Varying Model Parameters to Observe Test Results and Model Size

**1. Without Dropout and Batch Normalization layer**

**MODEL 1**

Epochs = 10

Image size = 120

Activation function = relu, sigmoid (sigmoid for binary classification – 1 output)

Layers:

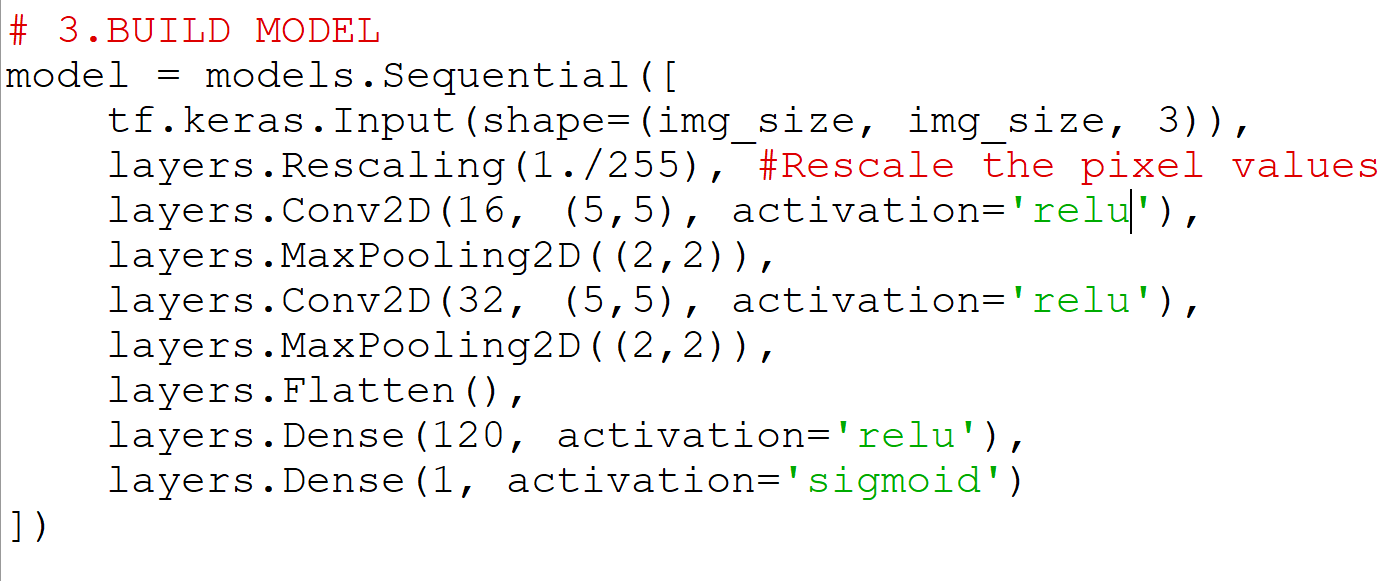
Convolution layers = 2 (Conv layer1 = 16 filters, Kernel size = 5\*5)

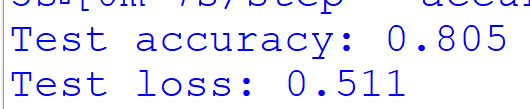
(Conv layer2 = 32 filters, Kernel size = 5\*5)

Max pooling layers = 2 (Kernel size = 2\*2)

Flattening layer (converting to 1D vector) = 1

Fully connected layers = 2





Model size = 33012 KB

**MODEL 2**

Epochs = 10

Image size = 120

Activation function = tanh, sigmoid (sigmoid for binary classification – 1 output)

Layers:

Convolution layers = 3 (Conv layer1 = 16 filters, Kernel size = 5\*5)

