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

Layer (type) Output Shape Param #

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

input\_3 (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, 1) 25089

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

Total params: 14,739,777

Trainable params: 25,089

Non-trainable params: 14,714,688

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

Epoch 1/100

17/17 [==============================] - 160s 9s/step - loss: 0.7671 - accuracy: 0.6089 - val\_loss: 0.5063 - val\_accuracy: 0.7537 - lr: 0.0010

Epoch 2/100

17/17 [==============================] - 7s 378ms/step - loss: 0.4256 - accuracy: 0.7933 - val\_loss: 0.4895 - val\_accuracy: 0.7836 - lr: 0.0010

Epoch 3/100

17/17 [==============================] - 7s 383ms/step - loss: 0.3180 - accuracy: 0.8790 - val\_loss: 0.3542 - val\_accuracy: 0.8507 - lr: 0.0010

Epoch 4/100

17/17 [==============================] - 7s 382ms/step - loss: 0.2300 - accuracy: 0.9330 - val\_loss: 0.3383 - val\_accuracy: 0.8657 - lr: 0.0010

Epoch 5/100

17/17 [==============================] - 6s 377ms/step - loss: 0.1928 - accuracy: 0.9572 - val\_loss: 0.3224 - val\_accuracy: 0.8507 - lr: 0.0010

Epoch 6/100

17/17 [==============================] - 6s 372ms/step - loss: 0.1598 - accuracy: 0.9628 - val\_loss: 0.3268 - val\_accuracy: 0.8358 - lr: 0.0010

Epoch 7/100

17/17 [==============================] - 7s 379ms/step - loss: 0.1452 - accuracy: 0.9721 - val\_loss: 0.3135 - val\_accuracy: 0.8806 - lr: 0.0010

Epoch 8/100

17/17 [==============================] - 7s 380ms/step - loss: 0.1120 - accuracy: 0.9888 - val\_loss: 0.2967 - val\_accuracy: 0.8582 - lr: 0.0010

Epoch 9/100

17/17 [==============================] - 7s 376ms/step - loss: 0.0981 - accuracy: 0.9926 - val\_loss: 0.2938 - val\_accuracy: 0.8507 - lr: 0.0010

Epoch 10/100

17/17 [==============================] - 7s 379ms/step - loss: 0.0877 - accuracy: 0.9944 - val\_loss: 0.2996 - val\_accuracy: 0.8881 - lr: 0.0010

Epoch 11/100

17/17 [==============================] - 7s 379ms/step - loss: 0.0872 - accuracy: 0.9963 - val\_loss: 0.3136 - val\_accuracy: 0.8358 - lr: 0.0010

Epoch 12/100

17/17 [==============================] - 7s 382ms/step - loss: 0.0699 - accuracy: 1.0000 - val\_loss: 0.2908 - val\_accuracy: 0.8731 - lr: 0.0010

Epoch 13/100

17/17 [==============================] - 7s 379ms/step - loss: 0.0648 - accuracy: 0.9981 - val\_loss: 0.2788 - val\_accuracy: 0.8731 - lr: 0.0010

Epoch 14/100

17/17 [==============================] - 7s 379ms/step - loss: 0.0545 - accuracy: 1.0000 - val\_loss: 0.2772 - val\_accuracy: 0.8806 - lr: 0.0010

Epoch 15/100

17/17 [==============================] - 7s 382ms/step - loss: 0.0474 - accuracy: 1.0000 - val\_loss: 0.2756 - val\_accuracy: 0.8806 - lr: 0.0010

Epoch 16/100

17/17 [==============================] - 7s 380ms/step - loss: 0.0444 - accuracy: 1.0000 - val\_loss: 0.2733 - val\_accuracy: 0.8881 - lr: 0.0010

Epoch 17/100

17/17 [==============================] - 7s 380ms/step - loss: 0.0417 - accuracy: 1.0000 - val\_loss: 0.2747 - val\_accuracy: 0.8657 - lr: 0.0010

Epoch 18/100

17/17 [==============================] - 7s 386ms/step - loss: 0.0389 - accuracy: 1.0000 - val\_loss: 0.2736 - val\_accuracy: 0.8881 - lr: 0.0010

Epoch 19/100

17/17 [==============================] - 7s 381ms/step - loss: 0.0370 - accuracy: 1.0000 - val\_loss: 0.2809 - val\_accuracy: 0.9030 - lr: 0.0010

Epoch 20/100

17/17 [==============================] - 7s 386ms/step - loss: 0.0324 - accuracy: 1.0000 - val\_loss: 0.2683 - val\_accuracy: 0.8806 - lr: 1.0000e-04

Epoch 21/100

17/17 [==============================] - 7s 380ms/step - loss: 0.0318 - accuracy: 1.0000 - val\_loss: 0.2686 - val\_accuracy: 0.8806 - lr: 1.0000e-04

Epoch 22/100

17/17 [==============================] - 7s 382ms/step - loss: 0.0313 - accuracy: 1.0000 - val\_loss: 0.2697 - val\_accuracy: 0.8806 - lr: 1.0000e-04

Epoch 23/100

17/17 [==============================] - 6s 372ms/step - loss: 0.0312 - accuracy: 1.0000 - val\_loss: 0.2700 - val\_accuracy: 0.8806 - lr: 1.0000e-04

Epoch 24/100

17/17 [==============================] - 7s 381ms/step - loss: 0.0309 - accuracy: 1.0000 - val\_loss: 0.2696 - val\_accuracy: 0.8806 - lr: 1.0000e-05

Epoch 25/100

17/17 [==============================] - 7s 379ms/step - loss: 0.0308 - accuracy: 1.0000 - val\_loss: 0.2695 - val\_accuracy: 0.8806 - lr: 1.0000e-05

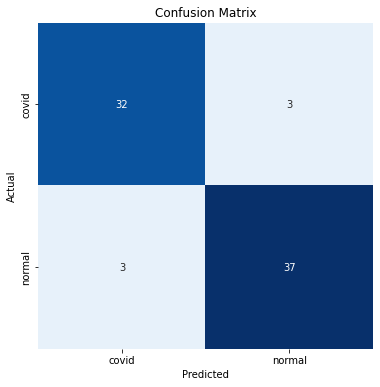
INFO:tensorflow:Assets written to: /content/drive/MyDrive/CTvggSplit0.9noAug/assets

Test Loss: 0.23102

Test Accuracy: 92.00%

/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 0.91 0.91 0.91 35

normal 0.93 0.93 0.93 40

accuracy 0.92 75

macro avg 0.92 0.92 0.92 75

weighted avg 0.92 0.92 0.92 75

