\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

input\_6 (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\_2 (Flatten) (None, 25088) 0

dense\_2 (Dense) (None, 1) 25089

=================================================================

Total params: 14,739,777

Trainable params: 25,089

Non-trainable params: 14,714,688

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Epoch 1/100

55/55 [==============================] - 55s 973ms/step - loss: 0.3908 - accuracy: 0.8456 - val\_loss: 0.1855 - val\_accuracy: 0.9171 - lr: 0.0010

Epoch 2/100

55/55 [==============================] - 51s 924ms/step - loss: 0.1565 - accuracy: 0.9430 - val\_loss: 0.2091 - val\_accuracy: 0.9171 - lr: 0.0010

Epoch 3/100

55/55 [==============================] - 51s 922ms/step - loss: 0.0987 - accuracy: 0.9637 - val\_loss: 0.1107 - val\_accuracy: 0.9470 - lr: 0.0010

Epoch 4/100

55/55 [==============================] - 51s 931ms/step - loss: 0.0752 - accuracy: 0.9724 - val\_loss: 0.0784 - val\_accuracy: 0.9700 - lr: 0.0010

Epoch 5/100

55/55 [==============================] - 51s 920ms/step - loss: 0.0611 - accuracy: 0.9810 - val\_loss: 0.0655 - val\_accuracy: 0.9700 - lr: 0.0010

Epoch 6/100

55/55 [==============================] - 51s 919ms/step - loss: 0.0479 - accuracy: 0.9850 - val\_loss: 0.0660 - val\_accuracy: 0.9724 - lr: 0.0010

Epoch 7/100

55/55 [==============================] - 50s 913ms/step - loss: 0.0390 - accuracy: 0.9856 - val\_loss: 0.0628 - val\_accuracy: 0.9747 - lr: 0.0010

Epoch 8/100

55/55 [==============================] - 50s 912ms/step - loss: 0.0276 - accuracy: 0.9914 - val\_loss: 0.0645 - val\_accuracy: 0.9700 - lr: 0.0010

Epoch 9/100

55/55 [==============================] - 51s 917ms/step - loss: 0.0303 - accuracy: 0.9891 - val\_loss: 0.0554 - val\_accuracy: 0.9724 - lr: 0.0010

Epoch 10/100

55/55 [==============================] - 50s 912ms/step - loss: 0.0265 - accuracy: 0.9919 - val\_loss: 0.0563 - val\_accuracy: 0.9724 - lr: 0.0010

Epoch 11/100

55/55 [==============================] - 49s 898ms/step - loss: 0.0150 - accuracy: 0.9983 - val\_loss: 0.0778 - val\_accuracy: 0.9724 - lr: 0.0010

Epoch 12/100

55/55 [==============================] - 48s 877ms/step - loss: 0.0170 - accuracy: 0.9942 - val\_loss: 0.0901 - val\_accuracy: 0.9631 - lr: 0.0010

Epoch 13/100

55/55 [==============================] - 49s 894ms/step - loss: 0.0065 - accuracy: 0.9994 - val\_loss: 0.0560 - val\_accuracy: 0.9724 - lr: 1.0000e-04

Epoch 14/100

55/55 [==============================] - 49s 885ms/step - loss: 0.0055 - accuracy: 1.0000 - val\_loss: 0.0558 - val\_accuracy: 0.9770 - lr: 1.0000e-04

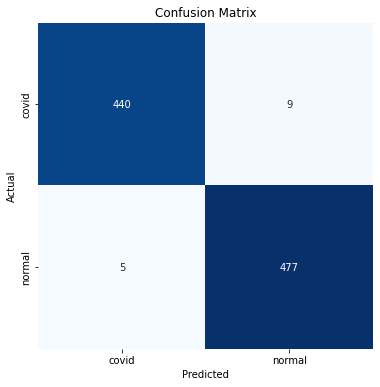
INFO:tensorflow:Assets written to: /content/drive/MyDrive/RMSpropVGG16Split0.7noAug/assets

Test Loss: 0.05893

Test Accuracy: 98.50%

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:123: DeprecationWarning: `np.int` is a deprecated alias for the builtin `int`. To silence this warning, use `int` by itself. Doing this will not modify any behavior and is safe. When replacing `np.int`, you may wish to use e.g. `np.int64` or `np.int32` to specify the precision. If you wish to review your current use, check the release note link for additional information.

Deprecated in NumPy 1.20; for more details and guidance: <https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>



Classification Report:

----------------------

precision recall f1-score support

covid 0.99 0.98 0.98 449

normal 0.98 0.99 0.99 482

accuracy 0.98 931

macro avg 0.99 0.98 0.98 931

weighted avg 0.98 0.98 0.98 931

