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

Layer (type) Output Shape Param #

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

input\_4 (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\_1 (Flatten) (None, 25088) 0

dense\_1 (Dense) (None, 1) 25089

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

Total params: 14,739,777

Trainable params: 25,089

Non-trainable params: 14,714,688

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

Epoch 1/100

55/55 [==============================] - 52s 918ms/step - loss: 5.9315 - accuracy: 0.5403 - val\_loss: 1.1180 - val\_accuracy: 0.7465 - lr: 0.0100

Epoch 2/100

55/55 [==============================] - 50s 909ms/step - loss: 1.2461 - accuracy: 0.8065 - val\_loss: 0.5218 - val\_accuracy: 0.8618 - lr: 0.0100

Epoch 3/100

55/55 [==============================] - 51s 918ms/step - loss: 0.7900 - accuracy: 0.8520 - val\_loss: 0.6267 - val\_accuracy: 0.8318 - lr: 0.0100

Epoch 4/100

55/55 [==============================] - 49s 901ms/step - loss: 0.3236 - accuracy: 0.9119 - val\_loss: 0.2154 - val\_accuracy: 0.9309 - lr: 0.0100

Epoch 5/100

55/55 [==============================] - 49s 897ms/step - loss: 0.5018 - accuracy: 0.8836 - val\_loss: 0.6839 - val\_accuracy: 0.8203 - lr: 0.0100

Epoch 6/100

55/55 [==============================] - 50s 898ms/step - loss: 1.0068 - accuracy: 0.8525 - val\_loss: 0.1726 - val\_accuracy: 0.9355 - lr: 0.0100

Epoch 7/100

55/55 [==============================] - 56s 1s/step - loss: 0.2600 - accuracy: 0.9257 - val\_loss: 0.1475 - val\_accuracy: 0.9447 - lr: 0.0100

Epoch 8/100

55/55 [==============================] - 50s 900ms/step - loss: 0.1406 - accuracy: 0.9522 - val\_loss: 0.9109 - val\_accuracy: 0.7742 - lr: 0.0100

Epoch 9/100

55/55 [==============================] - 56s 1s/step - loss: 0.3751 - accuracy: 0.9096 - val\_loss: 0.1766 - val\_accuracy: 0.9355 - lr: 0.0100

Epoch 10/100

55/55 [==============================] - 50s 901ms/step - loss: 0.1195 - accuracy: 0.9620 - val\_loss: 0.1029 - val\_accuracy: 0.9516 - lr: 0.0100

Epoch 11/100

55/55 [==============================] - 50s 909ms/step - loss: 0.1054 - accuracy: 0.9666 - val\_loss: 0.0893 - val\_accuracy: 0.9608 - lr: 0.0100

Epoch 12/100

55/55 [==============================] - 51s 947ms/step - loss: 0.1015 - accuracy: 0.9620 - val\_loss: 0.1027 - val\_accuracy: 0.9654 - lr: 0.0100

Epoch 13/100

55/55 [==============================] - 50s 917ms/step - loss: 0.0736 - accuracy: 0.9729 - val\_loss: 0.0836 - val\_accuracy: 0.9585 - lr: 0.0100

Epoch 14/100

55/55 [==============================] - 50s 912ms/step - loss: 0.0601 - accuracy: 0.9770 - val\_loss: 0.0800 - val\_accuracy: 0.9608 - lr: 0.0100

Epoch 15/100

55/55 [==============================] - 51s 923ms/step - loss: 0.0708 - accuracy: 0.9752 - val\_loss: 0.0882 - val\_accuracy: 0.9654 - lr: 0.0100

Epoch 16/100

55/55 [==============================] - 51s 917ms/step - loss: 0.0593 - accuracy: 0.9810 - val\_loss: 0.0788 - val\_accuracy: 0.9631 - lr: 0.0100

Epoch 17/100

55/55 [==============================] - 50s 917ms/step - loss: 0.0623 - accuracy: 0.9741 - val\_loss: 0.0755 - val\_accuracy: 0.9631 - lr: 0.0100

Epoch 18/100

55/55 [==============================] - 51s 920ms/step - loss: 0.0532 - accuracy: 0.9793 - val\_loss: 0.0752 - val\_accuracy: 0.9608 - lr: 0.0100

Epoch 19/100

55/55 [==============================] - 50s 917ms/step - loss: 0.0467 - accuracy: 0.9856 - val\_loss: 0.0722 - val\_accuracy: 0.9700 - lr: 0.0100

Epoch 20/100

55/55 [==============================] - 50s 917ms/step - loss: 0.0561 - accuracy: 0.9770 - val\_loss: 0.2488 - val\_accuracy: 0.9124 - lr: 0.0100

Epoch 21/100

55/55 [==============================] - 50s 914ms/step - loss: 0.0526 - accuracy: 0.9827 - val\_loss: 0.0723 - val\_accuracy: 0.9608 - lr: 0.0100

Epoch 22/100

55/55 [==============================] - 50s 916ms/step - loss: 0.0464 - accuracy: 0.9798 - val\_loss: 0.0725 - val\_accuracy: 0.9608 - lr: 0.0100

Epoch 23/100

55/55 [==============================] - 50s 915ms/step - loss: 0.0343 - accuracy: 0.9891 - val\_loss: 0.0730 - val\_accuracy: 0.9608 - lr: 1.0000e-03

Epoch 24/100

55/55 [==============================] - 50s 913ms/step - loss: 0.0334 - accuracy: 0.9891 - val\_loss: 0.0757 - val\_accuracy: 0.9585 - lr: 1.0000e-03

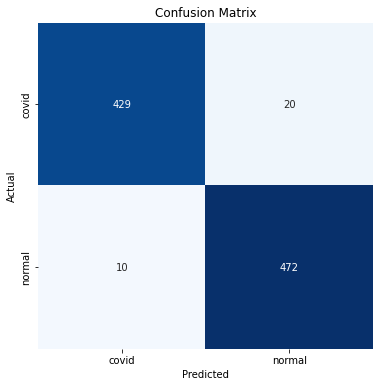
INFO:tensorflow:Assets written to: /content/drive/MyDrive/SGDVGG16Split0.7noAug/assets

Test Loss: 0.09246

Test Accuracy: 96.78%

/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 0.98 0.96 0.97 449

normal 0.96 0.98 0.97 482

accuracy 0.97 931

macro avg 0.97 0.97 0.97 931

weighted avg 0.97 0.97 0.97 931

