```
[2] import tensorflow as tf
    from tensorflow import keras
    from tensorflow.keras import layers
    from tensorflow.keras.preprocessing.image import ImageDataGenerator

    IMG_SIZE = 244
    BATCH_SIZE = 32
```

```
[3] train_datagen = ImageDataGenerator(rescale=1./255,validation_split=0.2)
    train_generator = train_datagen.flow_from_directory(
        '/content/drive/MyDrive/face',
        target_size=(IMG_SIZE,IMG_SIZE),
        batch_size=BATCH_SIZE,
        class_mode='binary',
        subset='training'
    )
    val_generator = train_datagen.flow_from_directory(
        '/content/drive/MyDrive/face',
        target_size=(IMG_SIZE,IMG_SIZE),
        batch_size=BATCH_SIZE,
        class_mode='binary',
        subset='validation'
    )
```

```
Found 144 images belonging to 2 classes.
Found 35 images belonging to 2 classes.
```

```
[4] model = keras.Sequential([
        layers.Conv2D(32, (3, 3), activation='relu', input_shape=(IMG_SIZE, IMG_SIZE, 3)),
        layers.MaxPooling2D((2, 2)),
        layers.Conv2D(64, (3, 3), activation='relu'),
        layers.MaxPooling2D((2, 2)),
        layers.Conv2D(128, (3, 3), activation='relu'),
        layers.MaxPooling2D((2, 2)),
        layers.Flatten(),
        layers.Dense(128, activation='relu'),
        layers.Dense(1, activation='sigmoid')
    ])
```

Loading...

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

```
[7] model.fit(train_generator, epochs=5, validation_data=val_generator)
```

```
Epoch 1/5
5/5 [==============================] - 23s 4s/step - loss: 0.1470 - accuracy: 1.0000 - val_loss: 2.4810e-25 - val_accuracy: 1.0000
Epoch 2/5
5/5 [==============================] - 19s 4s/step - loss: 1.7294e-25 - accuracy: 1.0000 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 3/5
5/5 [==============================] - 19s 4s/step - loss: 0.0000e+00 - accuracy: 1.0000 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 4/5
5/5 [==============================] - 20s 4s/step - loss: 0.0000e+00 - accuracy: 1.0000 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
Epoch 5/5
5/5 [==============================] - 19s 4s/step - loss: 0.0000e+00 - accuracy: 1.0000 - val_loss: 0.0000e+00 - val_accuracy: 1.0000
<keras.src.callbacks.History at 0x7c9a09662560>
```

```
[8] model.save("face.h5", "label.text")
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as an HDF5 file via `model.s
  saving_api.save_model(
```

```python
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np

model = load_model('/content/drive/MyDrive/face/Model.h5')

test_image_path = '/content/drive/MyDrive/face/face_detection/th100.jpg'

# Change target_size to (244, 244) to match the model's input shape
img =image.load_img(test_image_path,target_size=(244,244))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array,axis=0)

img_array /=225.0

prediction = model.predict(img_array)

print(prediction)
```

```
1/1 [==============================] - 0s 130ms/step
[[1.]]
```

```python
if prediction < 0.5:
   print("Prediction: This is a female(Probability:", prediction[0][0])

else:
   print("Prediction: This is a male (Probability:", prediction[0][0])
```

```
Prediction: This is a male (Probability: 1.0
```

th102.jpg