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

Layer (type) Output Shape Param #

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

input\_12 (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\_5 (Flatten) (None, 25088) 0

dense\_5 (Dense) (None, 1) 25089

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

Total params: 14,739,777

Trainable params: 25,089

Non-trainable params: 14,714,688

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

Epoch 1/100

138/138 [==============================] - 58s 414ms/step - loss: 0.2446 - accuracy: 0.8890 - val\_loss: 0.1259 - val\_accuracy: 0.9563 - lr: 0.0010

Epoch 2/100

138/138 [==============================] - 52s 373ms/step - loss: 0.1117 - accuracy: 0.9597 - val\_loss: 0.0979 - val\_accuracy: 0.9691 - lr: 0.0010

Epoch 3/100

138/138 [==============================] - 51s 370ms/step - loss: 0.0741 - accuracy: 0.9777 - val\_loss: 0.1073 - val\_accuracy: 0.9572 - lr: 0.0010

Epoch 4/100

138/138 [==============================] - 51s 369ms/step - loss: 0.0535 - accuracy: 0.9861 - val\_loss: 0.0684 - val\_accuracy: 0.9773 - lr: 0.0010

Epoch 5/100

138/138 [==============================] - 51s 368ms/step - loss: 0.0369 - accuracy: 0.9936 - val\_loss: 0.0669 - val\_accuracy: 0.9736 - lr: 0.0010

Epoch 6/100

138/138 [==============================] - 51s 369ms/step - loss: 0.0289 - accuracy: 0.9966 - val\_loss: 0.0645 - val\_accuracy: 0.9745 - lr: 0.0010

Epoch 7/100

138/138 [==============================] - 51s 367ms/step - loss: 0.0235 - accuracy: 0.9980 - val\_loss: 0.0604 - val\_accuracy: 0.9773 - lr: 0.0010

Epoch 8/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0199 - accuracy: 0.9993 - val\_loss: 0.0545 - val\_accuracy: 0.9791 - lr: 0.0010

Epoch 9/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0170 - accuracy: 0.9998 - val\_loss: 0.0589 - val\_accuracy: 0.9754 - lr: 0.0010

Epoch 10/100

138/138 [==============================] - 51s 367ms/step - loss: 0.0154 - accuracy: 0.9991 - val\_loss: 0.0492 - val\_accuracy: 0.9827 - lr: 0.0010

Epoch 11/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0109 - accuracy: 1.0000 - val\_loss: 0.0490 - val\_accuracy: 0.9836 - lr: 0.0010

Epoch 12/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0091 - accuracy: 1.0000 - val\_loss: 0.0485 - val\_accuracy: 0.9836 - lr: 0.0010

Epoch 13/100

138/138 [==============================] - 51s 367ms/step - loss: 0.0080 - accuracy: 1.0000 - val\_loss: 0.0459 - val\_accuracy: 0.9845 - lr: 0.0010

Epoch 14/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0071 - accuracy: 1.0000 - val\_loss: 0.0501 - val\_accuracy: 0.9836 - lr: 0.0010

Epoch 15/100

138/138 [==============================] - 51s 365ms/step - loss: 0.0060 - accuracy: 1.0000 - val\_loss: 0.0474 - val\_accuracy: 0.9836 - lr: 0.0010

Epoch 16/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0060 - accuracy: 1.0000 - val\_loss: 0.0434 - val\_accuracy: 0.9854 - lr: 0.0010

Epoch 17/100

138/138 [==============================] - 50s 365ms/step - loss: 0.0051 - accuracy: 1.0000 - val\_loss: 0.0423 - val\_accuracy: 0.9845 - lr: 0.0010

Epoch 18/100

138/138 [==============================] - 50s 365ms/step - loss: 0.0043 - accuracy: 1.0000 - val\_loss: 0.0441 - val\_accuracy: 0.9836 - lr: 0.0010

Epoch 19/100

138/138 [==============================] - 50s 364ms/step - loss: 0.0040 - accuracy: 1.0000 - val\_loss: 0.0425 - val\_accuracy: 0.9864 - lr: 0.0010

Epoch 20/100

138/138 [==============================] - 51s 365ms/step - loss: 0.0035 - accuracy: 1.0000 - val\_loss: 0.0441 - val\_accuracy: 0.9845 - lr: 0.0010

Epoch 21/100

138/138 [==============================] - 50s 363ms/step - loss: 0.0030 - accuracy: 1.0000 - val\_loss: 0.0422 - val\_accuracy: 0.9864 - lr: 1.0000e-04

Epoch 22/100

138/138 [==============================] - 51s 365ms/step - loss: 0.0030 - accuracy: 1.0000 - val\_loss: 0.0428 - val\_accuracy: 0.9845 - lr: 1.0000e-04

Epoch 23/100

138/138 [==============================] - 51s 365ms/step - loss: 0.0029 - accuracy: 1.0000 - val\_loss: 0.0426 - val\_accuracy: 0.9845 - lr: 1.0000e-04

Epoch 24/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0029 - accuracy: 1.0000 - val\_loss: 0.0427 - val\_accuracy: 0.9845 - lr: 1.0000e-04

Epoch 25/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0028 - accuracy: 1.0000 - val\_loss: 0.0427 - val\_accuracy: 0.9845 - lr: 1.0000e-05

Epoch 26/100

138/138 [==============================] - 51s 366ms/step - loss: 0.0028 - accuracy: 1.0000 - val\_loss: 0.0426 - val\_accuracy: 0.9854 - lr: 1.0000e-05

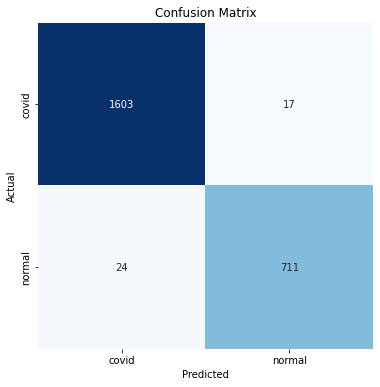
INFO:tensorflow:Assets written to: /content/drive/MyDrive/PNGMendelyCTVGG16Split0.7noAug9000/assets

Test Loss: 0.05780

Test Accuracy: 98.26%

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:127: 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

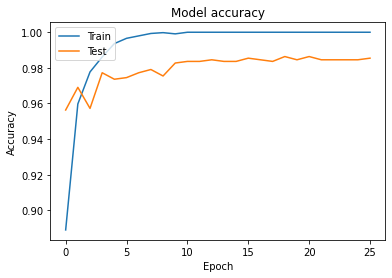
covid 0.99 0.99 0.99 1620

normal 0.98 0.97 0.97 735

accuracy 0.98 2355

macro avg 0.98 0.98 0.98 2355

weighted avg 0.98 0.98 0.98 2355



<matplotlib.legend.Legend at 0x7f978219cc90>

