happy-sad-cnn

August 26, 2024

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import tensorflow as tf
from tensorflow import keras
from keras import layers,Sequential
from keras.layers import Conv2D,MaxPooling2D,Dense,Flatten,BatchNormalization
import os
import cv2 as cv
```

[2]:
```
path=r"C:\Users\RAHUL PATIL\Downloads\archive (5)\happy"
```

[3]:
```
D=os.listdir(path)
```

[4]:
```
data=[]
for i in D:
    A=os.path.join(path,i)
    for j in os.listdir(A):
        B=os.path.join(A,j)
        img=cv.imread(B)
        C=cv.resize(img,(224,224))
        T=D.index(i)
        data.append([C,T])
```

[5]:
```
i=cv.imread(r"C:\Users\RAHUL PATIL\Downloads\archive␣
 ↪(5)\happy\happy\very-happy-people.jpg")
```

[6]:
```
i.shape
```

[6]:
```
(329, 584, 3)
```

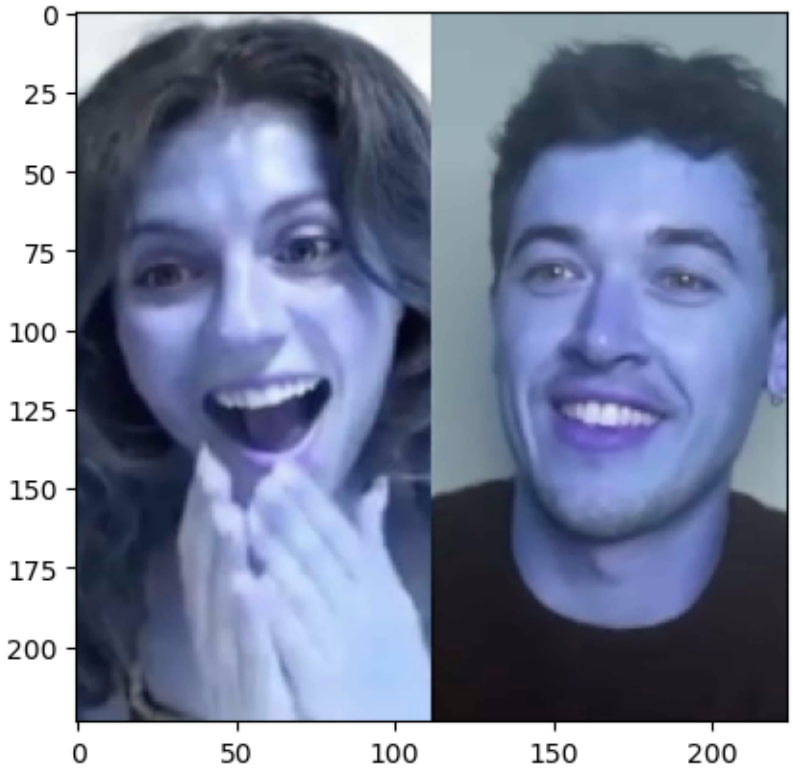
[7]:
```
data[1]
```

[7]:
```
[array([[[235, 238, 243],
         [235, 238, 243],
         [233, 236, 241],
         …,
```

```
          [153, 171, 178],
          [153, 171, 178],
          [153, 171, 178]],

         [[234, 237, 242],
          [234, 237, 242],
          [234, 237, 242],
          …,
          [153, 171, 178],
          [153, 171, 178],
          [153, 171, 178]],

         [[234, 237, 242],
          [234, 237, 242],
          [234, 237, 242],
          …,
          [153, 171, 178],
          [153, 171, 178],
          [153, 171, 178]],

         …,

         [[ 45,  59,  87],
          [ 47,  61,  89],
          [ 58,  67, 101],
          …,
          [ 31,  29,  35],
          [ 31,  29,  35],
          [ 31,  29,  35]],

         [[ 47,  61,  89],
          [ 47,  61,  89],
          [ 57,  71,  99],
          …,
          [ 28,  26,  32],
          [ 28,  26,  32],
          [ 28,  26,  32]],

