048	['conv5_block3_2_conv@0][0]																																																																		
conv5_block3_3_relu (Activ (None, 7, 7, 512)	0	['conv5_block3_3_bn@0][0]																																																																		
conv5_block3_3_conv (Conv2 (None, 7, 7, 2048)	1858624	['conv5_block3_3_relu@0][0]																																																																		
conv5_block3_3_bn (BatchNorm (None, 7, 7, 2048)	8192	['conv5_block3_3_conv@0][0]																																																																		
conv5_block3_add (Add) (None, 7, 7, 2048)	0	['conv5_block3_out@0][0], ['conv5_block3_3_bn@0][0]																																																																		
conv5_block3_out (Activati (None, 7, 7, 2048)	0	['conv5_block3_add@0][0]																																																																		
flatten_1 (Flatten) (None, 100352)	0	['conv5_block3_out@0][0]																																																																		
dense_1 (Dense) (None, 8)	802824	['flatten_1@0][0]																																																																		

Model 3

```

inception.summary()
batch_normalization_87 (Batch Normalization) (None, 8, 8, 184) 1152 ['conv2d_87[0][0]']
batch_normalization_88 (Batch Normalization) (None, 8, 8, 184) 1152 ['conv2d_88[0][0]']
batch_normalization_91 (Batch Normalization) (None, 8, 8, 184) 1152 ['conv2d_91[0][0]']
batch_normalization_92 (Batch Normalization) (None, 8, 8, 184) 1152 ['conv2d_92[0][0]']
conv2d_93 (Conv2D) (None, 8, 8, 192) 391216 ['average_pooling2d_8[0][0]']
batch_normalization_85 (Batch Normalization) (None, 8, 8, 128) 960 ['conv2d_85[0][0]']
activation_87 (Activation) (None, 8, 8, 184) 0 ['batch_normalization_87[0][0]']
activation_88 (Activation) (None, 8, 8, 184) 0 ['batch_normalization_88[0][0]']
activation_91 (Activation) (None, 8, 8, 184) 0 ['batch_normalization_91[0][0]']
activation_92 (Activation) (None, 8, 8, 184) 0 ['batch_normalization_92[0][0]']
batch_normalization_93 (Batch Normalization) (None, 8, 8, 192) 576 ['conv2d_93[0][0]']
activation_85 (Activation) (None, 8, 8, 128) 0 ['batch_normalization_85[0][0]']
mixed_91 (Concatenate) (None, 8, 8, 768) 0 ['activation_87[0][0]', 'activation_88[0][0]']
concatenate_1 (Concatenate) (None, 8, 8, 768) 0 ['activation_91[0][0]', 'activation_92[0][0]']
activation_93 (Activation) (None, 8, 8, 192) 0 ['batch_normalization_93[0][0]']
mixed_92 (Concatenate) (None, 8, 8, 2048) 0 ['activation_85[0][0]', 'mixed_91[0][0]', 'concatenate_1[0][0]', 'activation_93[0][0]']
flatten_2 (Flatten) (None, 131072) 0 ['mixed_92[0][0]']
dense_2 (Dense) (None, 8) 1048594 ['flatten_2[0][0]']

Total params: 22851368 (87.17 MB)
Trainable params: 22851368 (87.84 MB)
Non-trainable params: 34432 (134.58 KB)

```

```

***
inception.fit(train_generator, validation_data = test_generator, epochs=10 )

Epoch 1/10: ..... - 74s 2s/step - loss: 18.1743 - accuracy: 0.2128 - val_loss: 15236.6914 - val_accuracy: 0.1429
Epoch 2/10: ..... - 14s 1s/step - loss: 8.7789 - accuracy: 0.1941 - val_loss: 118663.7734 - val_accuracy: 0.1429
Epoch 3/10: ..... - 14s 1s/step - loss: 3.5277 - accuracy: 0.2181 - val_loss: 748522.1258 - val_accuracy: 0.1429
Epoch 4/10: ..... - 14s 1s/step - loss: 3.1899 - accuracy: 0.1889 - val_loss: 136493.8584 - val_accuracy: 0.1429
Epoch 5/10: ..... - 14s 1s/step - loss: 2.9712 - accuracy: 0.1755 - val_loss: 253621.9862 - val_accuracy: 0.2857
Epoch 6/10: ..... - 14s 1s/step - loss: 2.9888 - accuracy: 0.1622 - val_loss: 17886.1543 - val_accuracy: 0.1429
Epoch 7/10: ..... - 14s 1s/step - loss: 2.8311 - accuracy: 0.1489 - val_loss: 826.5386 - val_accuracy: 0.0000e+00
Epoch 8/10: ..... - 14s 1s/step - loss: 2.4029 - accuracy: 0.1841 - val_loss: 18529.5938 - val_accuracy: 0.1429
Epoch 9/10: ..... - 14s 1s/step - loss: 2.1347 - accuracy: 0.1888 - val_loss: 31282.3852 - val_accuracy: 0.1429
Epoch 10/10: ..... - 14s 1s/step - loss: 2.1408 - accuracy: 0.2207 - val_loss: 69.5367 - val_accuracy: 0.2857
keras.callbacks.History at 8c7ed6d0a0c0b