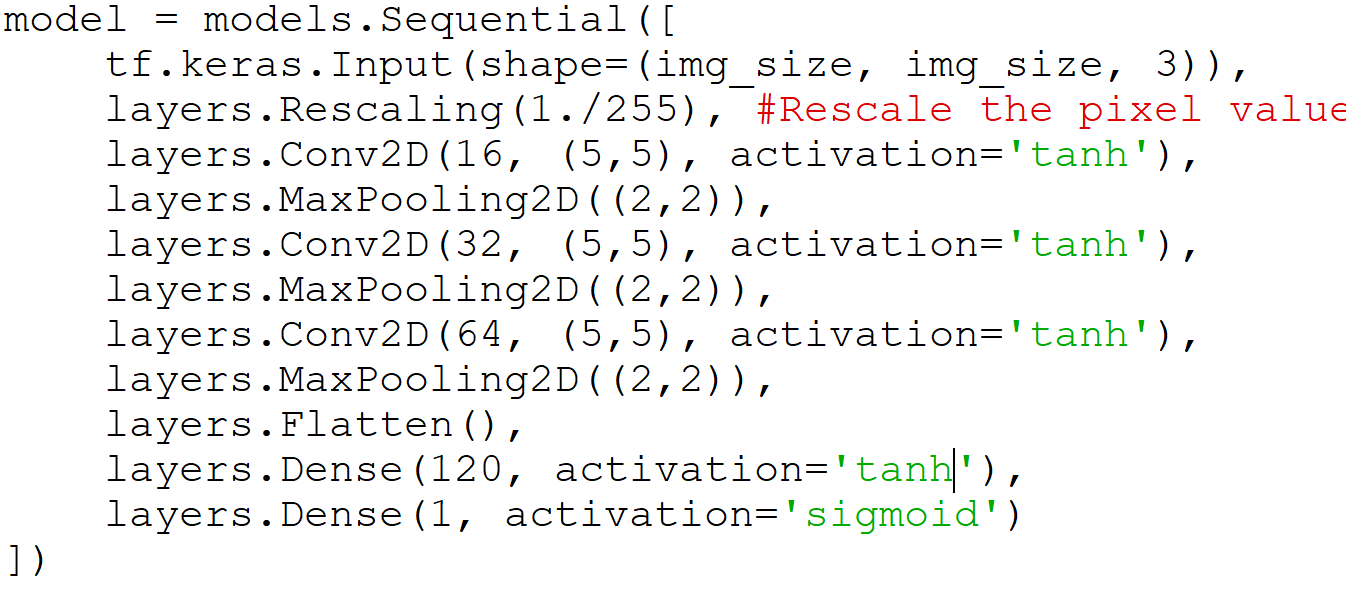
(Conv layer2 = 32 filters, Kernel size = 5\*5)

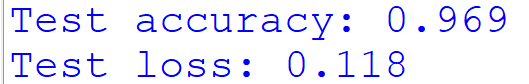
(Conv layer3 = 64 filters, Kernel size = 5\*5)

Max pooling layers = 3 (Kernel size = 2\*2)

Flattening layer (converting to 1D vector) = 1

Fully connected layers = 2





Model size = 11705 KB

**MODEL 3**

Epochs = 10

Image size = 120

Activation function = relu, tanh, sigmoid (sigmoid for binary classification – 1 output)

Layers:

Convolution layers = 3 (Conv layer1 = 12 filters, Kernel size = 3\*3)