         [[ 47,  61,  89],
          [ 47,  61,  89],
          [ 57,  71,  99],
          …,
          [ 28,  26,  32],
          [ 28,  26,  32],
          [ 28,  26,  32]]], dtype=uint8),
 0]
```

[8]:
```
plt.imshow(data[1][0])
```

[8]:
```
<matplotlib.image.AxesImage at 0x2b4dd5b5d00>
```



[9]:
```
len(data)
```

[9]:
```
690
```

[10]:
```
len(data[1][0])
```

[10]:
```
224
```

[11]:
```
import random
random.shuffle
```

[11]:
```
<bound method Random.shuffle of <random.Random object at 0x000002B4C93C0A60>>
```

[12]:
```
F=[]
T=[]
for i,j in data:
    F.append(i)
    T.append(j)
```

[13]:
```
len(F)
```

[13]:
```
690
```

[14]:
```
T
```

[14]:
```
[0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
0,
```

```
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0,
 0]
```

[15]:
```python
T1=pd.get_dummies(T).replace({True:1,False:0})
```

```
C:\Users\RAHUL PATIL\AppData\Local\Temp\ipykernel_48428\3959669914.py:1:
FutureWarning: Downcasting behavior in `replace` is deprecated and will be
removed in a future version. To retain the old behavior, explicitly call
`result.infer_objects(copy=False)`. To opt-in to the future behavior, set
`pd.set_option('future.no_silent_downcasting', True)`
  T1=pd.get_dummies(T).replace({True:1,False:0})
```

[16]:
```
T1
```

[16]:
```
     0
0    1
1    1
2    1
3    1
4    1
..  ..
685  1
686  1
687  1
688  1
689  1

[690 rows x 1 columns]
```

[17]:
```
F=np.array(F)
```

[18]:
```
F1=F/255
```

[19]:
```
F1
```

[19]:
```
array([[[[0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         …,
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ]],

        [[0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         …,
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ]],

        [[0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         …,
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ],
         [0.94901961, 0.85882353, 0.6627451 ]],
```

```
 …,

 [[0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  …,
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ]],

 [[0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  …,
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ]],

 [[0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  …,
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ],
  [0.94901961, 0.85882353, 0.6627451 ]]],


[[[0.92156863, 0.93333333, 0.95294118],
  [0.92156863, 0.93333333, 0.95294118],
  [0.91372549, 0.9254902 , 0.94509804],
  …,
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922]],

 [[0.91764706, 0.92941176, 0.94901961],
  [0.91764706, 0.92941176, 0.94901961],
  [0.91764706, 0.92941176, 0.94901961],
  …,
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922]],

 [[0.91764706, 0.92941176, 0.94901961],
  [0.91764706, 0.92941176, 0.94901961],
  [0.91764706, 0.92941176, 0.94901961],
  …,
```

```
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922],
  [0.6       , 0.67058824, 0.69803922]],

 …,

 [[0.17647059, 0.23137255, 0.34117647],
  [0.18431373, 0.23921569, 0.34901961],
  [0.22745098, 0.2627451 , 0.39607843],
  …,
  [0.12156863, 0.11372549, 0.1372549 ],
  [0.12156863, 0.11372549, 0.1372549 ],
  [0.12156863, 0.11372549, 0.1372549 ]],

 [[0.18431373, 0.23921569, 0.34901961],
  [0.18431373, 0.23921569, 0.34901961],
  [0.22352941, 0.27843137, 0.38823529],
  …,
  [0.10980392, 0.10196078, 0.1254902 ],
  [0.10980392, 0.10196078, 0.1254902 ],
  [0.10980392, 0.10196078, 0.1254902 ]],

 [[0.18431373, 0.23921569, 0.34901961],
  [0.18431373, 0.23921569, 0.34901961],
  [0.22352941, 0.27843137, 0.38823529],
  …,
  [0.10980392, 0.10196078, 0.1254902 ],
  [0.10980392, 0.10196078, 0.1254902 ],
  [0.10980392, 0.10196078, 0.1254902 ]]],


[[[0.84705882, 0.84313725, 0.85098039],
  [0.84313725, 0.83921569, 0.84705882],
  [0.84313725, 0.83921569, 0.84705882],
  …,
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647]],

 [[0.84705882, 0.84313725, 0.85098039],
  [0.84313725, 0.83921569, 0.84705882],
  [0.84313725, 0.83921569, 0.84705882],
  …,
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647]],
```

```
 [[0.84705882, 0.84313725, 0.85098039],
  [0.84705882, 0.84313725, 0.85098039],
  [0.84705882, 0.84313725, 0.85098039],
  …,
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647],
  [0.94509804, 0.93333333, 0.94117647]],

 …,

 [[0.6745098 , 0.58039216, 0.4       ],
  [0.65882353, 0.56862745, 0.39215686],
  [0.5372549 , 0.43529412, 0.29019608],
  …,
  [0.76470588, 0.75686275, 0.7254902 ],
  [0.75686275, 0.74901961, 0.71764706],
  [0.75686275, 0.74901961, 0.71764706]],

 [[0.69411765, 0.60392157, 0.42352941],
  [0.70196078, 0.61176471, 0.43529412],
  [0.49803922, 0.4       , 0.25490196],
  …,
  [0.75686275, 0.74901961, 0.71764706],
  [0.75686275, 0.74901961, 0.71764706],
  [0.74901961, 0.74117647, 0.70980392]],

 [[0.6745098 , 0.58431373, 0.40392157],
  [0.69803922, 0.60784314, 0.43137255],
  [0.52156863, 0.41960784, 0.2745098 ],
  …,
  [0.74901961, 0.74117647, 0.70980392],
  [0.74901961, 0.74117647, 0.70980392],
  [0.74901961, 0.74117647, 0.70980392]]],


…,


[[[0.51764706, 0.76862745, 0.74901961],
  [0.56470588, 0.75686275, 0.74901961],
  [0.69019608, 0.81568627, 0.81960784],
  …,
  [0.62352941, 0.59607843, 0.56862745],
  [0.56862745, 0.56078431, 0.52941176],
  [0.38431373, 0.4       , 0.36470588]],

 [[0.46666667, 0.70196078, 0.69019608],
```

```
  [0.54509804, 0.72941176, 0.72941176],
  [0.65098039, 0.77254902, 0.77254902],
  …,
  [0.54509804, 0.5254902 , 0.49803922],
  [0.44705882, 0.45882353, 0.42352941],
  [0.25490196, 0.28627451, 0.24705882]],

 [[0.41960784, 0.62745098, 0.62745098],
  [0.49803922, 0.65882353, 0.66666667],
  [0.61568627, 0.7254902 , 0.72941176],
  …,
  [0.37647059, 0.38823529, 0.35294118],
  [0.2745098 , 0.30196078, 0.26666667],
  [0.15686275, 0.2       , 0.16078431]],

 …,

 [[0.36862745, 0.36862745, 0.36862745],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  …,
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ]],

 [[0.36862745, 0.36862745, 0.36862745],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  …,
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ]],

 [[0.36862745, 0.36862745, 0.36862745],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  …,
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ],
  [1.        , 1.        , 1.        ]]],


[[[0.96078431, 0.95294118, 0.81960784],
  [0.89803922, 0.82745098, 0.50980392],
  [0.8745098 , 0.75686275, 0.46666667],
  …,
  [0.85098039, 0.83529412, 0.82352941],
```

```
  [0.85490196, 0.83921569, 0.82352941],
  [0.85490196, 0.83529412, 0.82352941]],

 [[0.94117647, 0.90588235, 0.70196078],
  [0.90196078, 0.86666667, 0.61568627],
  [0.86666667, 0.76470588, 0.5254902 ],
  …,
  [0.84705882, 0.83529412, 0.81960784],
  [0.85098039, 0.83137255, 0.81960784],
  [0.84705882, 0.83529412, 0.81960784]],

 [[0.45882353, 0.49019608, 0.58431373],
  [0.52941176, 0.5372549 , 0.55686275],
  [0.69019608, 0.58039216, 0.54901961],
  …,
  [0.85098039, 0.83921569, 0.82352941],
  [0.85098039, 0.83137255, 0.81960784],
  [0.84705882, 0.83529412, 0.81960784]],

 …,

 [[0.14901961, 0.14509804, 0.18431373],
  [0.14901961, 0.14901961, 0.17254902],
  [0.15686275, 0.15294118, 0.18431373],
  …,
  [0.04705882, 0.04313725, 0.05098039],
  [0.03921569, 0.03529412, 0.04313725],
  [0.03529412, 0.03529412, 0.03529412]],

 [[0.1372549 , 0.13333333, 0.16862745],
  [0.14509804, 0.14117647, 0.18039216],
  [0.14901961, 0.14509804, 0.19215686],
  …,
  [0.04313725, 0.03921569, 0.04705882],
  [0.04313725, 0.03921569, 0.04705882],
  [0.03921569, 0.03921569, 0.03921569]],

 [[0.16470588, 0.14509804, 0.18823529],
  [0.15686275, 0.16470588, 0.20392157],
  [0.18823529, 0.18823529, 0.23529412],
  …,
  [0.03921569, 0.03921569, 0.03921569],
  [0.03921569, 0.03529412, 0.04313725],
  [0.04313725, 0.03921569, 0.04705882]]],


[[[0.52941176, 0.40784314, 0.70196078],
```

```
 [0.52941176, 0.40784314, 0.70196078],
 [0.5254902 , 0.4       , 0.70196078],
 …,
 [0.50588235, 0.63921569, 0.91372549],
 [0.50588235, 0.63921569, 0.91372549],
 [0.49411765, 0.62352941, 0.90980392]],

[[0.5254902 , 0.40784314, 0.70196078],
 [0.5254902 , 0.40784314, 0.70196078],
 [0.5254902 , 0.40784314, 0.70588235],
 …,
 [0.49803922, 0.63921569, 0.91372549],
 [0.49411765, 0.63529412, 0.90980392],
 [0.49803922, 0.63529412, 0.92156863]],

[[0.51764706, 0.4       , 0.69411765],
 [0.5254902 , 0.41176471, 0.70588235],
 [0.53333333, 0.41568627, 0.71372549],
 …,
 [0.49803922, 0.65098039, 0.92156863],
 [0.49411765, 0.64705882, 0.91764706],
 [0.49803922, 0.64705882, 0.92941176]],

…,

[[0.00392157, 0.00392157, 0.00392157],
 [0.        , 0.        , 0.        ],
 [0.        , 0.        , 0.        ],
 …,
 [0.        , 0.        , 0.        ],
 [0.00392157, 0.00392157, 0.00392157],
 [0.00392157, 0.00392157, 0.00392157]],

[[0.        , 0.        , 0.        ],
 [0.00392157, 0.00392157, 0.00392157],
 [0.00392157, 0.00392157, 0.00392157],
 …,
 [0.00392157, 0.00392157, 0.00392157],
 [0.00392157, 0.00392157, 0.00392157],
 [0.        , 0.        , 0.        ]],

[[0.        , 0.        , 0.        ],
 [0.00392157, 0.00392157, 0.00392157],
 [0.00392157, 0.00392157, 0.00392157],
 …,
 [0.00392157, 0.00392157, 0.00392157],
 [0.00392157, 0.00392157, 0.00392157],
```

```
       [0.        , 0.        , 0.        ]]]])
```

[20]:
```
F1.shape
```

[20]:
```
(690, 224, 224, 3)
```

[21]:
```
T1.shape
```

[21]:
```
(690, 1)
```

[22]:
```
T=np.array(T)
```

[23]:
```
model=Sequential()

model.add(Conv2D(130,(5,5),activation='relu'))
model.add(MaxPooling2D((2,2),strides=(1,1)))

model.add(Conv2D(80,(4,4),activation='relu'))
model.add(MaxPooling2D((2,2),strides=(2,2)))

model.add(Flatten())

model.add(Dense(70,input_shape=(224,224,3),activation='relu'))
model.add(Dense(50,activation='relu'))
model.add(Dense(1,activation='sigmoid'))

model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])
```

```
C:\Users\RAHUL PATIL\AppData\Roaming\Python\Python312\site-
packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an
`input_shape`/`input_dim` argument to a layer. When using Sequential models,
prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

[24]:
```
from keras.layers import Input,Lambda,Dense,Flatten
from keras.models import Model
from keras.applications.vgg16 import VGG16
from keras.applications.vgg16 import preprocess_input
from keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from glob import glob
```

[25]:
```
vgg=VGG16(input_shape=[224,224,3],
          weights='imagenet',
```

```
        include_top=False)
```

[26]:
```
for i in vgg.layers:
    i.trainable=False
```

[27]:
```
x=Flatten()(vgg.output)
```

[28]:
```
y=Dense(1,activation='sigmoid')(x)
```

[29]:
```
model=keras.Model(vgg.input,y)
```

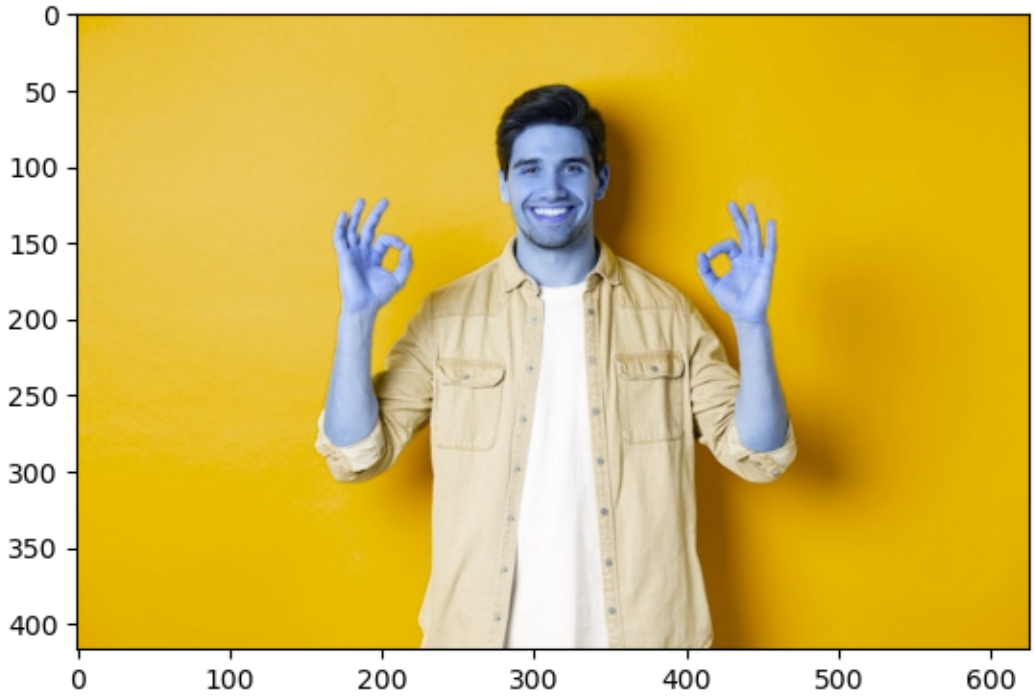
[30]:
```
model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])
```

[31]:
```
model.fit(F1,T,epochs=5,validation_split=0.15,batch_size=25)
```

```
Epoch 1/5
24/24              252s 10s/step -
accuracy: 0.9756 - loss: 0.0802 - val_accuracy: 1.0000 - val_loss: 1.1878e-09
Epoch 2/5
24/24              245s 10s/step -
accuracy: 1.0000 - loss: 1.2239e-09 - val_accuracy: 1.0000 - val_loss:
2.8015e-10
Epoch 3/5
24/24              247s 10s/step -
accuracy: 1.0000 - loss: 4.8190e-10 - val_accuracy: 1.0000 - val_loss:
2.4172e-10
Epoch 4/5
24/24              242s 10s/step -
accuracy: 1.0000 - loss: 2.5040e-10 - val_accuracy: 1.0000 - val_loss:
2.3833e-10
Epoch 5/5
24/24              242s 10s/step -
accuracy: 1.0000 - loss: 4.5476e-10 - val_accuracy: 1.0000 - val_loss:
2.3802e-10
```

[31]:
```
<keras.src.callbacks.history.History at 0x2b4f385b200>
```

[53]:
```
img_path=r"C:\Users\RAHUL PATIL\Downloads\archive␣
 ↪(5)\happy\happy\s-looking-pleased-like-something-standing-against-blue-background_1258-6505
 ↪jpg"
img=cv.imread(img_path)
img_resized=cv.resize(img,(224,224))
img_resized=img_resized.reshape(1,224,224,3)
plt.imshow(img)
plt.show()
```



[54]:
```
prediction=model.predict(img_resized)
prediction
```

```
1/1             0s 475ms/step
```

[54]:
```
array([[0.]], dtype=float32)
```

[56]:
```
if prediction[0][0]<=0.5:
    print('model:happy')
else:
    print('model:sad')
```

```
model:happy
```

[ ]: