EN2550 - Exercise 11

Name: B.S.V.W. Munasinghe

Index Number: 190397E

In [ ]:
```
#Importing Libraries
import numpy as np
import matplotlib.pyplot as plt
import cv2 as cv
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import datasets, layers, models

%matplotlib inline
```

Question 1 - LeNet5 network

In [ ]:
```
mnist = keras.datasets.mnist
(train_images, train_labels), (test_images, test_labels) = mnist.load_data()

# Padding
paddings = tf.constant([[0, 0], [2, 2], [2, 2]])
train_images = tf.pad(train_images, paddings, constant_values=0)
test_images = tf.pad(test_images, paddings, constant_values=0)

print('train_images.shape: ', train_images.shape)
print('train_labels.shape: ', train_labels.shape)
print('test_images.shape:', test_images.shape)
print('test_labels.shape:', test_labels.shape)
class_names = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

train_images = tf.dtypes.cast(train_images, tf.float32)
test_images = tf.dtypes.cast(test_images, tf.float32)
train_images, test_images = train_images[..., np.newaxis]/255.0, test_images[..., np.newaxis]/255.0
```

```
train_images.shape:  (60000, 32, 32)
train_labels.shape:  (60000,)
test_images.shape: (10000, 32, 32)
test_labels.shape: (10000,)
```

In [ ]:
```
model = models.Sequential()
model.add(layers.Conv2D(6,(5,5),activation='relu',input_shape=(32,32,1)))
model.add(layers.AveragePooling2D((2,2)))
model.add(layers.Conv2D(16,(5,5),activation='relu'))
model.add(layers.AveragePooling2D((2,2)))
model.add(layers.Flatten())
model.add(layers.Dense(120,activation='relu'))
model.add(layers.Dense(84,activation='relu'))
model.add(layers.Dense(10))

model.compile(optimizer='adam',loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),metrics=['accu
print(model.summary())

model.fit(train_images,train_labels,epochs=5)
test_loss,train_accuracy=model.evaluate(test_images,test_labels,verbose=2)
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d (Conv2D)             (None, 28, 28, 6)         156       
                                                                 
 average_pooling2d (AverageP  (None, 14, 14, 6)        0         
 ooling2D)                                                       
                                                                 
 conv2d_1 (Conv2D)           (None, 10, 10, 16)        2416      
                                                                 
 average_pooling2d_1 (Averag  (None, 5, 5, 16)         0         
 ePooling2D)                                                     
                                                                 
 flatten (Flatten)           (None, 400)               0         
                                                                 
 dense (Dense)               (None, 120)               48120     
                                                                 
 dense_1 (Dense)             (None, 84)                10164     
                                                                 
 dense_2 (Dense)             (None, 10)                850       
                                                                 
=================================================================
Total params: 61,706
Trainable params: 61,706
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/5
1875/1875 [==============================] - 16s 8ms/step - loss: 0.2242 - accuracy: 0.9316
Epoch 2/5
1875/1875 [==============================] - 17s 9ms/step - loss: 0.0676 - accuracy: 0.9788
Epoch 3/5
1875/1875 [==============================] - 18s 10ms/step - loss: 0.0488 - accuracy: 0.9845
Epoch 4/5
1875/1875 [==============================] - 16s 9ms/step - loss: 0.0375 - accuracy: 0.9880
Epoch 5/5
1875/1875 [==============================] - 17s 9ms/step - loss: 0.0307 - accuracy: 0.9902
313/313 - 1s - loss: 0.0370 - accuracy: 0.9878 - 1s/epoch - 4ms/step
```

Question 2 - CNN for CIFAR10

In [ ]:
```
from tensorflow.keras.datasets import cifar10, mnist
(train_images, train_labels), (test_images, test_labels) = datasets.cifar10.load_data()

# Normalize pixel values to be between 0 and 1
train_images, test_images = train_images / 255.0, test_images / 255.0
class_names = ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck']
```