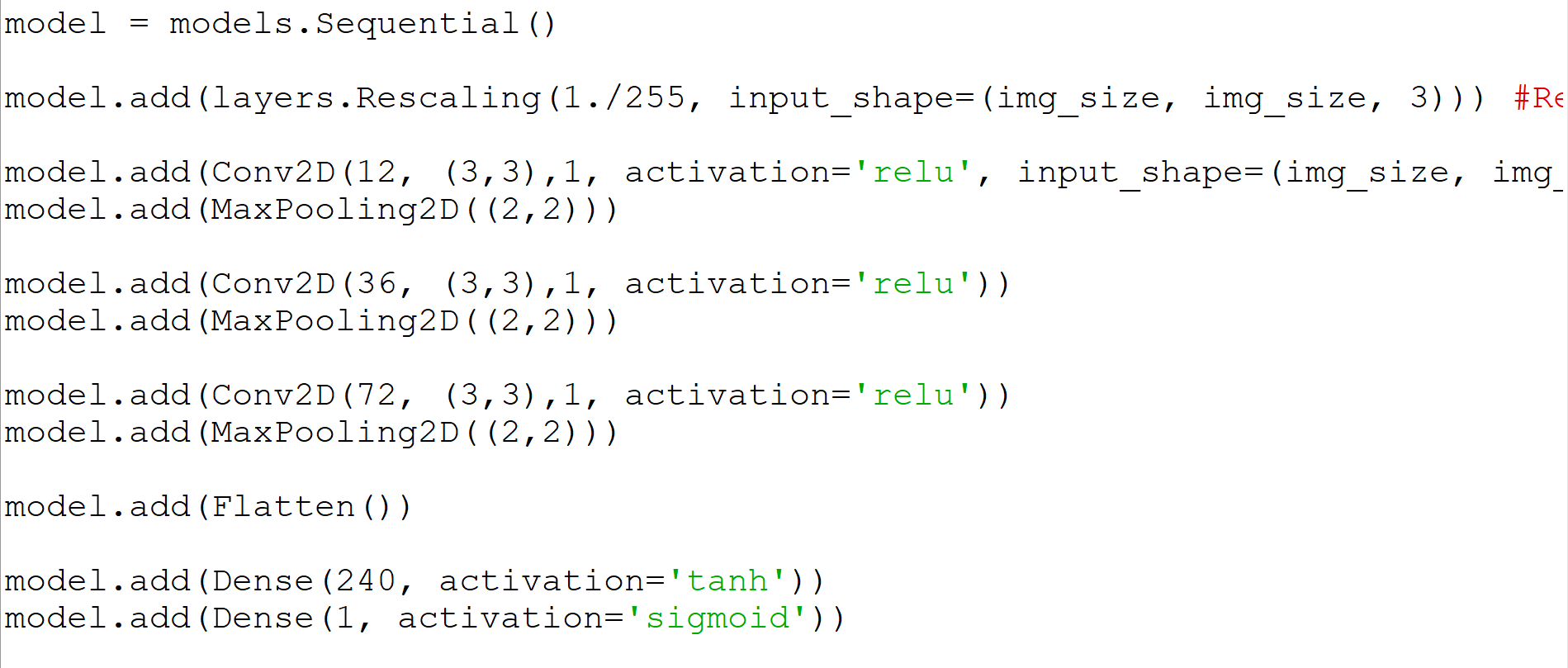
(Conv layer2 = 36 filters, Kernel size = 3\*3)

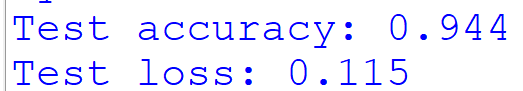
(Conv layer3 = 72 filters, Kernel size = 3\*3)

Max pooling layers = 3 (Kernel size = 2\*2)

Flattening layer (converting to 1D vector) = 1

Fully connected layers = 2





Model size = 34599 KB

**MODEL 4**

Epochs = 10

Image size = 120

Activation function = tanh, sigmoid (sigmoid for binary classification – 1 output)

Layers:

Convolution layers = 3 (Conv layer1 = 12 filters, Kernel size = 3\*3)

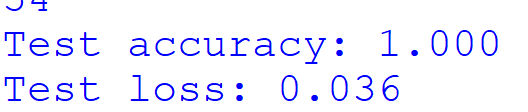
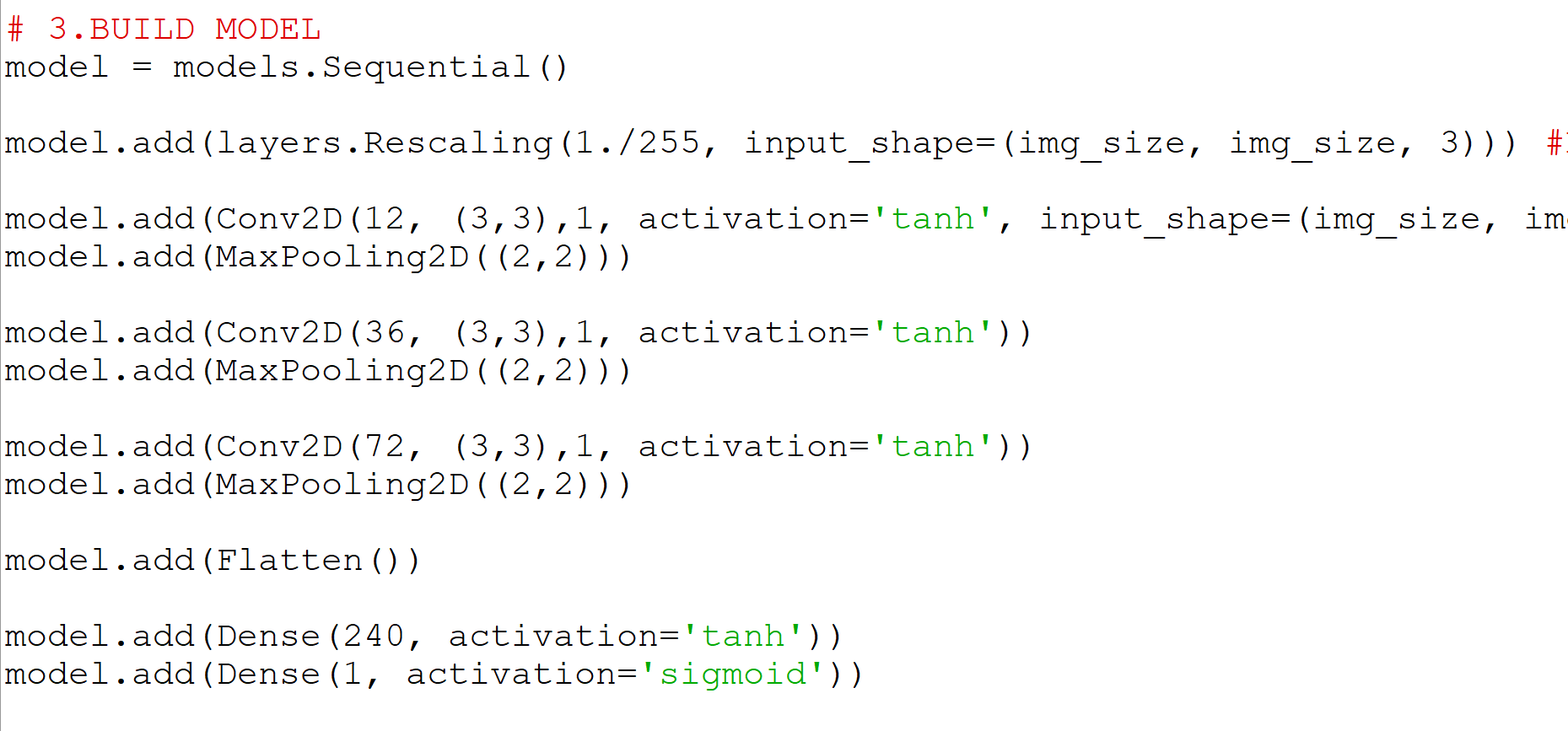
(Conv layer2 = 36 filters, Kernel size = 3\*3)

(Conv layer3 = 72 filters, Kernel size = 3\*3)

Max pooling layers = 3 (Kernel size = 2\*2)

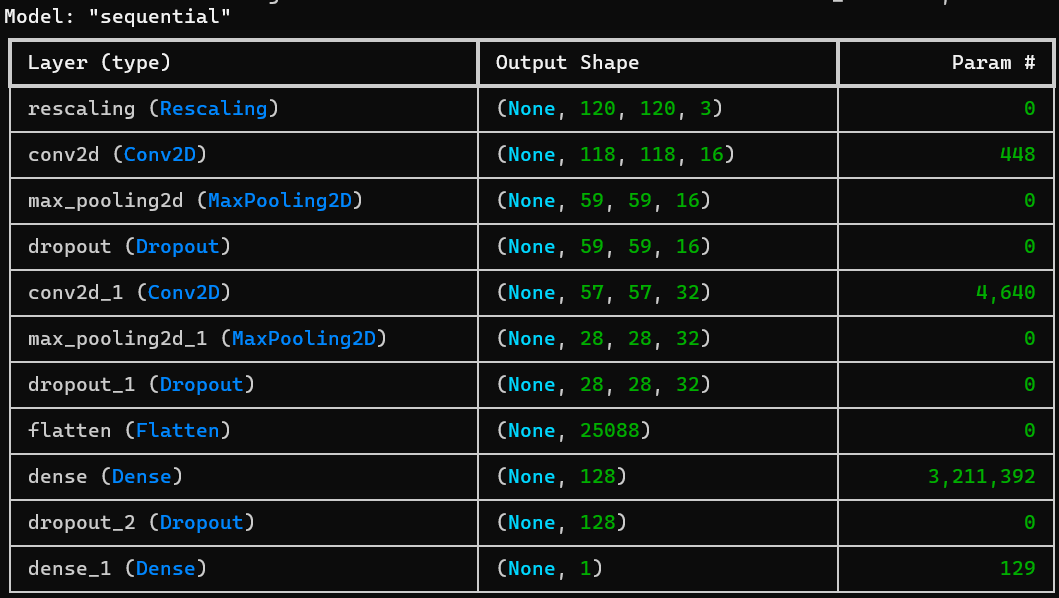
Flattening layer (converting to 1D vector) = 1

Fully connected layers = 2



Model size = 34599 KB

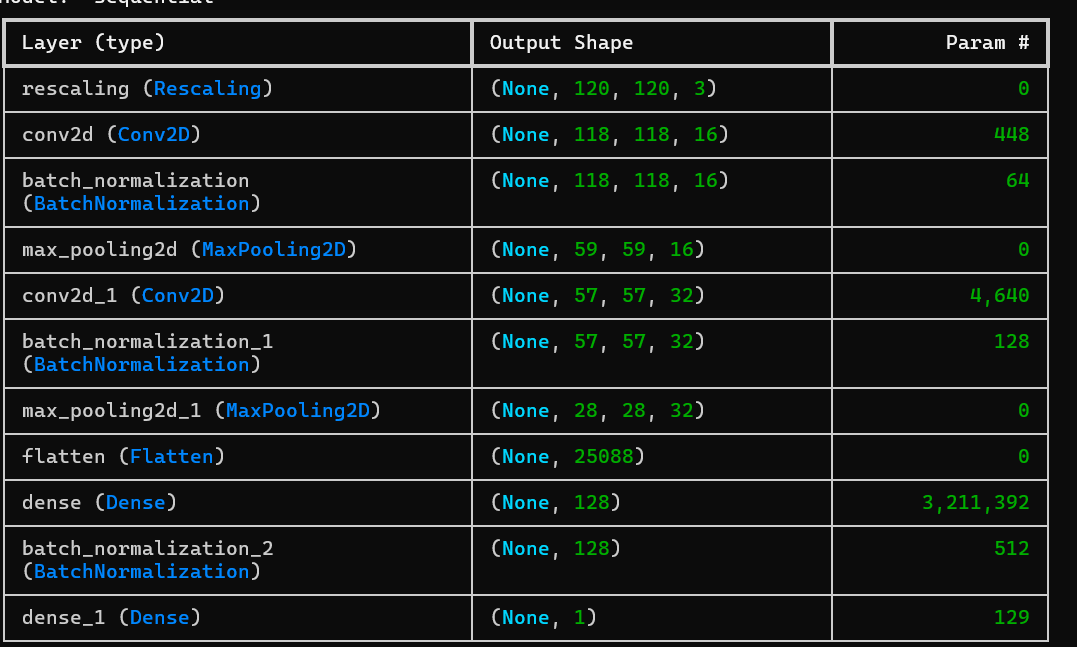
**2. With Dropout layer**





Model size= 35542 KB

**3. With BatchNormalization