In [4]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
import tensorflow as tf
import numpy as np
import cv2
import os
```

```
WARNING:tensorflow:From C:\Users\arnak\anaconda3\Lib\site-packages\keras\src
\losses.py:2976: The name tf.losses.sparse_softmax_cross_entropy is deprecate
d. Please use tf.compat.v1.losses.sparse_softmax_cross_entropy instead.
```

In [5]:
```
import warnings
warnings.filterwarnings("ignore")
```

In [6]:
```
img=image.load_img(r"D:\NIT\DATASCIENCE\ARNAK TASK\CNN\tra\sad\ima 2.jpg")
```

In [7]:
```
plt.imshow(img)
```

Out[7]:
```
<matplotlib.image.AxesImage at 0x215d086bd90>
```

In [8]:
```
i1 = cv2.imread(r"D:\NIT\DATASCIENCE\ARNAK TASK\CNN\tra\sad\ima 2.jpg")
i1
```

Out[8]:
```
array([[[206, 207, 211],
        [206, 207, 211],
        [206, 207, 211],
        ...,
        [207, 207, 213],
        [207, 207, 213],
        [206, 206, 212]],

       [[206, 207, 211],
        [206, 207, 211],
        [206, 207, 211],
        ...,
        [207, 207, 213],
        [207, 207, 213],
        [206, 206, 212]],

       [[206, 207, 211],
        [206, 207, 211],
        [206, 207, 211],
        ...,
        [207, 207, 213],
        [207, 207, 213],
        [206, 206, 212]],

       ...,

       [[195, 195, 201],
        [195, 195, 201],
        [195, 195, 201],
        ...,
        [ 96, 181, 129],
        [ 94, 179, 127],
        [ 91, 176, 124]],

       [[195, 195, 201],
        [195, 195, 201],
        [195, 195, 201],
        ...,
        [ 96, 181, 129],
        [ 94, 179, 127],
        [ 91, 176, 124]],

       [[195, 195, 201],
        [195, 195, 201],
        [195, 195, 201],
        ...,
        [ 96, 181, 129],
        [ 94, 179, 127],
        [ 91, 176, 124]]], dtype=uint8)
```

In [9]:
```
i1.shape
```

Out[9]:
```
(275, 183, 3)
```

In [10]:
```
train = ImageDataGenerator(rescale = 1/255)
validataion = ImageDataGenerator(rescale = 1/255)
```

In [11]:
```
train_dataset = train.flow_from_directory(r"D:\NIT\DATASCIENCE\ARNAK TASK\CNN\
                                          target_size = (200,200),
                                          batch_size = 3,
                                          class_mode = 'binary')
```

```
Found 45 images belonging to 2 classes.
```

In [12]:
```
validataion_dataset = validataion.flow_from_directory(r"D:\NIT\DATASCIENCE\ARN
                                           target_size = (200,200),
                                           batch_size = 3,
                                           class_mode = 'binary')
```

```
Found 0 images belonging to 2 classes.
```

In [13]:
```
train_dataset.class_indices
```

Out[13]:
```
{'happy': 0, 'sad': 1}
```

In [14]:
```
train_dataset.classes
```

Out[14]:
```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1])
```

In [15]:
```
model = tf.keras.models.Sequential([ tf.keras.layers.Conv2D(16,(3,3),activatio
                                     tf.keras.layers.MaxPool2D(2,2), #3 filtr w
                                     #
                                     tf.keras.layers.Conv2D(32,(3,3),activation
                                     tf.keras.layers.MaxPool2D(2,2),
                                     #
                                     tf.keras.layers.Conv2D(64,(3,3),activation
                                     tf.keras.layers.MaxPool2D(2,2),
                                     ##
                                     tf.keras.layers.Flatten(),
                                     ##
                                     tf.keras.layers.Dense(512, activation = 'r
                                     #
                                     tf.keras.layers.Dense(1,activation= 'sigmo
                                     ]
                                     )
```

```
WARNING:tensorflow:From C:\Users\arnak\anaconda3\Lib\site-packages\keras\src
\backend.py:873: The name tf.get_default_graph is deprecated. Please use tf.c
ompat.v1.get_default_graph instead.

WARNING:tensorflow:From C:\Users\arnak\anaconda3\Lib\site-packages\keras\src
\layers\pooling\max_pooling2d.py:161: The name tf.nn.max_pool is deprecated.
Please use tf.nn.max_pool2d instead.
```

In [16]:
```
model.compile(loss='binary_crossentropy',
              optimizer = tf.keras.optimizers.RMSprop(lr = 0.001),
              metrics = ['accuracy']
              )
```

```
WARNING:absl:`lr` is deprecated in Keras optimizer, please use `learning_rate
` or use the legacy optimizer, e.g.,tf.keras.optimizers.legacy.RMSprop.
```

In [63]:
```
model_fit = model.fit(train_dataset,
                      steps_per_epoch = 3,
                      epochs = 25,
                      validation_data = validataion_dataset)
```

```
Epoch 1/25
3/3 [==============================] - 1s 160ms/step - loss: 0.3501 - accu
racy: 0.8889
Epoch 2/25
3/3 [==============================] - 1s 175ms/step - loss: 0.4015 - accu
racy: 1.0000
Epoch 3/25
3/3 [==============================] - 0s 158ms/step - loss: 0.1786 - accu
racy: 1.0000
Epoch 4/25
3/3 [==============================] - 0s 153ms/step - loss: 0.6451 - accu
racy: 0.6667
Epoch 5/25
3/3 [==============================] - 0s 157ms/step - loss: 0.6064 - accu
racy: 0.6667
Epoch 6/25
3/3 [==============================] - 0s 158ms/step - loss: 0.7065 - accu
racy: 0.5556
Epoch 7/25
```

In [40]:
```
dir_path = r'D:\NIT\DATASCIENCE\ARNAK TASK\CNN\test'
for i in os.listdir(dir_path ):
    print(i)
```

```
12.jpg
23.jpg
24.jpg
25.jpg
26.jpg
27.jpg
28.jpg
29.jpg
30.jpg
31.jpg
32.jpg
33.jpg
34.jpg
35.jpg
36.jpg
37.jpg
38.jpg
39.jpg
40.jpg
41.webp
7.jpg
8.webp
9.jpg
anime_boy_be_happy_by_alarooksalah_df2wqkr-pre.png
download (1).jpg
download (2).jpg
download.jpg
ima 2.jpg
ima 3.jpg
ima 4.jpg
ima 5.jpg
images (1).jpg
images (10).jpg
images (11).jpg
images (12).jpg
images (13).jpg
images (14).jpg
images (15).jpg
images (2).jpg
images (3).jpg
images (4).jpg
images (5).jpg
images (6).jpg
images (7).jpg
images (8).jpg
images (9).jpg
images.jpg
sad-man-crying-young-desperate-expression-37013812.webp
```

In [41]:

```
dir_path = r'D:\NIT\DATASCIENCE\ARNAK TASK\CNN\test'


for filename in os.listdir(dir_path):
    img_path = os.path.join(dir_path, filename)
    img = image.load_img(img_path, target_size=(200, 200))
    plt.imshow(img)
    plt.axis('off')  # Disable axis
    plt.show()
```

In [42]:

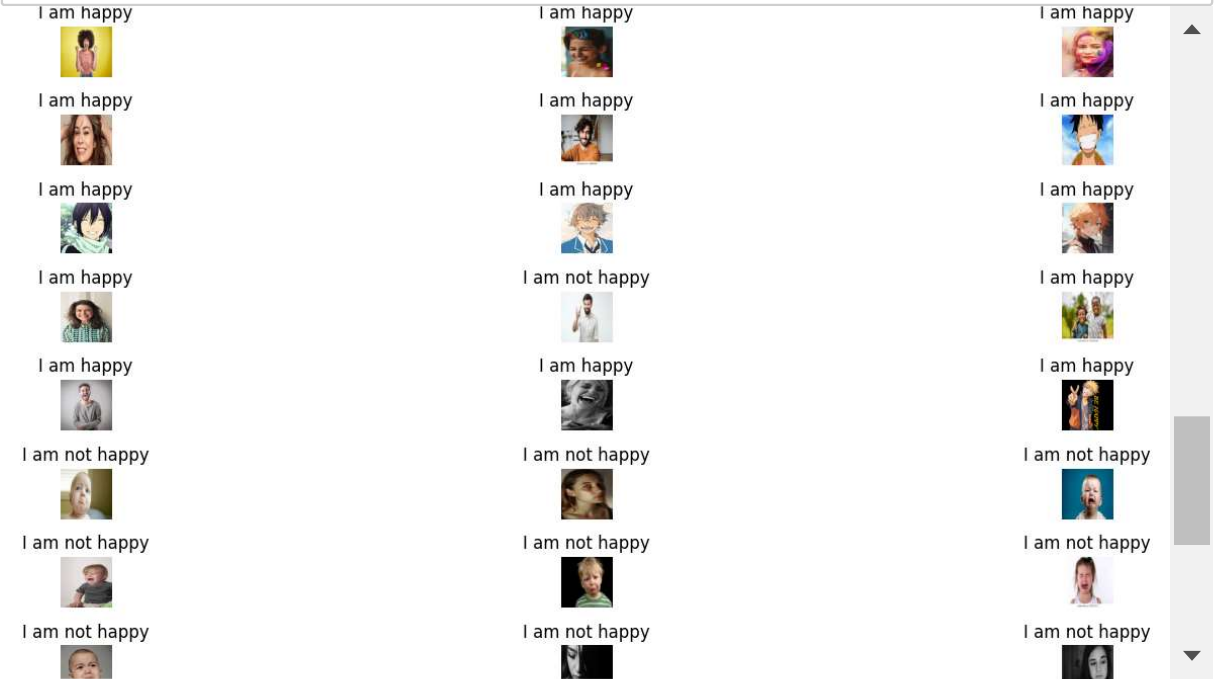
```
dir_path = r'D:\NIT\DATASCIENCE\ARNAK TASK\CNN\test'


image_filenames = os.listdir(dir_path)

num_images = len(image_filenames)
fig, axes = plt.subplots(1, num_images, figsize=(20, 5))


for i, filename in enumerate(image_filenames):
    img_path = os.path.join(dir_path, filename)
    img = image.load_img(img_path, target_size=(200, 200))
    axes[i].imshow(img)
    axes[i].axis('off')  # Disable axis
    axes[i].set_title(filename)

plt.tight_layout()
plt.show()
```

In [44]:

```
dir_path = r'D:\NIT\DATASCIENCE\ARNAK TASK\CNN\test'

for i in os.listdir(dir_path ):
    img = image.load_img(dir_path+ '//'+i, target_size = (200,200))
    plt.imshow(img)
    plt.show()

    x= image.img_to_array(img)
    x=np.expand_dims(x,axis = 0)
    images = np.vstack([x])

    val = model.predict(images)
    if val == 0:
        print( ' i am  happy')
    else:
        print('i am not happy')
```

In [45]:

```
dir_path = r'D:\NIT\DATASCIENCE\ARNAK TASK\CNN\test'


plt.figure(figsize=(15, 15))
columns = 3
rows = len(os.listdir(dir_path)) // columns + 1

for i, filename in enumerate(os.listdir(dir_path)):
    img_path = os.path.join(dir_path, filename)
    img = image.load_img(img_path, target_size=(200, 200))
    plt.subplot(rows, columns, i + 1)
    plt.imshow(img)
    plt.axis('off')  # Disable axis

    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)
    images = np.vstack([x])

    val = model.predict(images)
    if val == 0:
        prediction = 'I am happy'
    else:
        prediction = 'I am not happy'
    plt.title(prediction)

plt.tight_layout()
plt.show()
```



In [ ]: