

```

import tensorflow as tf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
import cv2
from tensorflow.keras.preprocessing.image import ImageDataGenerator

src_path = "/Users/nageshjadhav/Desktop/Face_dataset/train/"
sub_class = os.listdir(src_path)
print(sub_class)

['Nagesh_Jadhav', '.DS_Store', 'Slate']

src_path_train = "/Users/nageshjadhav/Desktop/Face_dataset/train/"
src_path_test = "/Users/nageshjadhav/Desktop/Face_dataset/test/"
src_path_valid = "/Users/nageshjadhav/Desktop/Face_dataset/valid/"

train_datagen = ImageDataGenerator(
    rescale=1 / 255.0,
    rotation_range=20,
    zoom_range=0.05,
    width_shift_range=0.05,
    height_shift_range=0.05,
    shear_range=0.05,
    horizontal_flip=True,
    fill_mode="nearest",
    validation_split=0.20)

valid_datagen = ImageDataGenerator(
    rescale=1 / 255.0,
    rotation_range=20,
    zoom_range=0.05,
    width_shift_range=0.05,
    height_shift_range=0.05,
    shear_range=0.05,
    horizontal_flip=True,
    fill_mode="nearest",
    validation_split=0.20)

test_datagen = ImageDataGenerator(rescale=1 / 255.0)

batch_size = 4
train_generator = train_datagen.flow_from_directory(
    directory=src_path_train,
    target_size=(100, 100),
    color_mode="rgb",
    batch_size=batch_size,
    class_mode="categorical",

```

```

        subset='training',
        shuffle=True,
        seed=42
    )
    valid_generator = valid_datagen.flow_from_directory(
        directory=src_path_valid,
        target_size=(100, 100),
        color_mode="rgb",
        batch_size=batch_size,
        class_mode="categorical",
        subset='validation',
        shuffle=True,
        seed=42
    )
    test_generator = test_datagen.flow_from_directory(
        directory=src_path_test,
        target_size=(100, 100),
        color_mode="rgb",
        batch_size=1,
        class_mode=None,
        shuffle=False,
        seed=42
    )

```

Found 133 images belonging to 2 classes.
 Found 4 images belonging to 2 classes.
 Found 14 images belonging to 1 classes.

```

model = tf.keras.models.Sequential()

model.add(tf.keras.layers.Conv2D(32, (3, 3), activation='relu',
input_shape=(100, 100, 3)))
model.add(tf.keras.layers.MaxPooling2D(pool_size=(2, 2)))

model.add(tf.keras.layers.Conv2D(16, (3, 3), activation='relu'))
model.add(tf.keras.layers.MaxPooling2D(pool_size=(2, 2)))

model.add(tf.keras.layers.Flatten())
model.add(tf.keras.layers.Dense(8, activation='relu'))
model.add(tf.keras.layers.Dense(2, activation='sigmoid'))

model.compile(loss="binary_crossentropy", optimizer="adam", metrics=['ac
curacy'])

model.fit_generator(train_generator,
                    validation_data = valid_generator,
                    steps_per_epoch =
train_generator.n//train_generator.batch_size,
                    validation_steps =

```

```
valid_generator.n//valid_generator.batch_size,  
epochs=5)
```

Epoch 1/5

```
/var/folders/yh/sv7lkgq112103y8hwl95rv1r0000gn/T/  
ipykernel_2926/1321084248.py:1: UserWarning: `Model.fit_generator` is  
deprecated and will be removed in a future version. Please use  
`Model.fit`, which supports generators.  
model.fit_generator(train_generator,
```

```
33/33 [=====] - 1s 23ms/step - loss: 0.5949 -  
accuracy: 0.7209 - val_loss: 0.4479 - val_accuracy: 1.0000
```

Epoch 2/5

```
33/33 [=====] - 0s 15ms/step - loss: 0.1656 -  
accuracy: 0.9845 - val_loss: 0.0262 - val_accuracy: 1.0000
```

Epoch 3/5

```
33/33 [=====] - 1s 16ms/step - loss: 0.0288 -  
accuracy: 1.0000 - val_loss: 0.0747 - val_accuracy: 1.0000
```

Epoch 4/5

```
33/33 [=====] - 1s 15ms/step - loss: 0.0059 -  
accuracy: 1.0000 - val_loss: 0.0118 - val_accuracy: 1.0000
```

Epoch 5/5

```
33/33 [=====] - 0s 15ms/step - loss: 9.9139e-  
04 - accuracy: 1.0000 - val_loss: 0.1347 - val_accuracy: 1.0000
```

<keras.callbacks.History at 0x2bdb7a160>

```
score = model.evaluate_generator(valid_generator)  
print('Test loss:', score[0])  
print('Test accuracy:', score[1])
```

Test loss: 0.006777426227927208

Test accuracy: 1.0

```
/var/folders/yh/sv7lkgq112103y8hwl95rv1r0000gn/T/  
ipykernel_2926/1909056500.py:1: UserWarning:  
`Model.evaluate_generator` is deprecated and will be removed in a  
future version. Please use `Model.evaluate`, which supports  
generators.
```

```
score = model.evaluate_generator(valid_generator)
```

```
predict=model.predict_generator(test_generator)  
# predict the class label  
y_classes = predict.argmax(axis=-1)
```

```
/var/folders/yh/sv7lkgq112103y8hwl95rv1r0000gn/T/  
ipykernel_2926/3467329514.py:1: UserWarning: `Model.predict_generator`  
is deprecated and will be removed in a future version. Please use  
`Model.predict`, which supports generators.  
predict=model.predict_generator(test_generator)
```

```

print(y_classes)

[0 1 1 1 1 1 1 1 1 0 0 0 0 0]

labels = (train_generator.class_indices)
print(labels)
labels = dict((v,k) for k,v in labels.items())
print(labels)
predictions = [labels[k] for k in y_classes]
print(predictions)

{'Nagesh_Jadhav': 0, 'Slate': 1}
{0: 'Nagesh_Jadhav', 1: 'Slate'}
['Nagesh_Jadhav', 'Slate', 'Slate', 'Slate', 'Slate', 'Slate',
'Slate', 'Slate', 'Slate', 'Nagesh_Jadhav', 'Nagesh_Jadhav',
'Nagesh_Jadhav', 'Nagesh_Jadhav', 'Nagesh_Jadhav']

filenames=test_generator.filenames
results=pd.DataFrame({"Filename":filenames,
                      "Predictions":predictions})
results.head(20)

```

	Filename	Predictions
0	predict/100.jpg	Nagesh_Jadhav
1	predict/82.jpg	Slate
2	predict/83.jpg	Slate
3	predict/84.jpg	Slate
4	predict/85.jpg	Slate
5	predict/86.jpg	Slate
6	predict/87.jpg	Slate
7	predict/88.jpg	Slate
8	predict/94.jpg	Slate
9	predict/95.jpg	Nagesh_Jadhav
10	predict/96.jpg	Nagesh_Jadhav
11	predict/97.jpg	Nagesh_Jadhav
12	predict/98.jpg	Nagesh_Jadhav
13	predict/99.jpg	Nagesh_Jadhav