Layer**

****

****

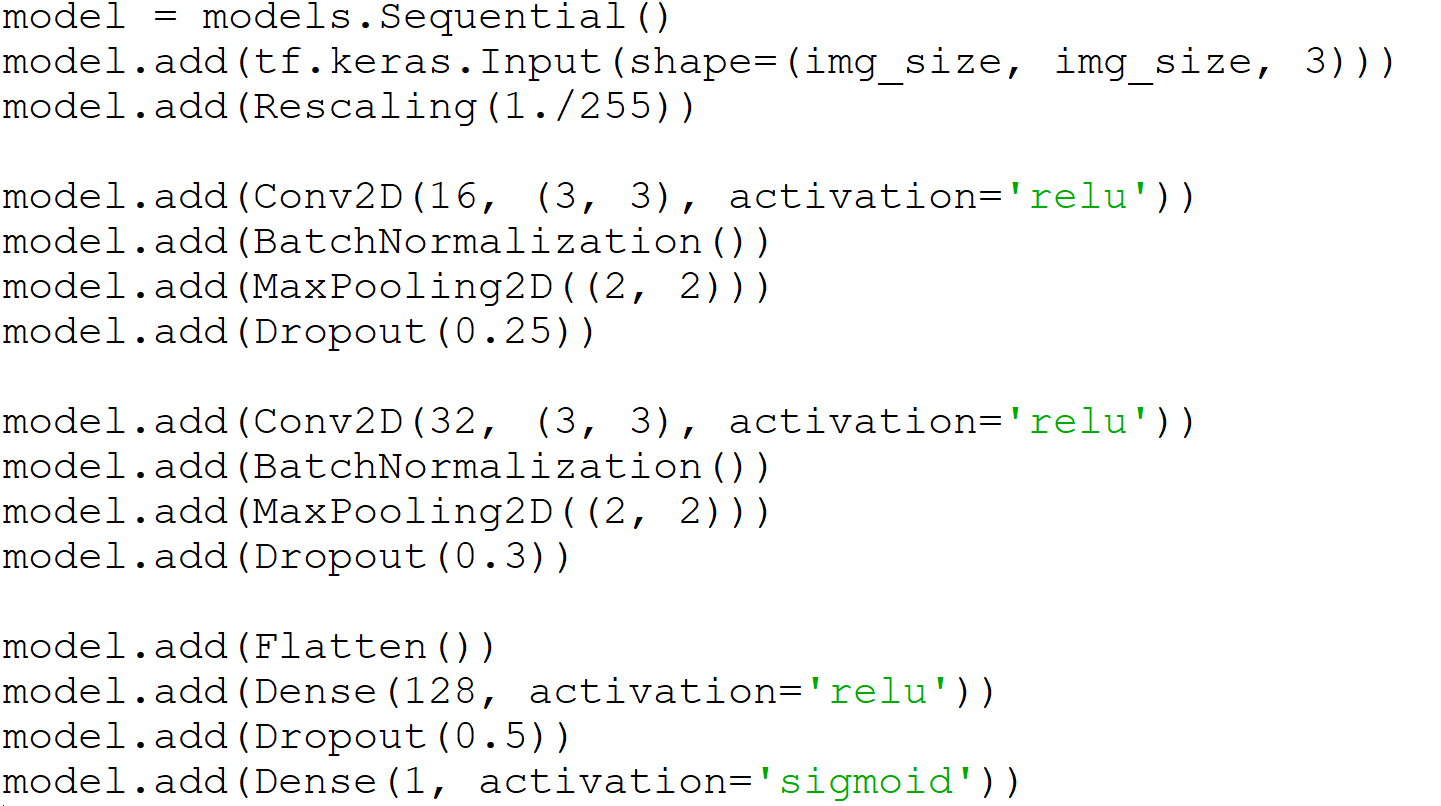
Model Size=377632 KB

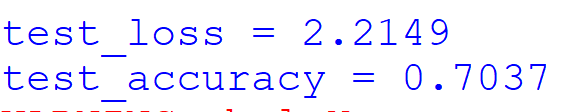
**4. Dropout + BatchNormalization layer**

(a)