In [ ]:
```
model = models.Sequential()
model.add(layers.Conv2D(32,(5,5),activation='relu',input_shape=(32,32,3)))
model.add(layers.MaxPool2D((2,2)))
model.add(layers.Conv2D(64,(3,3),activation='relu'))
model.add(layers.MaxPool2D((2,2)))
model.add(layers.Conv2D(128,(3,3),activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64,activation='relu'))
model.add(layers.Dense(10))

model.compile(optimizer=keras.optimizers.Adam(learning_rate=0.001),loss = tf.keras.losses.SparseCategoricalCrossentr
print(model.summary())

model.fit(train_images,train_labels,epochs=5)
test_loss,train_accuracy=model.evaluate(test_images,test_labels,verbose=2)
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_2 (Conv2D)           (None, 28, 28, 32)        2432      
                                                                 
 max_pooling2d (MaxPooling2D  (None, 14, 14, 32)       0         
 )                                                               
                                                                 
 conv2d_3 (Conv2D)           (None, 12, 12, 64)        18496     
                                                                 
 max_pooling2d_1 (MaxPooling  (None, 6, 6, 64)         0         
 2D)                                                             
                                                                 
 conv2d_4 (Conv2D)           (None, 4, 4, 128)         73856     
                                                                 
 flatten_1 (Flatten)         (None, 2048)              0         
                                                                 
 dense_3 (Dense)             (None, 64)                131136    
                                                                 
 dense_4 (Dense)             (None, 10)                650       
                                                                 
=================================================================
Total params: 226,570
Trainable params: 226,570
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/5
1563/1563 [==============================] - 30s 19ms/step - loss: 1.5492 - accuracy: 0.4368
Epoch 2/5
1563/1563 [==============================] - 32s 20ms/step - loss: 1.1972 - accuracy: 0.5759
Epoch 3/5
1563/1563 [==============================] - 32s 20ms/step - loss: 1.0430 - accuracy: 0.6349
Epoch 4/5
1563/1563 [==============================] - 33s 21ms/step - loss: 0.9348 - accuracy: 0.6685
Epoch 5/5
1563/1563 [==============================] - 32s 21ms/step - loss: 0.8409 - accuracy: 0.7063
313/313 - 2s - loss: 0.9337 - accuracy: 0.6762 - 2s/epoch - 6ms/step
```

Question 3 - Implementing the "model_base" network

In [ ]:
```python
mnist = keras.datasets.mnist
(train_images, train_labels), (test_images, test_labels) = mnist.load_data()

# Padding
paddings = tf.constant([[0, 0], [2, 2], [2, 2]])
train_images = tf.pad(train_images, paddings, constant_values=0)
test_images = tf.pad(test_images, paddings, constant_values=0)

print('train_images.shape: ', train_images.shape)
print('train_labels.shape: ', train_labels.shape)
print('test_images.shape:', test_images.shape)
print('test_labels.shape:', test_labels.shape)
class_names = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

train_images = tf.dtypes.cast(train_images, tf.float32)
test_images = tf.dtypes.cast(test_images, tf.float32)
train_images, test_images = train_images[..., np.newaxis]/255.0, test_images[..., np.newaxis]/255.0

model_base = models.Sequential()
model_base.add(layers.Conv2D(32,(3,3),activation='relu',input_shape=(32,32,1)))
model_base.add(layers.MaxPool2D((2,2)))
model_base.add(layers.Conv2D(64,(3,3),activation='relu'))
model_base.add(layers.MaxPool2D((2,2)))
model_base.add(layers.Conv2D(64,(3,3),activation='relu'))
model_base.add(layers.Flatten())
model_base.add(layers.Dense(64,activation='relu'))
model_base.add(layers.Dense(10))

model_base.compile(optimizer=keras.optimizers.Adam(),loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logit
print(model_base.summary())

model_base.fit(train_images,train_labels,epochs=2)
test_loss,train_accuracy=model_base.evaluate(test_images,test_labels,verbose=2)
model_base.save_weights('saved_model_weights/')
```

```
train_images.shape:  (60000, 32, 32)
train_labels.shape:  (60000,)
test_images.shape: (10000, 32, 32)
test_labels.shape: (10000,)
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_5 (Conv2D)           (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_2 (MaxPooling  (None, 15, 15, 32)       0         
 2D)                                                             
                                                                 
 conv2d_6 (Conv2D)           (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_3 (MaxPooling  (None, 6, 6, 64)         0         
 2D)                                                             
                                                                 
 conv2d_7 (Conv2D)           (None, 4, 4, 64)          36928     
                                                                 
 flatten_2 (Flatten)         (None, 1024)              0         
                                                                 
 dense_5 (Dense)             (None, 64)                65600     
                                                                 
 dense_6 (Dense)             (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/2
1875/1875 [==============================] - 29s 15ms/step - loss: 0.1411 - accuracy: 0.9557
Epoch 2/2
1875/1875 [==============================] - 33s 18ms/step - loss: 0.0433 - accuracy: 0.9868
313/313 - 2s - loss: 0.0301 - accuracy: 0.9910 - 2s/epoch - 6ms/step
```

Question 4 - Using saved weights on "model_lw"

In [ ]:

```
model_lw = models.Sequential()
model_lw.add(layers.Conv2D(32,(3,3),activation='relu',input_shape=(32,32,1)))
model_lw.add(layers.MaxPool2D((2,2)))
model_lw.add(layers.Conv2D(64,(3,3),activation='relu'))
model_lw.add(layers.MaxPool2D((2,2)))
model_lw.add(layers.Conv2D(64,(3,3),activation='relu'))
model_lw.add(layers.Flatten())
model_lw.add(layers.Dense(64,activation='relu'))
model_lw.add(layers.Dense(10))

model_lw.compile(optimizer=keras.optimizers.Adam(),loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=
print(model_lw.summary())

model_lw.load_weights('saved_model_weights/')

model_lw.fit(train_images,train_labels,epochs=2)
test_loss,train_accuracy=model_lw.evaluate(test_images,test_labels,verbose=2)

model_lw.save('saved_model/')
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_8 (Conv2D)           (None, 30, 30, 32)        320

 max_pooling2d_4 (MaxPooling  (None, 15, 15, 32)       0
 2D)

 conv2d_9 (Conv2D)           (None, 13, 13, 64)        18496

 max_pooling2d_5 (MaxPooling  (None, 6, 6, 64)         0
 2D)

 conv2d_10 (Conv2D)          (None, 4, 4, 64)          36928

 flatten_3 (Flatten)         (None, 1024)              0

 dense_7 (Dense)             (None, 64)                65600

 dense_8 (Dense)             (None, 10)                650

=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/2
1875/1875 [==============================] - 29s 15ms/step - loss: 0.0297 - accuracy: 0.9903
Epoch 2/2
1875/1875 [==============================] - 33s 18ms/step - loss: 0.0228 - accuracy: 0.9925
313/313 - 2s - loss: 0.0353 - accuracy: 0.9880 - 2s/epoch - 6ms/step
WARNING:absl:Found untraced functions such as _jit_compiled_convolution_op, _jit_compiled_convolution_op, _jit_compi
led_convolution_op while saving (showing 3 of 3). These functions will not be directly callable after loading.
INFO:tensorflow:Assets written to: saved_model/assets
INFO:tensorflow:Assets written to: saved_model/assets
```

Question 5 - Load "model_lw"

In [ ]:
```
model_ld = keras.models.load_model('saved_model/')
print(model_ld.summary())
model_ld.evaluate(test_images,test_labels,verbose=2)
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_8 (Conv2D)           (None, 30, 30, 32)        320

 max_pooling2d_4 (MaxPooling  (None, 15, 15, 32)       0
 2D)

 conv2d_9 (Conv2D)           (None, 13, 13, 64)        18496

 max_pooling2d_5 (MaxPooling  (None, 6, 6, 64)         0
 2D)

 conv2d_10 (Conv2D)          (None, 4, 4, 64)          36928

 flatten_3 (Flatten)         (None, 1024)              0

 dense_7 (Dense)             (None, 64)                65600

 dense_8 (Dense)             (None, 10)                650

=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
313/313 - 2s - loss: 0.0353 - accuracy: 0.9880 - 2s/epoch - 6ms/step
```

Out[ ]:
```
[0.035340044647455215, 0.9879999756813049]
```

Question 6 - Transfer Learning Example

In [ ]:
```python
base_inputs = model_ld.layers[0].input
base_outputs = model_ld.layers[-2].output
output = layers.Dense(10)(base_outputs)

new_model = keras.Model(inputs = base_inputs,outputs=output)
new_model.compile(optimizer=keras.optimizers.Adam(),loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits
print(new_model.summary())

new_model.fit(train_images,train_labels,epochs=2,verbose=2)
new_model.evaluate(test_images,test_labels,verbose=2)
```

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_8_input (InputLayer)  [(None, 32, 32, 1)]      0         
                                                                 
 conv2d_8 (Conv2D)           (None, 30, 30, 32)        320       
                                                                 
 max_pooling2d_4 (MaxPooling  (None, 15, 15, 32)       0         
 2D)                                                             
                                                                 
 conv2d_9 (Conv2D)           (None, 13, 13, 64)        18496     
                                                                 
 max_pooling2d_5 (MaxPooling  (None, 6, 6, 64)         0         
 2D)                                                             
                                                                 
 conv2d_10 (Conv2D)          (None, 4, 4, 64)          36928     
                                                                 
 flatten_3 (Flatten)         (None, 1024)              0         
                                                                 
 dense_7 (Dense)             (None, 64)                65600     
                                                                 
 dense_9 (Dense)             (None, 10)                650       
                                                                 
=================================================================
Total params: 121,994
Trainable params: 121,994
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/2
1875/1875 - 26s - loss: 0.0850 - accuracy: 0.9773 - 26s/epoch - 14ms/step
Epoch 2/2
1875/1875 - 29s - loss: 0.0192 - accuracy: 0.9942 - 29s/epoch - 16ms/step
313/313 - 2s - loss: 0.0302 - accuracy: 0.9910 - 2s/epoch - 6ms/step
```

Out[ ]:
```
[0.030212702229619026, 0.9909999966621399]
```

Question 7 - Fine Tuning

In [ ]:
```python
model_for_tl = keras.models.load_model('saved_model/')
model_for_tl.trainable = False

for layer in model_for_tl.layers:
    assert layer.trainable == False

base_inputs = model_for_tl.layers[0].input
base_outputs = model_for_tl.layers[-2].output
output = layers.Dense(10)(base_outputs)

new_model = keras.Model(inputs = base_inputs,outputs=output)
new_model.compile(optimizer=keras.optimizers.Adam(),loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits

new_model.fit(train_images,train_labels,epochs=2,verbose=2)
new_model.evaluate(test_images,test_labels,verbose=2)
```

```
Epoch 1/2
1875/1875 - 10s - loss: 0.2110 - accuracy: 0.9537 - 10s/epoch - 6ms/step
Epoch 2/2
1875/1875 - 10s - loss: 0.0156 - accuracy: 0.9954 - 10s/epoch - 5ms/step
313/313 - 2s - loss: 0.0272 - accuracy: 0.9918 - 2s/epoch - 5ms/step
```

Out[ ]:
```
[0.027205144986510277, 0.9918000102043152]
```

Question 8 - Transfer learn a ResNet model

In [ ]:
```python
#Import the pre-trained model as non trainable layers
pretrained_model= tf.keras.applications.ResNet50(include_top=True)
```

In [ ]:
```
# Creating sample images to train the model
# ResNet50 expects image size of (224,224,3)
# Number of classes is chosen as 5
sample_images = tf.random.normal(shape=(5,224,224,3))
sample_labels = tf.constant([0,1,2,3,4])

# Adding 5 additional output nodes
base_inputs = pretrained_model.layers[0].input
base_outputs = pretrained_model.layers[-2].output
output = layers.Dense(5)(base_outputs)

# Compile the model with new input, output layers
resnet_model = keras.Model(inputs = base_inputs,outputs=output)
resnet_model.compile(optimizer=keras.optimizers.Adam(),loss = tf.keras.losses.SparseCategoricalCrossentropy(from_log
print(resnet_model.summary())

# Transfer learn
# Do 10 tests
resnet_model.fit(sample_images,sample_labels,epochs=10,verbose=2)
```

```
Model: "model_1"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to                     
==================================================================================================
 input_2 (InputLayer)           [(None, 224, 224, 3  0           []                               
                                )]                                                                
                                                                                                  
 conv1_pad (ZeroPadding2D)      (None, 230, 230, 3)  0           ['input_2[0][0]']                
                                                                                                  
 conv1_conv (Conv2D)            (None, 112, 112, 64  9472        ['conv1_pad[0][0]']              
                                )                                                                 
                                                                                                  
 conv1_bn (BatchNormalization)  (None, 112, 112, 64  256         ['conv1_conv[0][0]']             
                                )                                                                 
                                                                                                  
 conv1_relu (Activation)        (None, 112, 112, 64  0           ['conv1_bn[0][0]']               
                                )                                                                 
                                                                                                  
 pool1_pad (ZeroPadding2D)      (None, 114, 114, 64  0           ['conv1_relu[0][0]']             
                                )                                                                 
                                                                                                  
 pool1_pool (MaxPooling2D)      (None, 56, 56, 64)   0           ['pool1_pad[0][0]']              
                                                                                                  
 conv2_block1_1_conv (Conv2D)   (None, 56, 56, 64)   4160        ['pool1_pool[0][0]']             
                                                                                                  
 conv2_block1_1_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block1_1_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block1_1_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block1_1_bn[0][0]']      
 n)                                                                                               
                                                                                                  
 conv2_block1_2_conv (Conv2D)   (None, 56, 56, 64)   36928       ['conv2_block1_1_relu[0][0]']    
                                                                                                  
 conv2_block1_2_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block1_2_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block1_2_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block1_2_bn[0][0]']      
 n)                                                                                               
                                                                                                  
 conv2_block1_0_conv (Conv2D)   (None, 56, 56, 256)  16640       ['pool1_pool[0][0]']             
                                                                                                  
 conv2_block1_3_conv (Conv2D)   (None, 56, 56, 256)  16640       ['conv2_block1_2_relu[0][0]']    
                                                                                                  
 conv2_block1_0_bn (BatchNormal  (None, 56, 56, 256)  1024       ['conv2_block1_0_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block1_3_bn (BatchNormal  (None, 56, 56, 256)  1024       ['conv2_block1_3_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block1_add (Add)         (None, 56, 56, 256)  0           ['conv2_block1_0_bn[0][0]',      
                                                                  'conv2_block1_3_bn[0][0]']      
                                                                                                  
 conv2_block1_out (Activation)  (None, 56, 56, 256)  0           ['conv2_block1_add[0][0]']       
                                                                                                  
 conv2_block2_1_conv (Conv2D)   (None, 56, 56, 64)   16448       ['conv2_block1_out[0][0]']       
                                                                                                  
 conv2_block2_1_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block2_1_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block2_1_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block2_1_bn[0][0]']      
 n)                                                                                               
                                                                                                  
