In [ ]:
```
import numpy as np
import h5py
from tensorflow.keras.layers import *
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.optimizers import Adam, SGD
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.metrics import AUC
import tensorflow as tf
import warnings
warnings.filterwarnings('ignore')
```

In [ ]:
```
file_electron = "SingleElectronPt50_IMGCROPS_n249k_RHv1.hdf5"
file_photon = "SinglePhotonPt50_IMGCROPS_n249k_RHv1.hdf5"


with h5py.File(file_electron, "r") as f1:
    X_elec = np.array(f1['X'][:])
    y_elec = np.array(f1['y'][:])
with h5py.File(file_photon, "r") as f2:
    X_phot = np.array(f2['X'][:])
    y_phot = np.array(f2['y'][:])
```

In [ ]:
```
print(X_elec.shape)
print(X_phot.shape)
```

```
(249000, 32, 32, 2)
(249000, 32, 32, 2)
```

In [ ]:
```
X = np.append(X_elec, X_phot, axis=0)
y = np.append(y_elec, y_phot)
X.shape
```

Out[ ]:
```
(498000, 32, 32, 2)
```

In [ ]:
```
X = np.swapaxes(X, 3,1)
X.shape
```

Out[ ]:
```
(498000, 2, 32, 32)
```

In [ ]:
```
X[:,0].shape
```

Out[ ]:
```
(498000, 32, 32)
```

In [ ]:
```
y = to_categorical(y, num_classes=2)
```

In [ ]:
```
input1 = Input(shape=(32,32,1))
input2 = Input(shape=(32,32,1))

conv1 = Conv2D(3, (4,4), activation='relu', input_shape=(32,32,1), padding='same')(input1)
conv2 = Conv2D(3, (4,4), activation='relu', input_shape=(32,32,1), padding='same')(input2)
x1 = MaxPooling2D((4,4))(conv1)
x2 = MaxPooling2D((4,4))(conv2)
x1 = Flatten()(x1)
x2 = Flatten()(x2)

x1 = Dense(32, activation='relu')(x1)

x2 = Dense(32, activation='relu')(x2)

x = Concatenate(axis=1)([x1, x2])
x = Dense(8, activation='relu')(x)
output = Dense(2, activation='softmax')(x)
model = Model(inputs=[input1, input2], outputs=output)
```

```
2022-03-24 21:10:11.762364: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:11.778410: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:11.781363: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:11.785380: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimized with oneAPI D
eep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations:  AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-03-24 21:10:11.787689: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:11.790520: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:11.793075: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:12.725689: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
```

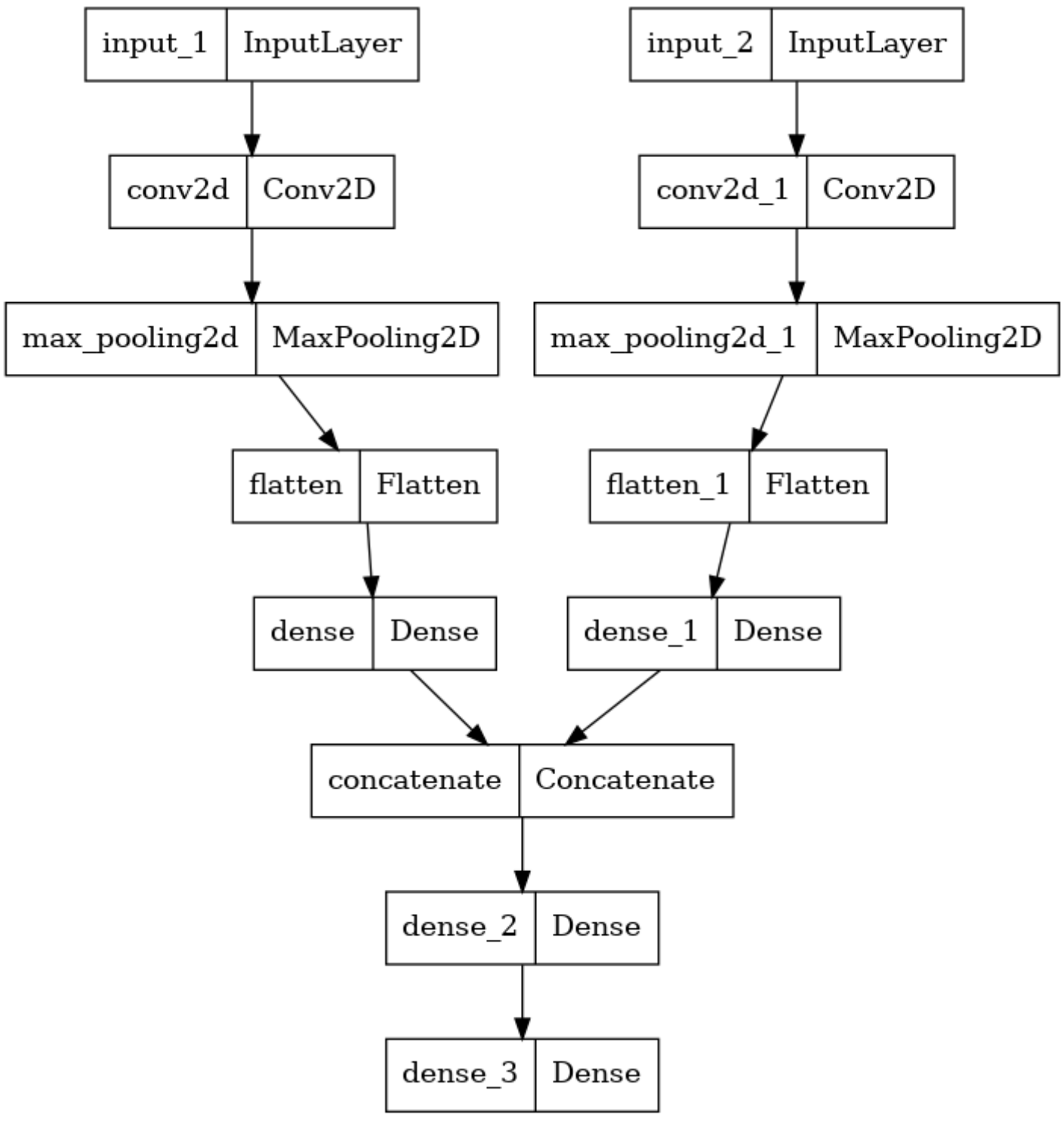
```
2022-03-24 21:10:12.728464: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:12.731152: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-24 21:10:12.733702: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1525] Created device /job:localhost/replica:0/task:
0/device:GPU:0 with 47216 MB memory:  -> device: 0, name: Quadro RTX 8000, pci bus id: 0000:04:00.0, compute capability: 7.5
```

In [ ]:

```
model.summary()
tf.keras.utils.plot_model(model)
```

```
Model: "model"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_1 (InputLayer)           [(None, 32, 32, 1)]  0           []

 input_2 (InputLayer)           [(None, 32, 32, 1)]  0           []

 conv2d (Conv2D)                (None, 32, 32, 3)    51          ['input_1[0][0]']

 conv2d_1 (Conv2D)              (None, 32, 32, 3)    51          ['input_2[0][0]']

 max_pooling2d (MaxPooling2D)   (None, 8, 8, 3)      0           ['conv2d[0][0]']

 max_pooling2d_1 (MaxPooling2D)  (None, 8, 8, 3)     0           ['conv2d_1[0][0]']

 flatten (Flatten)              (None, 192)          0           ['max_pooling2d[0][0]']

 flatten_1 (Flatten)            (None, 192)          0           ['max_pooling2d_1[0][0]']

 dense (Dense)                  (None, 32)           6176        ['flatten[0][0]']

 dense_1 (Dense)                (None, 32)           6176        ['flatten_1[0][0]']

 concatenate (Concatenate)      (None, 64)           0           ['dense[0][0]',
                                                                  'dense_1[0][0]']

 dense_2 (Dense)                (None, 8)            520         ['concatenate[0][0]']

 dense_3 (Dense)                (None, 2)            18          ['dense_2[0][0]']

==================================================================================================
Total params: 12,992
Trainable params: 12,992
Non-trainable params: 0
__________________________________________________________________________________________________
```

Out[ ]:



In [ ]:
```
model.compile(loss="categorical_crossentropy", optimizer=Adam(learning_rate=0.002), metrics=[AUC()])
```

In [ ]:
```
with tf.device('/gpu:0'):
    model.fit([X[:,0], X[:,1]], y, epochs=40, batch_size=256, validation_split=0.1)
```

```
Epoch 1/40
2022-03-24 21:10:26.309691: I tensorflow/stream_executor/cuda/cuda_dnn.cc:368] Loaded cuDNN version 8101
2022-03-24 21:10:26.992375: I tensorflow/core/platform/default/subprocess.cc:304] Start cannot spawn child process: No such file o
r directory
1751/1751 [==============================] - 23s 11ms/step - loss: 0.6147 - auc: 0.7237 - val_loss: 0.6780 - val_auc: 0.6438
Epoch 2/40
1751/1751 [==============================] - 18s 10ms/step - loss: 0.5762 - auc: 0.7693 - val_loss: 0.6275 - val_auc: 0.7102
Epoch 3/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5684 - auc: 0.7772 - val_loss: 0.6662 - val_auc: 0.6667
Epoch 4/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5651 - auc: 0.7804 - val_loss: 0.6387 - val_auc: 0.6958
Epoch 5/40
1751/1751 [==============================] - 18s 10ms/step - loss: 0.5627 - auc: 0.7826 - val_loss: 0.6389 - val_auc: 0.7044
Epoch 6/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5610 - auc: 0.7840 - val_loss: 0.6710 - val_auc: 0.6637
Epoch 7/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5599 - auc: 0.7851 - val_loss: 0.7462 - val_auc: 0.5803
Epoch 8/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5581 - auc: 0.7868 - val_loss: 0.6585 - val_auc: 0.6805
Epoch 9/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5568 - auc: 0.7880 - val_loss: 0.6209 - val_auc: 0.7258
Epoch 10/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5553 - auc: 0.7894 - val_loss: 0.7347 - val_auc: 0.5997
Epoch 11/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5544 - auc: 0.7903 - val_loss: 0.6659 - val_auc: 0.6735
Epoch 12/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5537 - auc: 0.7909 - val_loss: 0.7014 - val_auc: 0.6419
Epoch 13/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5530 - auc: 0.7916 - val_loss: 0.6000 - val_auc: 0.7503
Epoch 14/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5522 - auc: 0.7924 - val_loss: 0.7512 - val_auc: 0.5937
Epoch 15/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5518 - auc: 0.7927 - val_loss: 0.6323 - val_auc: 0.7137
Epoch 16/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5507 - auc: 0.7937 - val_loss: 0.6436 - val_auc: 0.6975
Epoch 17/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5502 - auc: 0.7942 - val_loss: 0.5775 - val_auc: 0.7711
Epoch 18/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5499 - auc: 0.7944 - val_loss: 0.6683 - val_auc: 0.6782
Epoch 19/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5496 - auc: 0.7947 - val_loss: 0.6449 - val_auc: 0.7063
Epoch 20/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5489 - auc: 0.7954 - val_loss: 0.6737 - val_auc: 0.6672
Epoch 21/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5483 - auc: 0.7959 - val_loss: 0.6284 - val_auc: 0.7216
Epoch 22/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5478 - auc: 0.7964 - val_loss: 0.6332 - val_auc: 0.7055
Epoch 23/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5479 - auc: 0.7963 - val_loss: 0.5650 - val_auc: 0.7794
Epoch 24/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5469 - auc: 0.7972 - val_loss: 0.6223 - val_auc: 0.7202
Epoch 25/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5466 - auc: 0.7975 - val_loss: 0.5512 - val_auc: 0.7895
Epoch 26/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5461 - auc: 0.7979 - val_loss: 0.6686 - val_auc: 0.6779
Epoch 27/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5458 - auc: 0.7982 - val_loss: 0.6803 - val_auc: 0.6745
Epoch 28/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5458 - auc: 0.7982 - val_loss: 0.6163 - val_auc: 0.7340
Epoch 29/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5453 - auc: 0.7986 - val_loss: 0.6896 - val_auc: 0.6546
Epoch 30/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5453 - auc: 0.7986 - val_loss: 0.6515 - val_auc: 0.6950
Epoch 31/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5448 - auc: 0.7991 - val_loss: 0.6805 - val_auc: 0.6602
Epoch 32/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5445 - auc: 0.7993 - val_loss: 0.6648 - val_auc: 0.6725
Epoch 33/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5441 - auc: 0.7996 - val_loss: 0.7336 - val_auc: 0.6115
Epoch 34/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5440 - auc: 0.7998 - val_loss: 0.7102 - val_auc: 0.6259
Epoch 35/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5441 - auc: 0.7997 - val_loss: 0.6525 - val_auc: 0.6959
Epoch 36/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5435 - auc: 0.8001 - val_loss: 0.6577 - val_auc: 0.6891
Epoch 37/40
1751/1751 [==============================] - 19s 11ms/step - loss: 0.5434 - auc: 0.8003 - val_loss: 0.6273 - val_auc: 0.7217
Epoch 38/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5431 - auc: 0.8005 - val_loss: 0.6503 - val_auc: 0.7009
Epoch 39/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5433 - auc: 0.8003 - val_loss: 0.7254 - val_auc: 0.6194
Epoch 40/40
1751/1751 [==============================] - 18s 11ms/step - loss: 0.5427 - auc: 0.8008 - val_loss: 0.6816 - val_auc: 0.6805
```

As seen from the epoch metrics, we got:

Train AUC Score: 0.8008

Validation AUC Score: 0.6805

Best Validation AUC Score: 0.7895

We can save model, based on best `val_auc` score epoch.