Epochs=10

Image size=120

****

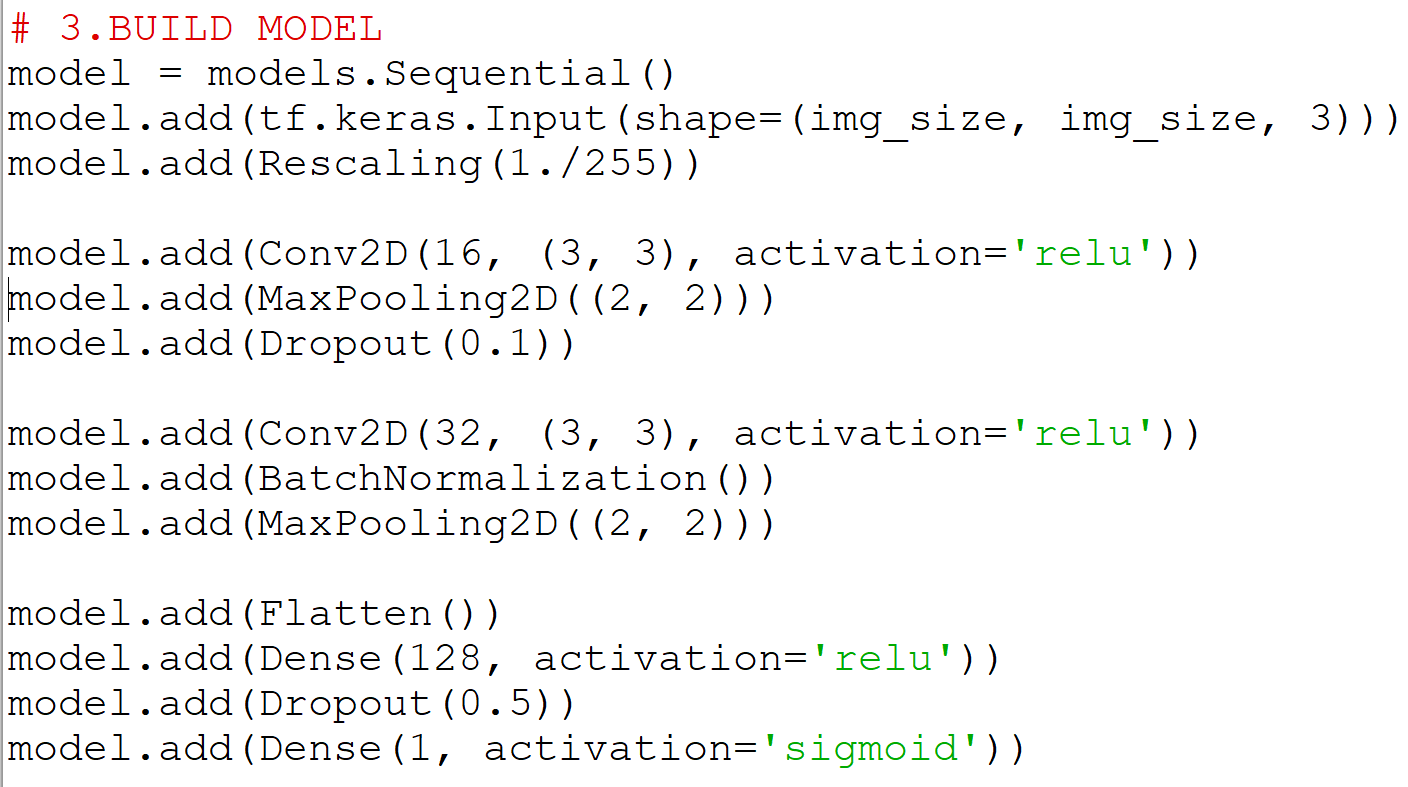
****

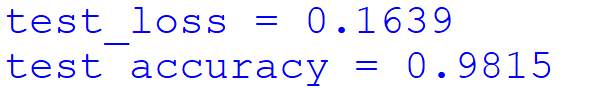
Model size= 37756 KB

(b)

Epochs=20

Image size = 120



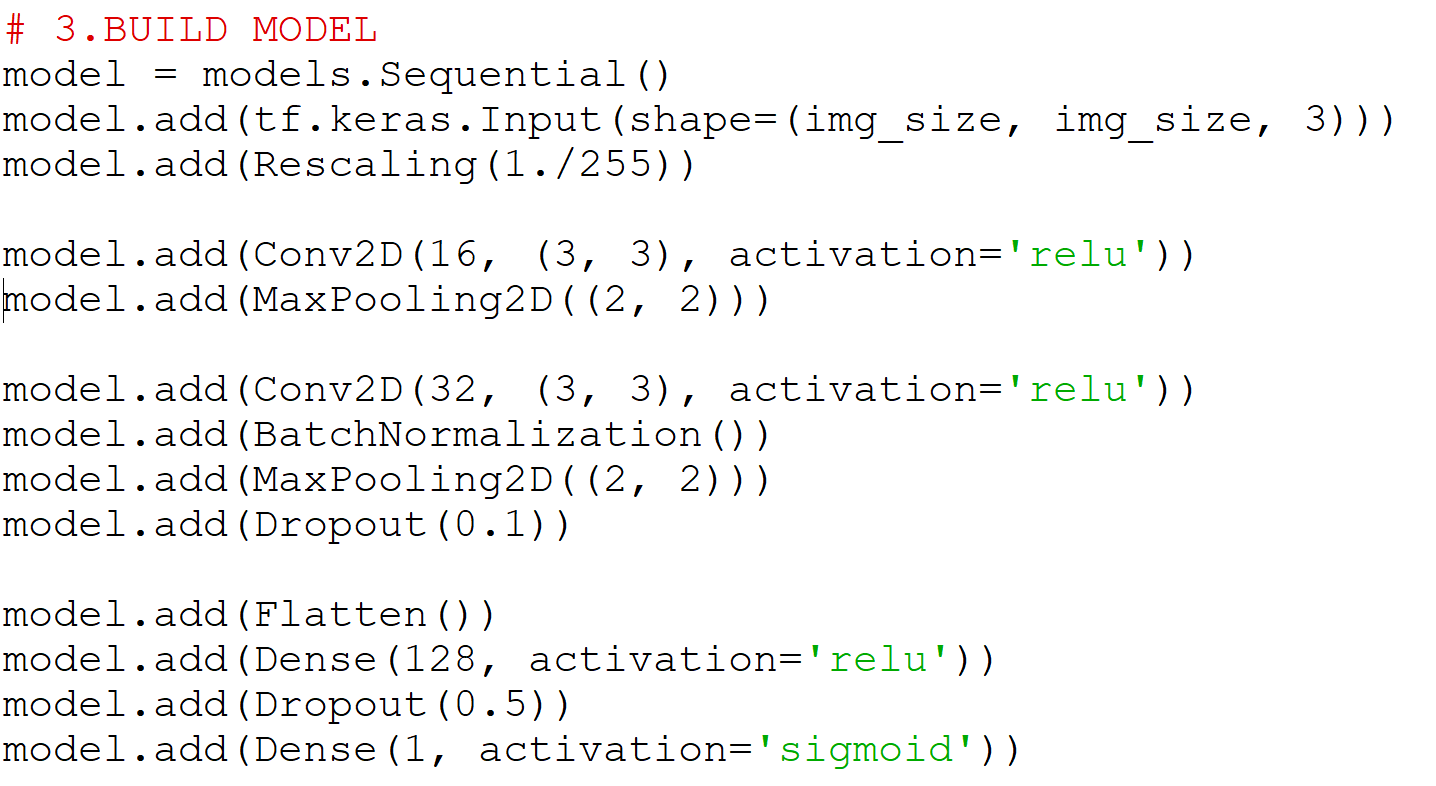


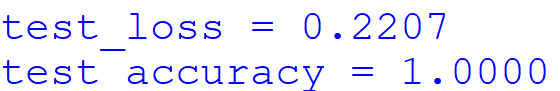
Model size=37748 KB

(c)

Epochs=20

Image size=256

****

****

Model size= 184,628 KB

**Observations**

1. **Input image size**

Larger image size:

-> Model size increases

-> Training time increase

**2**. **Number of layers**

More number of layers:

-> Model size increases

-> Accuracy increases (for large datasets and overfitting might occur for smaller datasets)

-> Training time increases

**3.** **Number of filters**

More number of filters:

-> Model size increases

-> Accuracy increases

**4. Activation function**

Relu and Sigmoid gives better accuracies in the above models

**5. Batch size**

Smaller size – more generalization

(Generalization - how well a trained model performs on unseen - test orvalidation data, not just the training data.)

**6. Epochs**

More number of epochs:

-> Accuracy increases but overfitting occurs with large number of epochs