 conv2_block2_2_conv (Conv2D)   (None, 56, 56, 64)   36928       ['conv2_block2_1_relu[0][0]']    
                                                                                                  
 conv2_block2_2_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block2_2_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block2_2_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block2_2_bn[0][0]']      
 n)                                                                                               
                                                                                                  
 conv2_block2_3_conv (Conv2D)   (None, 56, 56, 256)  16640       ['conv2_block2_2_relu[0][0]']    
                                                                                                  
 conv2_block2_3_bn (BatchNormal  (None, 56, 56, 256)  1024       ['conv2_block2_3_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block2_add (Add)         (None, 56, 56, 256)  0           ['conv2_block1_out[0][0]',       
                                                                  'conv2_block2_3_bn[0][0]']      
                                                                                                  
 conv2_block2_out (Activation)  (None, 56, 56, 256)  0           ['conv2_block2_add[0][0]']       
                                                                                                  
 conv2_block3_1_conv (Conv2D)   (None, 56, 56, 64)   16448       ['conv2_block2_out[0][0]']       
                                                                                                  
 conv2_block3_1_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block3_1_conv[0][0]']    
 ization)                                                                                         
                                                                                                  
 conv2_block3_1_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block3_1_bn[0][0]']      
 n)

 conv2_block3_2_conv (Conv2D)   (None, 56, 56, 64)   36928       ['conv2_block3_1_relu[0][0]']

 conv2_block3_2_bn (BatchNormal  (None, 56, 56, 64)  256         ['conv2_block3_2_conv[0][0]']
 ization)

 conv2_block3_2_relu (Activatio  (None, 56, 56, 64)  0           ['conv2_block3_2_bn[0][0]']
 n)

 conv2_block3_3_conv (Conv2D)   (None, 56, 56, 256)  16640       ['conv2_block3_2_relu[0][0]']

 conv2_block3_3_bn (BatchNormal  (None, 56, 56, 256)  1024       ['conv2_block3_3_conv[0][0]']
 ization)

 conv2_block3_add (Add)         (None, 56, 56, 256)  0           ['conv2_block2_out[0][0]',
                                                                   'conv2_block3_3_bn[0][0]']

 conv2_block3_out (Activation)  (None, 56, 56, 256)  0           ['conv2_block3_add[0][0]']

 conv3_block1_1_conv (Conv2D)   (None, 28, 28, 128)  32896       ['conv2_block3_out[0][0]']

 conv3_block1_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block1_1_conv[0][0]']
 ization)

 conv3_block1_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block1_1_bn[0][0]']
 n)

 conv3_block1_2_conv (Conv2D)   (None, 28, 28, 128)  147584      ['conv3_block1_1_relu[0][0]']

 conv3_block1_2_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block1_2_conv[0][0]']
 ization)

 conv3_block1_2_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block1_2_bn[0][0]']
 n)

 conv3_block1_0_conv (Conv2D)   (None, 28, 28, 512)  131584      ['conv2_block3_out[0][0]']

 conv3_block1_3_conv (Conv2D)   (None, 28, 28, 512)  66048       ['conv3_block1_2_relu[0][0]']

 conv3_block1_0_bn (BatchNormal  (None, 28, 28, 512)  2048       ['conv3_block1_0_conv[0][0]']
 ization)

 conv3_block1_3_bn (BatchNormal  (None, 28, 28, 512)  2048       ['conv3_block1_3_conv[0][0]']
 ization)

 conv3_block1_add (Add)         (None, 28, 28, 512)  0           ['conv3_block1_0_bn[0][0]',
                                                                   'conv3_block1_3_bn[0][0]']

 conv3_block1_out (Activation)  (None, 28, 28, 512)  0           ['conv3_block1_add[0][0]']

 conv3_block2_1_conv (Conv2D)   (None, 28, 28, 128)  65664       ['conv3_block1_out[0][0]']

 conv3_block2_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block2_1_conv[0][0]']
 ization)

 conv3_block2_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block2_1_bn[0][0]']
 n)

 conv3_block2_2_conv (Conv2D)   (None, 28, 28, 128)  147584      ['conv3_block2_1_relu[0][0]']

 conv3_block2_2_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block2_2_conv[0][0]']
 ization)

 conv3_block2_2_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block2_2_bn[0][0]']
 n)

 conv3_block2_3_conv (Conv2D)   (None, 28, 28, 512)  66048       ['conv3_block2_2_relu[0][0]']

 conv3_block2_3_bn (BatchNormal  (None, 28, 28, 512)  2048       ['conv3_block2_3_conv[0][0]']
 ization)

 conv3_block2_add (Add)         (None, 28, 28, 512)  0           ['conv3_block1_out[0][0]',
                                                                   'conv3_block2_3_bn[0][0]']

 conv3_block2_out (Activation)  (None, 28, 28, 512)  0           ['conv3_block2_add[0][0]']

 conv3_block3_1_conv (Conv2D)   (None, 28, 28, 128)  65664       ['conv3_block2_out[0][0]']

 conv3_block3_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block3_1_conv[0][0]']
 ization)

 conv3_block3_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block3_1_bn[0][0]']
 n)

 conv3_block3_2_conv (Conv2D)   (None, 28, 28, 128)  147584      ['conv3_block3_1_relu[0][0]']
```

