In [1]:

```
from keras.models import Sequential
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
```

```
/home/rajesh/anaconda3/lib/python3.6/site-packages/h5py/__init__.py:3
6: FutureWarning: Conversion of the second argument of issubdtype fro
m `float` to `np.floating` is deprecated. In future, it will be treat
ed as `np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
Using TensorFlow backend.
```

In [2]:

```
#initialiseing the CNN
classifier = Sequential()
```

In [ ]:

```
#if image size requird then use this parameter
# img_width = 256
# img_height = 256
```

In [3]:

```
classifier.add(Conv2D( 32,(3,3), input_shape=( 64, 64, 3 ), activation = 'relu'

#when image size give replce input_shape with img_width and img_height
classifier.add(Conv2D( 32,(3,3), input_shape=( img_width, img_height, 3 ), acti
#32  filter (feature detector) then 64 then 128 like that , 3 * 3 fetaure detec
#input shape - forcet to fixed image -3D ALL colored 256,256,
#because we useing cpu s0 64,64,3
```

In [4]:

```
classifier.add(MaxPooling2D(pool_size = (2,2)))
```

In [5]:

```
# Adding a second convolutional layer
classifier.add(Conv2D(32, (3, 3), activation = 'relu'))
classifier.add(MaxPooling2D(pool_size = (2, 2)))
```

In [6]:

```
classifier.add(Flatten())
```

In [7]:

```
classifier.add(Dense(units = 128, activation='relu'))
classifier.add(Dense(units = 1, activation='sigmoid'))
```

In [8]:

```
classifier.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accu
```

In [9]:

```python
#https://keras.io/preprocessing/image/
from keras.preprocessing.image import  ImageDataGenerator

train_datagen = ImageDataGenerator(
        rescale=1./255,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True)


test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
        'demodata/humandata/humantrain',
        #when the image perticular size given then use
        #target_size=(img_width, img_height),
        target_size=(64, 64),
        batch_size=32,
        class_mode='binary')

validation_generator = test_datagen.flow_from_directory(
        'demodata/humandata/humanvalidation',
        #when the image perticular size given then use
        #target_size=(img_width, img_height),
        target_size=(64, 64),
        batch_size=32,
        class_mode='binary')

classifier.fit_generator(train_generator,
        #std steps_per_erpoch = 8000
        steps_per_epoch=2000,
        #std epochs = 25
        epochs=10,
        validation_data=validation_generator,
        #std validation steps 2000
        validation_steps=800)
```

```
Found 994 images belonging to 2 classes.
Found 252 images belonging to 2 classes.
Epoch 1/10
2000/2000 [==============================] - 1821s 911ms/step - loss:
0.2758 - acc: 0.8752 - val_loss: 1.1318 - val_acc: 0.7024
Epoch 2/10
2000/2000 [==============================] - 1817s 909ms/step - loss:
0.0337 - acc: 0.9893 - val_loss: 1.7043 - val_acc: 0.7262
Epoch 3/10
2000/2000 [==============================] - 1813s 906ms/step - loss:
0.0148 - acc: 0.9952 - val_loss: 2.3478 - val_acc: 0.7063
Epoch 4/10
2000/2000 [==============================] - 1806s 903ms/step - loss:
0.0243 - acc: 0.9920 - val_loss: 2.1062 - val_acc: 0.6905
Epoch 5/10
2000/2000 [==============================] - 1463s 731ms/step - loss:
0.0077 - acc: 0.9978 - val_loss: 1.9369 - val_acc: 0.7063
Epoch 6/10
2000/2000 [==============================] - 1024s 512ms/step - loss:
0.0153 - acc: 0.9953 - val_loss: 2.1513 - val_acc: 0.7024
Epoch 7/10
2000/2000 [==============================] - 1021s 511ms/step - loss:
```

```
0.0034 - acc: 0.9991 - val_loss: 1.9565 - val_acc: 0.7063
Epoch 8/10
2000/2000 [==============================] - 1025s 513ms/step - loss:
0.0110 - acc: 0.9970 - val_loss: 2.2793 - val_acc: 0.7222
Epoch 9/10
2000/2000 [==============================] - 1019s 510ms/step - loss:
0.0077 - acc: 0.9976 - val_loss: 2.4657 - val_acc: 0.6984
Epoch 10/10
2000/2000 [==============================] - 1023s 511ms/step - loss:
0.0038 - acc: 0.9989 - val_loss: 2.6658 - val_acc: 0.6984
```

Out[9]:

```
<keras.callbacks.History at 0x7f311ad3b6a0>
```

In [36]:

```python
import numpy as np
from keras.preprocessing import image
test_image=image.load_img("/home/rajesh/demodata/humandata/testimg/applauding_0
test_image=image.img_to_array(test_image)
test_image=np.expand_dims(test_image,axis=0)
result=classifier.predict(test_image)
trainig_set=classifier.predict(test_image)
train_generator.class_indices
if result[0][0]==1:
    prediction='Not Human'
    print("Not Human")
else:
    prediction='Human'
    print("Human")
```

```
Human
```

In [38]:

```python
import numpy as np
from keras.preprocessing import image
test_image=image.load_img("/home/rajesh/demodata/humandata/testimg/69020950.jpg
test_image=image.img_to_array(test_image)
test_image=np.expand_dims(test_image,axis=0)
result=classifier.predict(test_image)
trainig_set=classifier.predict(test_image)
train_generator.class_indices
if result[0][0]==1:
    prediction='Not Human'
    print("Not Human")
else:
    prediction='Human'
    print("Human")
```

```
Not Human
```