realvsfakefin2-2

May 17, 2024

[ ]:
```
# /content/drive/MyDrive/realvsfakeus
```

[ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

[ ]:
```
!pip install tensorflow
```

```
Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-
packages (2.15.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-
packages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (24.3.25)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-
packages (from tensorflow) (3.9.0)
Requirement already satisfied: libclang>=13.0.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (18.1.1)
Requirement already satisfied: ml-dtypes~=0.2.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: numpy<2.0.0,>=1.23.5 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (1.25.2)
Requirement already satisfied: opt-einsum>=2.3.2 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-
packages (from tensorflow) (24.0)
Requirement already satisfied:
protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<5.0.0dev,>=3.20.3
in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.20.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-
```

```
packages (from tensorflow) (67.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-
packages (from tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (2.4.0)
Requirement already satisfied: typing-extensions>=3.6.6 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (4.11.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (0.37.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (1.63.0)
Requirement already satisfied: tensorboard<2.16,>=2.15 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.2)
Requirement already satisfied: tensorflow-estimator<2.16,>=2.15.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: keras<2.16,>=2.15.0 in
/usr/local/lib/python3.10/dist-packages (from tensorflow) (2.15.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in
/usr/local/lib/python3.10/dist-packages (from astunparse>=1.6.0->tensorflow)
(0.43.0)
Requirement already satisfied: google-auth<3,>=1.6.3 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (2.27.0)
Requirement already satisfied: google-auth-oauthlib<2,>=0.5 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (1.2.0)
Requirement already satisfied: markdown>=2.6.8 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (3.6)
Requirement already satisfied: requests<3,>=2.21.0 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (2.31.0)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from
tensorboard<2.16,>=2.15->tensorflow) (3.0.3)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from google-
auth<3,>=1.6.3->tensorboard<2.16,>=2.15->tensorflow) (5.3.3)
Requirement already satisfied: pyasn1-modules>=0.2.1 in
/usr/local/lib/python3.10/dist-packages (from google-
auth<3,>=1.6.3->tensorboard<2.16,>=2.15->tensorflow) (0.4.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-
packages (from google-auth<3,>=1.6.3->tensorboard<2.16,>=2.15->tensorflow) (4.9)
```

```
Requirement already satisfied: requests-oauthlib>=0.7.0 in
/usr/local/lib/python3.10/dist-packages (from google-auth-
oauthlib<2,>=0.5->tensorboard<2.16,>=2.15->tensorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3,>=2.21.0->tensorboard<2.16,>=2.15->tensorflow) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3,>=2.21.0->tensorboard<2.16,>=2.15->tensorflow) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3,>=2.21.0->tensorboard<2.16,>=2.15->tensorflow) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
requests<3,>=2.21.0->tensorboard<2.16,>=2.15->tensorflow) (2024.2.2)
Requirement already satisfied: MarkupSafe>=2.1.1 in
/usr/local/lib/python3.10/dist-packages (from
werkzeug>=1.0.1->tensorboard<2.16,>=2.15->tensorflow) (2.1.5)
Requirement already satisfied: pyasn1<0.7.0,>=0.4.6 in
/usr/local/lib/python3.10/dist-packages (from pyasn1-modules>=0.2.1->google-
auth<3,>=1.6.3->tensorboard<2.16,>=2.15->tensorflow) (0.6.0)
Requirement already satisfied: oauthlib>=3.0.0 in
/usr/local/lib/python3.10/dist-packages (from requests-oauthlib>=0.7.0->google-
auth-oauthlib<2,>=0.5->tensorboard<2.16,>=2.15->tensorflow) (3.2.2)
```

[ ]:
```
import tensorflow as tf
from tensorflow.keras import models, layers
import matplotlib.pyplot as plt
```

[ ]:
```
# setting the image fixed size for training and intializing the batch size,␣
 ↪channel and number of epochs
Image_Size= 256
Batch_Size = 32
Channels=3
Epochs=55
```

[ ]:
```
# determining number of pics and classes
imgdata = tf.keras.preprocessing.image_dataset_from_directory(
    "/content/drive/MyDrive/realvsfakeus",
    shuffle=True,
    image_size = (Image_Size,Image_Size),
    batch_size=Batch_Size

)
```

```
Found 2041 files belonging to 2 classes.
```

[ ]:
```
# above code creates
# This code creates a dataset of images using the Keras utility function␣
 ↪image_dataset_from_directory.
# It reads images from a specified directory and organizes them into batches␣
 ↪for training or validation.
# Here's what each parameter means:
# This is the path to the directory containing your image data.
# The function will look for subdirectories within this path, where each␣
 ↪subdirectory
# corresponds to a different class or label (e.g., "real" and "fake" faces).
# Images in each subdirectory will be treated as examples of that class.
```

[ ]:
```
class_names = imgdata.class_names
class_names  # 0 means fake and 1 means real
```

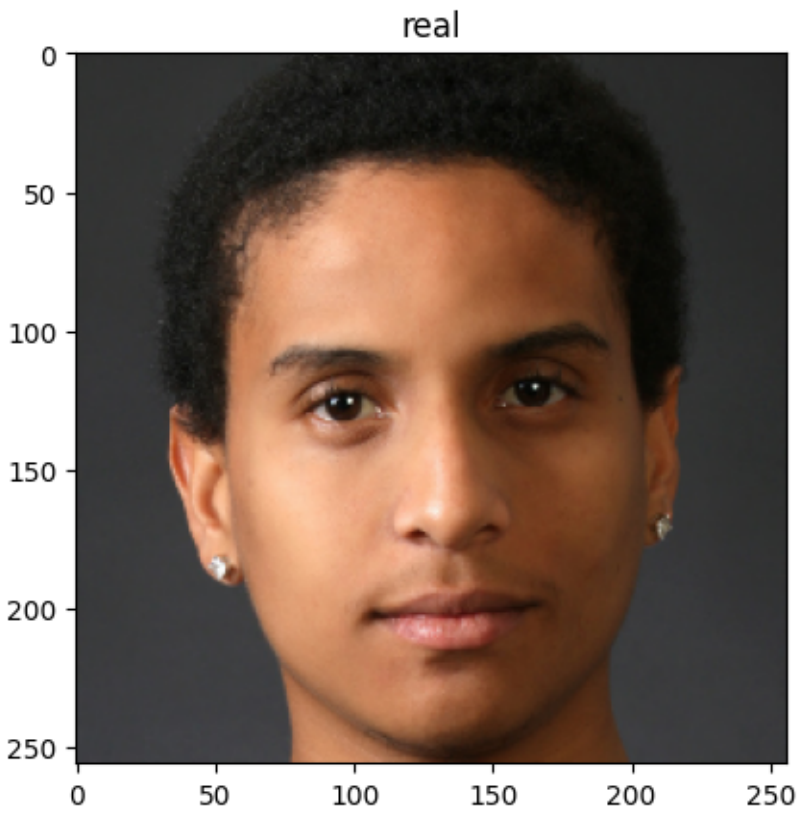
[ ]:
```
['fake', 'real']
```

[ ]:
```
for image_batch, label_batch in imgdata.take(1):
    print(image_batch.shape)
    print(label_batch.numpy())
```

```
(32, 256, 256, 3)
[1 0 0 1 0 0 1 1 1 0 0 1 1 0 1 1 0 1 0 0 1 0 0 0 1 0 0 1 0 1 0 1]
```

[ ]:
```
# The above code iterates over the first batch of data from the dataset.
# The take(1) method ensures that only one batch is retrieved.
# Each batch contains a set of images and their corresponding labels.
# The .numpy() method converts the labels from TensorFlow tensors to NumPy
# arrays for easier printing.
# The labels correspond to the class names ("fake" or "real") associated with␣
 ↪each image
```

[ ]:
```
# Displaying our first image
for image_batch, label_batch in imgdata.take(1):
    plt.imshow(image_batch[0].numpy().astype("uint8"))
    plt.title(class_names[label_batch[0]])
```



[ ]:
```
# function to create dataset
```

[ ]:
```
def splitting_dataset_tf(ds, train_split=0.8, val_split=0.1, test_split=0.1,␣
↪shuffle=True, shuffle_size=10000):

    ds_size=len(ds)

    if shuffle:
        ds = ds.shuffle(shuffle_size, seed=12)

    train_size=int(train_split * ds_size)
    val_size= int(val_split * ds_size)

    train_ds= ds.take(train_size)

    val_ds = ds.skip(train_size).take(val_size)
    test_ds = ds.skip(train_size).skip(val_size)

    return train_ds, val_ds, test_ds
```

[ ]:
```
train_ds, val_ds, test_ds=splitting_dataset_tf(imgdata)
```

[ ]:
```
print(len(train_ds),len(val_ds),len(test_ds))
```

```
51 6 7
```

[ ]:
```
#  Pipeline for Optimization for Training, Validation, and Testing Datasets
train_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
val_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
test_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
```

[ ]:
```
# Image Preprocessing for Resizing and Rescaling
resize_and_rescale = tf.keras.Sequential([
    layers.experimental.preprocessing.Resizing(Image_Size,Image_Size),
    layers.experimental.preprocessing.Rescaling(1.0/255)
])
```

[ ]:
```
# Data augmentation
data_aug = tf.keras.Sequential([
    layers.experimental.preprocessing.RandomFlip("horizontal_and_vertical"),
    layers.experimental.preprocessing.RandomRotation(0.2),

])
```

[ ]:
```
# Architecture of the model
input_shape = (Batch_Size,Image_Size, Image_Size,Channels)
n_classes = 3

model = models.Sequential([
    resize_and_rescale,
    data_aug,
    layers.Conv2D(32, (3,3), activation='relu', input_shape = input_shape),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64, kernel_size = (3,3), activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64, kernel_size = (3,3), activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64, (3,3), activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64, (3,3), activation='relu'),
    layers.MaxPooling2D((2,2)),
    layers.Conv2D(64, (3,3), activation='relu'),
    layers.MaxPooling2D((2,2)),


    layers.Flatten(),
    layers.Dense(64, activation = 'relu'),
    layers.Dense(n_classes, activation= 'softmax'),
```

```
])

model.build(input_shape=input_shape)
```

[ ]:
```
# Compiling the model with loss function and optimizer
model.compile(
    optimizer='adam',
    loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False),
    metrics=['accuracy']
)
```

[ ]:
```
# Model Training with Training and Validation Data
history = model.fit(
    train_ds,
    epochs=275,
    batch_size=Batch_Size,
    verbose=1,
    validation_data=val_ds
)
```

```
Epoch 1/275
51/51 [==============================] - 238s 80ms/step - loss: 0.7425 -
accuracy: 0.4798 - val_loss: 0.7070 - val_accuracy: 0.4657
Epoch 2/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6985 - accuracy:
0.5159 - val_loss: 0.7510 - val_accuracy: 0.5343
Epoch 3/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6983 - accuracy:
0.5190 - val_loss: 0.7012 - val_accuracy: 0.5343
Epoch 4/275
51/51 [==============================] - 4s 76ms/step - loss: 0.7018 - accuracy:
0.5172 - val_loss: 0.6968 - val_accuracy: 0.4675
Epoch 5/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6987 - accuracy:
0.4933 - val_loss: 0.7091 - val_accuracy: 0.4657
Epoch 6/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6957 - accuracy:
0.5263 - val_loss: 0.6924 - val_accuracy: 0.5343
Epoch 7/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6963 - accuracy:
0.5031 - val_loss: 0.6940 - val_accuracy: 0.5343
Epoch 8/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6932 - accuracy:
0.5214 - val_loss: 0.6887 - val_accuracy: 0.5392
Epoch 9/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6938 - accuracy:
0.5214 - val_loss: 0.6961 - val_accuracy: 0.4730
```

```
Epoch 10/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6900 - accuracy:
0.5478 - val_loss: 0.6865 - val_accuracy: 0.5466
Epoch 11/275
51/51 [==============================] - 4s 85ms/step - loss: 0.6919 - accuracy:
0.5343 - val_loss: 0.6848 - val_accuracy: 0.5784
Epoch 12/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6870 - accuracy:
0.5625 - val_loss: 0.6839 - val_accuracy: 0.5545
Epoch 13/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6903 - accuracy:
0.5227 - val_loss: 0.6838 - val_accuracy: 0.5435
Epoch 14/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6910 - accuracy:
0.5484 - val_loss: 0.6788 - val_accuracy: 0.5888
Epoch 15/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6875 - accuracy:
0.5398 - val_loss: 0.6935 - val_accuracy: 0.5343
Epoch 16/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6907 - accuracy:
0.5263 - val_loss: 0.6823 - val_accuracy: 0.5570
Epoch 17/275
51/51 [==============================] - 4s 82ms/step - loss: 0.6834 - accuracy:
0.5797 - val_loss: 0.6831 - val_accuracy: 0.5496
Epoch 18/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6873 - accuracy:
0.5472 - val_loss: 0.6786 - val_accuracy: 0.5705
Epoch 19/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6857 - accuracy:
0.5650 - val_loss: 0.6756 - val_accuracy: 0.5754
Epoch 20/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6810 - accuracy:
0.5754 - val_loss: 0.6776 - val_accuracy: 0.5692
Epoch 21/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6798 - accuracy:
0.5852 - val_loss: 0.6738 - val_accuracy: 0.5803