```
conv3_block3_2_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block3_2_conv[0][0]']
ization)

conv3_block3_2_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block3_2_bn[0][0]']
n)

conv3_block3_3_conv (Conv2D)   (None, 28, 28, 512)  66048       ['conv3_block3_2_relu[0][0]']

conv3_block3_3_bn (BatchNormal  (None, 28, 28, 512)  2048        ['conv3_block3_3_conv[0][0]']
ization)

conv3_block3_add (Add)         (None, 28, 28, 512)  0           ['conv3_block2_out[0][0]',
                                                                 'conv3_block3_3_bn[0][0]']

conv3_block3_out (Activation)  (None, 28, 28, 512)  0           ['conv3_block3_add[0][0]']

conv3_block4_1_conv (Conv2D)   (None, 28, 28, 128)  65664       ['conv3_block3_out[0][0]']

conv3_block4_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block4_1_conv[0][0]']
ization)

conv3_block4_1_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block4_1_bn[0][0]']
n)

conv3_block4_2_conv (Conv2D)   (None, 28, 28, 128)  147584      ['conv3_block4_1_relu[0][0]']

conv3_block4_2_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block4_2_conv[0][0]']
ization)

conv3_block4_2_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block4_2_bn[0][0]']
n)

conv3_block4_3_conv (Conv2D)   (None, 28, 28, 512)  66048       ['conv3_block4_2_relu[0][0]']

conv3_block4_3_bn (BatchNormal  (None, 28, 28, 512)  2048        ['conv3_block4_3_conv[0][0]']
ization)

conv3_block4_add (Add)         (None, 28, 28, 512)  0           ['conv3_block3_out[0][0]',
                                                                 'conv3_block4_3_bn[0][0]']

conv3_block4_out (Activation)  (None, 28, 28, 512)  0           ['conv3_block4_add[0][0]']

conv4_block1_1_conv (Conv2D)   (None, 14, 14, 256)  131328      ['conv3_block4_out[0][0]']

conv4_block1_1_bn (BatchNormal  (None, 14, 14, 256)  1024        ['conv4_block1_1_conv[0][0]']
ization)

conv4_block1_1_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block1_1_bn[0][0]']
n)

conv4_block1_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block1_1_relu[0][0]']

conv4_block1_2_bn (BatchNormal  (None, 14, 14, 256)  1024        ['conv4_block1_2_conv[0][0]']
ization)

conv4_block1_2_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block1_2_bn[0][0]']
n)

conv4_block1_0_conv (Conv2D)   (None, 14, 14, 1024  525312      ['conv3_block4_out[0][0]']
                               )

conv4_block1_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block1_2_relu[0][0]']
                               )

conv4_block1_0_bn (BatchNormal  (None, 14, 14, 1024  4096        ['conv4_block1_0_conv[0][0]']
ization)                       )

conv4_block1_3_bn (BatchNormal  (None, 14, 14, 1024  4096        ['conv4_block1_3_conv[0][0]']
ization)                       )

conv4_block1_add (Add)         (None, 14, 14, 1024  0           ['conv4_block1_0_bn[0][0]',
                               )                                 'conv4_block1_3_bn[0][0]']

conv4_block1_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block1_add[0][0]']
                               )

conv4_block2_1_conv (Conv2D)   (None, 14, 14, 256)  262400      ['conv4_block1_out[0][0]']

conv4_block2_1_bn (BatchNormal  (None, 14, 14, 256)  1024        ['conv4_block2_1_conv[0][0]']
ization)

conv4_block2_1_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block2_1_bn[0][0]']
n)

conv4_block2_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block2_1_relu[0][0]']
```

```
conv4_block2_2_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block2_2_conv[0][0]']
ization)

conv4_block2_2_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block2_2_bn[0][0]']
n)

conv4_block2_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block2_2_relu[0][0]']
                               )

conv4_block2_3_bn (BatchNormal  (None, 14, 14, 1024  4096       ['conv4_block2_3_conv[0][0]']
ization)                       )

conv4_block2_add (Add)         (None, 14, 14, 1024  0           ['conv4_block1_out[0][0]',
                               )                                 'conv4_block2_3_bn[0][0]']

conv4_block2_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block2_add[0][0]']
                               )

conv4_block3_1_conv (Conv2D)   (None, 14, 14, 256)  262400      ['conv4_block2_out[0][0]']

conv4_block3_1_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block3_1_conv[0][0]']
ization)

conv4_block3_1_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block3_1_bn[0][0]']
n)

conv4_block3_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block3_1_relu[0][0]']

conv4_block3_2_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block3_2_conv[0][0]']
ization)

conv4_block3_2_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block3_2_bn[0][0]']
n)

conv4_block3_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block3_2_relu[0][0]']
                               )

conv4_block3_3_bn (BatchNormal  (None, 14, 14, 1024  4096       ['conv4_block3_3_conv[0][0]']
ization)                       )

conv4_block3_add (Add)         (None, 14, 14, 1024  0           ['conv4_block2_out[0][0]',
                               )                                 'conv4_block3_3_bn[0][0]']

conv4_block3_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block3_add[0][0]']
                               )

conv4_block4_1_conv (Conv2D)   (None, 14, 14, 256)  262400      ['conv4_block3_out[0][0]']

conv4_block4_1_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block4_1_conv[0][0]']
ization)

conv4_block4_1_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block4_1_bn[0][0]']
n)

conv4_block4_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block4_1_relu[0][0]']

conv4_block4_2_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block4_2_conv[0][0]']
ization)

conv4_block4_2_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block4_2_bn[0][0]']
n)

conv4_block4_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block4_2_relu[0][0]']
                               )

conv4_block4_3_bn (BatchNormal  (None, 14, 14, 1024  4096       ['conv4_block4_3_conv[0][0]']
ization)                       )

conv4_block4_add (Add)         (None, 14, 14, 1024  0           ['conv4_block3_out[0][0]',
                               )                                 'conv4_block4_3_bn[0][0]']

conv4_block4_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block4_add[0][0]']
                               )

conv4_block5_1_conv (Conv2D)   (None, 14, 14, 256)  262400      ['conv4_block4_out[0][0]']

conv4_block5_1_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block5_1_conv[0][0]']
ization)

conv4_block5_1_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block5_1_bn[0][0]']
n)

conv4_block5_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block5_1_relu[0][0]']

conv4_block5_2_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block5_2_conv[0][0]']
ization)

conv4_block5_2_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block5_2_bn[0][0]']
n)

conv4_block5_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block5_2_relu[0][0]']
                               )

conv4_block5_3_bn (BatchNormal  (None, 14, 14, 1024  4096       ['conv4_block5_3_conv[0][0]']
ization)                       )

conv4_block5_add (Add)         (None, 14, 14, 1024  0           ['conv4_block4_out[0][0]',
                               )                                 'conv4_block5_3_bn[0][0]']

conv4_block5_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block5_add[0][0]']
                               )

conv4_block6_1_conv (Conv2D)   (None, 14, 14, 256)  262400      ['conv4_block5_out[0][0]']

conv4_block6_1_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block6_1_conv[0][0]']
ization)

conv4_block6_1_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block6_1_bn[0][0]']
n)

conv4_block6_2_conv (Conv2D)   (None, 14, 14, 256)  590080      ['conv4_block6_1_relu[0][0]']

conv4_block6_2_bn (BatchNormal  (None, 14, 14, 256)  1024       ['conv4_block6_2_conv[0][0]']
ization)

conv4_block6_2_relu (Activatio  (None, 14, 14, 256)  0           ['conv4_block6_2_bn[0][0]']
n)

conv4_block6_3_conv (Conv2D)   (None, 14, 14, 1024  263168      ['conv4_block6_2_relu[0][0]']
                               )

conv4_block6_3_bn (BatchNormal  (None, 14, 14, 1024  4096       ['conv4_block6_3_conv[0][0]']
ization)                       )

conv4_block6_add (Add)         (None, 14, 14, 1024  0           ['conv4_block5_out[0][0]',
                               )                                 'conv4_block6_3_bn[0][0]']

conv4_block6_out (Activation)  (None, 14, 14, 1024  0           ['conv4_block6_add[0][0]']
                               )

conv5_block1_1_conv (Conv2D)   (None, 7, 7, 512)    524800      ['conv4_block6_out[0][0]']

conv5_block1_1_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block1_1_conv[0][0]']
ization)

conv5_block1_1_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block1_1_bn[0][0]']
n)

conv5_block1_2_conv (Conv2D)   (None, 7, 7, 512)    2359808     ['conv5_block1_1_relu[0][0]']

conv5_block1_2_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block1_2_conv[0][0]']
ization)

conv5_block1_2_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block1_2_bn[0][0]']
n)

conv5_block1_0_conv (Conv2D)   (None, 7, 7, 2048)   2099200     ['conv4_block6_out[0][0]']

conv5_block1_3_conv (Conv2D)   (None, 7, 7, 2048)   1050624     ['conv5_block1_2_relu[0][0]']

conv5_block1_0_bn (BatchNormal  (None, 7, 7, 2048)  8192        ['conv5_block1_0_conv[0][0]']
ization)

conv5_block1_3_bn (BatchNormal  (None, 7, 7, 2048)  8192        ['conv5_block1_3_conv[0][0]']
ization)

conv5_block1_add (Add)         (None, 7, 7, 2048)   0           ['conv5_block1_0_bn[0][0]',
                                                                 'conv5_block1_3_bn[0][0]']

conv5_block1_out (Activation)  (None, 7, 7, 2048)   0           ['conv5_block1_add[0][0]']

conv5_block2_1_conv (Conv2D)   (None, 7, 7, 512)    1049088     ['conv5_block1_out[0][0]']

conv5_block2_1_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block2_1_conv[0][0]']
ization)

conv5_block2_1_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block2_1_bn[0][0]']
n)

conv5_block2_2_conv (Conv2D)   (None, 7, 7, 512)    2359808     ['conv5_block2_1_relu[0][0]']
```

```
 conv5_block2_2_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block2_2_conv[0][0]']
 ization)

 conv5_block2_2_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block2_2_bn[0][0]']
 n)

 conv5_block2_3_conv (Conv2D)   (None, 7, 7, 2048)   1050624     ['conv5_block2_2_relu[0][0]']

 conv5_block2_3_bn (BatchNormal  (None, 7, 7, 2048)  8192        ['conv5_block2_3_conv[0][0]']
 ization)

 conv5_block2_add (Add)         (None, 7, 7, 2048)   0           ['conv5_block1_out[0][0]',
                                                                  'conv5_block2_3_bn[0][0]']

 conv5_block2_out (Activation)  (None, 7, 7, 2048)   0           ['conv5_block2_add[0][0]']

 conv5_block3_1_conv (Conv2D)   (None, 7, 7, 512)    1049088     ['conv5_block2_out[0][0]']

 conv5_block3_1_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block3_1_conv[0][0]']
 ization)

 conv5_block3_1_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block3_1_bn[0][0]']
 n)

 conv5_block3_2_conv (Conv2D)   (None, 7, 7, 512)    2359808     ['conv5_block3_1_relu[0][0]']

 conv5_block3_2_bn (BatchNormal  (None, 7, 7, 512)   2048        ['conv5_block3_2_conv[0][0]']
 ization)

 conv5_block3_2_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block3_2_bn[0][0]']
 n)

 conv5_block3_3_conv (Conv2D)   (None, 7, 7, 2048)   1050624     ['conv5_block3_2_relu[0][0]']

 conv5_block3_3_bn (BatchNormal  (None, 7, 7, 2048)  8192        ['conv5_block3_3_conv[0][0]']
 ization)

 conv5_block3_add (Add)         (None, 7, 7, 2048)   0           ['conv5_block2_out[0][0]',
                                                                  'conv5_block3_3_bn[0][0]']

 conv5_block3_out (Activation)  (None, 7, 7, 2048)   0           ['conv5_block3_add[0][0]']

 avg_pool (GlobalAveragePooling  (None, 2048)        0           ['conv5_block3_out[0][0]']
 2D)

 dense_1 (Dense)                (None, 5)            10245       ['avg_pool[0][0]']

==================================================================================================
Total params: 23,597,957
Trainable params: 23,544,837
Non-trainable params: 53,120
__________________________________________________________________________________________________
None
Epoch 1/10
1/1 - 8s - loss: 2.0120 - accuracy: 0.0000e+00 - 8s/epoch - 8s/step
Epoch 2/10
1/1 - 2s - loss: 0.0119 - accuracy: 1.0000 - 2s/epoch - 2s/step
Epoch 3/10
1/1 - 2s - loss: 3.7076e-04 - accuracy: 1.0000 - 2s/epoch - 2s/step
Epoch 4/10
1/1 - 2s - loss: 1.8584e-04 - accuracy: 1.0000 - 2s/epoch - 2s/step
Epoch 5/10
1/1 - 2s - loss: 1.3722e-04 - accuracy: 1.0000 - 2s/epoch - 2s/step
Epoch 6/10
1/1 - 1s - loss: 1.0692e-04 - accuracy: 1.0000 - 1s/epoch - 1s/step
Epoch 7/10
1/1 - 1s - loss: 9.0211e-05 - accuracy: 1.0000 - 1s/epoch - 1s/step
Epoch 8/10
1/1 - 1s - loss: 8.1414e-05 - accuracy: 1.0000 - 1s/epoch - 1s/step
Epoch 9/10
1/1 - 1s - loss: 7.4477e-05 - accuracy: 1.0000 - 1s/epoch - 1s/step
Epoch 10/10
1/1 - 2s - loss: 6.8255e-05 - accuracy: 1.0000 - 2s/epoch - 2s/step
```

```
Out[ ]: <keras.callbacks.History at 0x1f3ca583f70>
```

- It can be observed that, Accuracy of the model is so high (100%) because we are using a pre-trained model for the test.