```

Model 4

```

block11_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_18[0][0]']
block11_sepconv1_bn (Batch Normalization) (None, 16, 16, 728) 536536 ['block11_sepconv1_act[0][0]']
conv2d_97 (Conv2D) (None, 16, 16, 728) 2912 ['block11_sepconv1_bn[0][0]']
block11_sepconv2_act (Activation) (None, 16, 16, 728) 0 ['block11_sepconv1_bn[0][0]']
block11_sepconv2_bn (Batch Normalization) (None, 16, 16, 1024) 752824 ['block11_sepconv2_act[0][0]']
conv2d_98 (Conv2D) (None, 16, 16, 1024) 4896 ['block11_sepconv2_bn[0][0]']
conv2d_99 (Conv2D) (None, 16, 16, 1024) 745472 ['add_18[0][0]']
block11_pool (MaxPooling2D) (None, 16, 16, 1024) 0 ['block11_sepconv2_bn[0][0]']
batch_normalization_97 (Batch Normalization) (None, 16, 16, 1024) 4896 ['conv2d_99[0][0]']
add_11 (Add) (None, 16, 16, 1024) 0 ['block11_pool[0][0]', 'batch_normalization_97[0][0]']
block14_sepconv1 (Separable Conv2D) (None, 16, 16, 1536) 1562080 ['add_11[0][0]']
block14_sepconv1_bn (Batch Normalization) (None, 16, 16, 1536) 6144 ['block14_sepconv1[0][0]']
block14_sepconv1_act (Activation) (None, 16, 16, 1536) 0 ['block14_sepconv1_bn[0][0]']
block14_sepconv2 (Separable Conv2D) (None, 16, 16, 2048) 3159552 ['block14_sepconv1_act[0][0]']
block14_sepconv2_bn (Batch Normalization) (None, 16, 16, 2048) 8192 ['block14_sepconv2[0][0]']
block14_sepconv2_act (Activation) (None, 16, 16, 2048) 0 ['block14_sepconv2_bn[0][0]']
flatten_3 (Flatten) (None, 268480) 0 ['block14_sepconv2_act[0][0]']
dense_3 (Dense) (None, 8) 1638408 ['flatten_3[0][0]']

Total params: 22859888 (86.83 MB)
Trainable params: 1638408 (6.25 MB)
Non-trainable params: 20861480 (79.58 MB)

```

```

inception.fit(train_generator, validation_data = test_generator, epochs=10 )

Epoch 1/10: ..... - 29s 2s/step - loss: 1.8818 - accuracy: 0.8431 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 2/10: ..... - 13s 1s/step - loss: 0.2874 - accuracy: 0.9814 - val_loss: 0.8364 - val_accuracy: 1.0000
Epoch 3/10: ..... - 13s 1s/step - loss: 0.1984 - accuracy: 0.9814 - val_loss: 2.8977 - val_accuracy: 0.8571
Epoch 4/10: ..... - 13s 1s/step - loss: 0.2394 - accuracy: 0.9814 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 5/10: ..... - 13s 1s/step - loss: 0.1316 - accuracy: 0.9947 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 6/10: ..... - 13s 1s/step - loss: 0.8178 - accuracy: 0.9947 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 7/10: ..... - 12s 1s/step - loss: 0.8562 - accuracy: 0.9928 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 8/10: ..... - 13s 1s/step - loss: 0.8845 - accuracy: 0.9973 - val_loss: 8.549e-08 - val_accuracy: 1.0000
Epoch 9/10: ..... - 12s 1s/step - loss: 0.8821 - accuracy: 0.9973 - val_loss: 1.762e-08 - val_accuracy: 1.0000
Epoch 10/10: ..... - 13s 1s/step - loss: 0.8814 - accuracy: 1.0000 - val_loss: 3.469e-08 - val_accuracy: 1.0000
keras.callbacks.History at 8c7ed6d0a0c0b

```