Epoch 22/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6767 - accuracy:
0.5919 - val_loss: 0.6660 - val_accuracy: 0.5987
Epoch 23/275
51/51 [==============================] - 4s 83ms/step - loss: 0.6758 - accuracy:
0.5821 - val_loss: 0.6741 - val_accuracy: 0.5711
Epoch 24/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6832 - accuracy:
0.5564 - val_loss: 0.6783 - val_accuracy: 0.5870
Epoch 25/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6787 - accuracy:
0.5754 - val_loss: 0.6658 - val_accuracy: 0.5888
```

```
Epoch 26/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6757 - accuracy:
0.5895 - val_loss: 0.6741 - val_accuracy: 0.5876
Epoch 27/275
51/51 [==============================] - 4s 79ms/step - loss: 0.6753 - accuracy:
0.5803 - val_loss: 0.6801 - val_accuracy: 0.5846
Epoch 28/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6812 - accuracy:
0.5692 - val_loss: 0.6733 - val_accuracy: 0.5748
Epoch 29/275
51/51 [==============================] - 4s 79ms/step - loss: 0.6725 - accuracy:
0.5882 - val_loss: 0.6726 - val_accuracy: 0.5925
Epoch 30/275
51/51 [==============================] - 4s 85ms/step - loss: 0.6721 - accuracy:
0.5858 - val_loss: 0.6869 - val_accuracy: 0.5839
Epoch 31/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6696 - accuracy:
0.6005 - val_loss: 0.6760 - val_accuracy: 0.5999
Epoch 32/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6767 - accuracy:
0.5711 - val_loss: 0.6638 - val_accuracy: 0.5888
Epoch 33/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6735 - accuracy:
0.5864 - val_loss: 0.6701 - val_accuracy: 0.5987
Epoch 34/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6667 - accuracy:
0.5956 - val_loss: 0.6632 - val_accuracy: 0.5993
Epoch 35/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6686 - accuracy:
0.5944 - val_loss: 0.6665 - val_accuracy: 0.5858
Epoch 36/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6708 - accuracy:
0.5980 - val_loss: 0.6603 - val_accuracy: 0.5974
Epoch 37/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6683 - accuracy:
0.6005 - val_loss: 0.6662 - val_accuracy: 0.5888
Epoch 38/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6739 - accuracy:
0.5839 - val_loss: 0.6605 - val_accuracy: 0.6152
Epoch 39/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6703 - accuracy:
0.5931 - val_loss: 0.6729 - val_accuracy: 0.5705
Epoch 40/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6741 - accuracy:
0.5944 - val_loss: 0.6653 - val_accuracy: 0.5938
Epoch 41/275
51/51 [==============================] - 4s 82ms/step - loss: 0.6700 - accuracy:
0.5882 - val_loss: 0.6621 - val_accuracy: 0.6164
```

```
Epoch 42/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6673 - accuracy:
0.5980 - val_loss: 0.6521 - val_accuracy: 0.6195
Epoch 43/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6580 - accuracy:
0.6140 - val_loss: 0.6559 - val_accuracy: 0.6268
Epoch 44/275
51/51 [==============================] - 4s 82ms/step - loss: 0.6596 - accuracy:
0.6115 - val_loss: 0.6554 - val_accuracy: 0.6348
Epoch 45/275
51/51 [==============================] - 4s 79ms/step - loss: 0.6653 - accuracy:
0.6091 - val_loss: 0.6565 - val_accuracy: 0.6140
Epoch 46/275
51/51 [==============================] - 4s 75ms/step - loss: 0.6579 - accuracy:
0.6158 - val_loss: 0.6575 - val_accuracy: 0.6029
Epoch 47/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6635 - accuracy:
0.6042 - val_loss: 0.6586 - val_accuracy: 0.6134
Epoch 48/275
51/51 [==============================] - 4s 79ms/step - loss: 0.6668 - accuracy:
0.6029 - val_loss: 0.6474 - val_accuracy: 0.6207
Epoch 49/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6585 - accuracy:
0.6109 - val_loss: 0.6543 - val_accuracy: 0.6305
Epoch 50/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6603 - accuracy:
0.6183 - val_loss: 0.6504 - val_accuracy: 0.6293
Epoch 51/275
51/51 [==============================] - 4s 83ms/step - loss: 0.6564 - accuracy:
0.6213 - val_loss: 0.6529 - val_accuracy: 0.6213
Epoch 52/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6556 - accuracy:
0.6109 - val_loss: 0.6506 - val_accuracy: 0.6207
Epoch 53/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6512 - accuracy:
0.6299 - val_loss: 0.6451 - val_accuracy: 0.6311
Epoch 54/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6524 - accuracy:
0.6250 - val_loss: 0.6512 - val_accuracy: 0.6097
Epoch 55/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6480 - accuracy:
0.6170 - val_loss: 0.6367 - val_accuracy: 0.6305
Epoch 56/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6453 - accuracy:
0.6262 - val_loss: 0.6270 - val_accuracy: 0.6520
Epoch 57/275
51/51 [==============================] - 4s 84ms/step - loss: 0.6338 - accuracy:
0.6360 - val_loss: 0.6287 - val_accuracy: 0.6581
```

```
Epoch 58/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6366 - accuracy:
0.6477 - val_loss: 0.6283 - val_accuracy: 0.6489
Epoch 59/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6330 - accuracy:
0.6275 - val_loss: 0.6257 - val_accuracy: 0.6415
Epoch 60/275
51/51 [==============================] - 4s 83ms/step - loss: 0.6293 - accuracy:
0.6477 - val_loss: 0.6073 - val_accuracy: 0.6734
Epoch 61/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6225 - accuracy:
0.6581 - val_loss: 0.6197 - val_accuracy: 0.6538
Epoch 62/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6105 - accuracy:
0.6697 - val_loss: 0.6385 - val_accuracy: 0.6299
Epoch 63/275
51/51 [==============================] - 4s 83ms/step - loss: 0.6241 - accuracy:
0.6587 - val_loss: 0.6069 - val_accuracy: 0.6667
Epoch 64/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6136 - accuracy:
0.6685 - val_loss: 0.5939 - val_accuracy: 0.6703
Epoch 65/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6116 - accuracy:
0.6618 - val_loss: 0.6142 - val_accuracy: 0.6489
Epoch 66/275
51/51 [==============================] - 4s 78ms/step - loss: 0.6050 - accuracy:
0.6728 - val_loss: 0.5986 - val_accuracy: 0.6703
Epoch 67/275
51/51 [==============================] - 4s 77ms/step - loss: 0.6088 - accuracy:
0.6710 - val_loss: 0.6033 - val_accuracy: 0.6716
Epoch 68/275
51/51 [==============================] - 4s 76ms/step - loss: 0.6022 - accuracy:
0.6734 - val_loss: 0.5835 - val_accuracy: 0.6881
Epoch 69/275
51/51 [==============================] - 4s 83ms/step - loss: 0.5990 - accuracy:
0.6728 - val_loss: 0.6071 - val_accuracy: 0.6832
Epoch 70/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5966 - accuracy:
0.6771 - val_loss: 0.5910 - val_accuracy: 0.6789
Epoch 71/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5905 - accuracy:
0.6961 - val_loss: 0.6042 - val_accuracy: 0.6697
Epoch 72/275
51/51 [==============================] - 4s 78ms/step - loss: 0.5904 - accuracy:
0.6887 - val_loss: 0.6055 - val_accuracy: 0.6752
Epoch 73/275
51/51 [==============================] - 4s 78ms/step - loss: 0.5848 - accuracy:
0.6955 - val_loss: 0.5748 - val_accuracy: 0.6936
```

```
Epoch 74/275
51/51 [==============================] - 4s 75ms/step - loss: 0.5770 - accuracy:
0.6900 - val_loss: 0.5896 - val_accuracy: 0.6869
Epoch 75/275
51/51 [==============================] - 4s 84ms/step - loss: 0.5836 - accuracy:
0.7040 - val_loss: 0.5837 - val_accuracy: 0.6961
Epoch 76/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5843 - accuracy:
0.6783 - val_loss: 0.5677 - val_accuracy: 0.6985
Epoch 77/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5834 - accuracy:
0.6826 - val_loss: 0.5720 - val_accuracy: 0.6961
Epoch 78/275
51/51 [==============================] - 4s 78ms/step - loss: 0.5728 - accuracy:
0.6961 - val_loss: 0.5529 - val_accuracy: 0.7089
Epoch 79/275
51/51 [==============================] - 4s 75ms/step - loss: 0.5699 - accuracy:
0.6949 - val_loss: 0.5502 - val_accuracy: 0.7145
Epoch 80/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5697 - accuracy:
0.6961 - val_loss: 0.5726 - val_accuracy: 0.6942
Epoch 81/275
51/51 [==============================] - 4s 83ms/step - loss: 0.5714 - accuracy:
0.6906 - val_loss: 0.5551 - val_accuracy: 0.7077
Epoch 82/275
51/51 [==============================] - 4s 77ms/step - loss: 0.5551 - accuracy:
0.7077 - val_loss: 0.5569 - val_accuracy: 0.7077
Epoch 83/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5597 - accuracy:
0.6961 - val_loss: 0.5716 - val_accuracy: 0.6936
Epoch 84/275
51/51 [==============================] - 4s 83ms/step - loss: 0.5587 - accuracy:
0.7077 - val_loss: 0.5446 - val_accuracy: 0.7267
Epoch 85/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5569 - accuracy:
0.7175 - val_loss: 0.5713 - val_accuracy: 0.6869
Epoch 86/275
51/51 [==============================] - 4s 84ms/step - loss: 0.5608 - accuracy:
0.7071 - val_loss: 0.5931 - val_accuracy: 0.6685
Epoch 87/275
51/51 [==============================] - 4s 77ms/step - loss: 0.5565 - accuracy:
0.7096 - val_loss: 0.5558 - val_accuracy: 0.7102
Epoch 88/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5511 - accuracy:
0.7157 - val_loss: 0.5826 - val_accuracy: 0.6985
Epoch 89/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5535 - accuracy:
0.7132 - val_loss: 0.5616 - val_accuracy: 0.7028
```

```
Epoch 90/275
51/51 [==============================] - 4s 85ms/step - loss: 0.5610 - accuracy:
0.7114 - val_loss: 0.5184 - val_accuracy: 0.7359
Epoch 91/275
51/51 [==============================] - 4s 82ms/step - loss: 0.5447 - accuracy:
0.7194 - val_loss: 0.5090 - val_accuracy: 0.7390
Epoch 92/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5429 - accuracy:
0.7145 - val_loss: 0.5335 - val_accuracy: 0.7255
Epoch 93/275
51/51 [==============================] - 4s 84ms/step - loss: 0.5322 - accuracy:
0.7249 - val_loss: 0.5377 - val_accuracy: 0.7224
Epoch 94/275
51/51 [==============================] - 4s 75ms/step - loss: 0.5287 - accuracy:
0.7273 - val_loss: 0.5979 - val_accuracy: 0.6838
Epoch 95/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5417 - accuracy:
0.7261 - val_loss: 0.5294 - val_accuracy: 0.7224
Epoch 96/275
51/51 [==============================] - 4s 79ms/step - loss: 0.5205 - accuracy:
0.7439 - val_loss: 0.5140 - val_accuracy: 0.7335
Epoch 97/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5210 - accuracy:
0.7292 - val_loss: 0.5287 - val_accuracy: 0.7273
Epoch 98/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5211 - accuracy:
0.7347 - val_loss: 0.5008 - val_accuracy: 0.7555
Epoch 99/275
51/51 [==============================] - 4s 84ms/step - loss: 0.5154 - accuracy:
0.7439 - val_loss: 0.5073 - val_accuracy: 0.7279
Epoch 100/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5317 - accuracy:
0.7120 - val_loss: 0.5142 - val_accuracy: 0.7390
Epoch 101/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5138 - accuracy:
0.7377 - val_loss: 0.5063 - val_accuracy: 0.7475
Epoch 102/275
51/51 [==============================] - 4s 83ms/step - loss: 0.5030 - accuracy:
0.7488 - val_loss: 0.5013 - val_accuracy: 0.7604
Epoch 103/275
51/51 [==============================] - 4s 77ms/step - loss: 0.4995 - accuracy:
0.7457 - val_loss: 0.5216 - val_accuracy: 0.7426
Epoch 104/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5056 - accuracy:
0.7439 - val_loss: 0.4929 - val_accuracy: 0.7512
Epoch 105/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5007 - accuracy:
0.7512 - val_loss: 0.5072 - val_accuracy: 0.7463
```

```
Epoch 106/275
51/51 [==============================] - 4s 78ms/step - loss: 0.5157 - accuracy:
0.7402 - val_loss: 0.4777 - val_accuracy: 0.7635
Epoch 107/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4879 - accuracy:
0.7574 - val_loss: 0.5172 - val_accuracy: 0.7347
Epoch 108/275
51/51 [==============================] - 4s 76ms/step - loss: 0.5065 - accuracy:
0.7439 - val_loss: 0.4944 - val_accuracy: 0.7525
Epoch 109/275
51/51 [==============================] - 4s 79ms/step - loss: 0.4978 - accuracy:
0.7500 - val_loss: 0.4816 - val_accuracy: 0.7641
Epoch 110/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4917 - accuracy:
0.7604 - val_loss: 0.5004 - val_accuracy: 0.7469
Epoch 111/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4812 - accuracy:
0.7635 - val_loss: 0.4766 - val_accuracy: 0.7684
Epoch 112/275
51/51 [==============================] - 4s 83ms/step - loss: 0.4899 - accuracy:
0.7678 - val_loss: 0.4615 - val_accuracy: 0.7708
Epoch 113/275
51/51 [==============================] - 4s 78ms/step - loss: 0.4878 - accuracy:
0.7733 - val_loss: 0.4699 - val_accuracy: 0.7678
Epoch 114/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4741 - accuracy:
0.7684 - val_loss: 0.4726 - val_accuracy: 0.7727
Epoch 115/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4758 - accuracy:
0.7727 - val_loss: 0.4716 - val_accuracy: 0.7672
Epoch 116/275
51/51 [==============================] - 4s 85ms/step - loss: 0.4793 - accuracy:
0.7672 - val_loss: 0.4705 - val_accuracy: 0.7659
Epoch 117/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4827 - accuracy:
0.7623 - val_loss: 0.4645 - val_accuracy: 0.7708
Epoch 118/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4826 - accuracy:
0.7696 - val_loss: 0.4690 - val_accuracy: 0.7616
Epoch 119/275
51/51 [==============================] - 4s 79ms/step - loss: 0.4869 - accuracy:
0.7567 - val_loss: 0.4668 - val_accuracy: 0.7745
Epoch 120/275
51/51 [==============================] - 4s 77ms/step - loss: 0.4799 - accuracy:
0.7623 - val_loss: 0.4763 - val_accuracy: 0.7604
Epoch 121/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4726 - accuracy:
0.7592 - val_loss: 0.4609 - val_accuracy: 0.7837
```

```
Epoch 122/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4860 - accuracy:
0.7641 - val_loss: 0.4434 - val_accuracy: 0.7812
Epoch 123/275
51/51 [==============================] - 4s 77ms/step - loss: 0.4627 - accuracy:
0.7782 - val_loss: 0.4597 - val_accuracy: 0.7708
Epoch 124/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4691 - accuracy:
0.7616 - val_loss: 0.4594 - val_accuracy: 0.7745
Epoch 125/275
51/51 [==============================] - 4s 77ms/step - loss: 0.4707 - accuracy:
0.7635 - val_loss: 0.4475 - val_accuracy: 0.7733
Epoch 126/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4520 - accuracy:
0.7880 - val_loss: 0.4383 - val_accuracy: 0.7904
Epoch 127/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4693 - accuracy:
0.7733 - val_loss: 0.4792 - val_accuracy: 0.7708
Epoch 128/275
51/51 [==============================] - 4s 83ms/step - loss: 0.4389 - accuracy:
0.7886 - val_loss: 0.4413 - val_accuracy: 0.7923
Epoch 129/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4579 - accuracy:
0.7806 - val_loss: 0.4669 - val_accuracy: 0.7745
Epoch 130/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4497 - accuracy:
0.7874 - val_loss: 0.4822 - val_accuracy: 0.7714
Epoch 131/275
51/51 [==============================] - 4s 78ms/step - loss: 0.4473 - accuracy:
0.7843 - val_loss: 0.4204 - val_accuracy: 0.7947
Epoch 132/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4480 - accuracy:
0.7831 - val_loss: 0.4261 - val_accuracy: 0.7923
Epoch 133/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4410 - accuracy:
0.7825 - val_loss: 0.4198 - val_accuracy: 0.8021
Epoch 134/275
51/51 [==============================] - 4s 79ms/step - loss: 0.4491 - accuracy:
0.7763 - val_loss: 0.4792 - val_accuracy: 0.7708
Epoch 135/275
51/51 [==============================] - 4s 82ms/step - loss: 0.4379 - accuracy:
0.7843 - val_loss: 0.4374 - val_accuracy: 0.7978
Epoch 136/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4343 - accuracy:
0.7996 - val_loss: 0.3992 - val_accuracy: 0.8131
Epoch 137/275
51/51 [==============================] - 4s 80ms/step - loss: 0.4232 - accuracy:
0.7868 - val_loss: 0.4477 - val_accuracy: 0.7843
```

```
Epoch 138/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4371 - accuracy:
0.7904 - val_loss: 0.4792 - val_accuracy: 0.7586
Epoch 139/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4336 - accuracy:
0.7960 - val_loss: 0.4239 - val_accuracy: 0.8002
Epoch 140/275
51/51 [==============================] - 4s 80ms/step - loss: 0.4288 - accuracy:
0.7911 - val_loss: 0.4167 - val_accuracy: 0.8076
Epoch 141/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4191 - accuracy:
0.7917 - val_loss: 0.4282 - val_accuracy: 0.7904
Epoch 142/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4401 - accuracy:
0.7837 - val_loss: 0.4241 - val_accuracy: 0.7941
Epoch 143/275
51/51 [==============================] - 4s 80ms/step - loss: 0.4328 - accuracy:
0.8015 - val_loss: 0.4040 - val_accuracy: 0.8119
Epoch 144/275
51/51 [==============================] - 4s 78ms/step - loss: 0.4180 - accuracy:
0.8033 - val_loss: 0.4066 - val_accuracy: 0.8009
Epoch 145/275
51/51 [==============================] - 4s 79ms/step - loss: 0.4024 - accuracy:
0.8064 - val_loss: 0.4311 - val_accuracy: 0.7929
Epoch 146/275
51/51 [==============================] - 4s 79ms/step - loss: 0.4505 - accuracy:
0.7806 - val_loss: 0.4173 - val_accuracy: 0.7972
Epoch 147/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3995 - accuracy:
0.8094 - val_loss: 0.4350 - val_accuracy: 0.7929
Epoch 148/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4296 - accuracy:
0.7898 - val_loss: 0.3940 - val_accuracy: 0.8100
Epoch 149/275
51/51 [==============================] - 4s 85ms/step - loss: 0.4154 - accuracy:
0.8021 - val_loss: 0.4401 - val_accuracy: 0.7880
Epoch 150/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4284 - accuracy:
0.7953 - val_loss: 0.4077 - val_accuracy: 0.8051
Epoch 151/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4027 - accuracy:
0.8137 - val_loss: 0.3759 - val_accuracy: 0.8223
Epoch 152/275
51/51 [==============================] - 4s 84ms/step - loss: 0.4167 - accuracy:
0.7984 - val_loss: 0.4035 - val_accuracy: 0.8051
Epoch 153/275
51/51 [==============================] - 4s 77ms/step - loss: 0.4036 - accuracy:
0.8156 - val_loss: 0.3716 - val_accuracy: 0.8223
```

```
Epoch 154/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4175 - accuracy:
0.8021 - val_loss: 0.3760 - val_accuracy: 0.8303
Epoch 155/275
51/51 [==============================] - 4s 83ms/step - loss: 0.4045 - accuracy:
0.8088 - val_loss: 0.3984 - val_accuracy: 0.8107
Epoch 156/275
51/51 [==============================] - 4s 78ms/step - loss: 0.4170 - accuracy:
0.8045 - val_loss: 0.3747 - val_accuracy: 0.8254
Epoch 157/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3979 - accuracy:
0.8033 - val_loss: 0.3815 - val_accuracy: 0.8174
Epoch 158/275
51/51 [==============================] - 4s 76ms/step - loss: 0.4022 - accuracy:
0.8076 - val_loss: 0.3702 - val_accuracy: 0.8266
Epoch 159/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3968 - accuracy:
0.8156 - val_loss: 0.3854 - val_accuracy: 0.8229
Epoch 160/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3932 - accuracy:
0.8076 - val_loss: 0.3915 - val_accuracy: 0.8174
Epoch 161/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3714 - accuracy:
0.8327 - val_loss: 0.3592 - val_accuracy: 0.8388
Epoch 162/275
51/51 [==============================] - 4s 84ms/step - loss: 0.3934 - accuracy:
0.8125 - val_loss: 0.3629 - val_accuracy: 0.8278
Epoch 163/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3985 - accuracy:
0.8100 - val_loss: 0.4170 - val_accuracy: 0.8045
Epoch 164/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3966 - accuracy:
0.8137 - val_loss: 0.4093 - val_accuracy: 0.8113
Epoch 165/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3869 - accuracy:
0.8143 - val_loss: 0.3784 - val_accuracy: 0.8143
Epoch 166/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3745 - accuracy:
0.8180 - val_loss: 0.3784 - val_accuracy: 0.8211
Epoch 167/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3730 - accuracy:
0.8223 - val_loss: 0.3525 - val_accuracy: 0.8382
Epoch 168/275
51/51 [==============================] - 4s 83ms/step - loss: 0.3838 - accuracy:
0.8143 - val_loss: 0.3460 - val_accuracy: 0.8382
Epoch 169/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3793 - accuracy:
0.8174 - val_loss: 0.4066 - val_accuracy: 0.8150
```

```
Epoch 170/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3807 - accuracy:
0.8241 - val_loss: 0.3558 - val_accuracy: 0.8358
Epoch 171/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3897 - accuracy:
0.8168 - val_loss: 0.3604 - val_accuracy: 0.8388
Epoch 172/275
51/51 [==============================] - 4s 79ms/step - loss: 0.3873 - accuracy:
0.8199 - val_loss: 0.4242 - val_accuracy: 0.8021
Epoch 173/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3679 - accuracy:
0.8248 - val_loss: 0.3671 - val_accuracy: 0.8315
Epoch 174/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3476 - accuracy:
0.8352 - val_loss: 0.3982 - val_accuracy: 0.8229
Epoch 175/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3647 - accuracy:
0.8278 - val_loss: 0.3529 - val_accuracy: 0.8462
Epoch 176/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3622 - accuracy:
0.8370 - val_loss: 0.4098 - val_accuracy: 0.8082
Epoch 177/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3743 - accuracy:
0.8333 - val_loss: 0.3625 - val_accuracy: 0.8333
Epoch 178/275
51/51 [==============================] - 4s 81ms/step - loss: 0.3499 - accuracy:
0.8327 - val_loss: 0.3865 - val_accuracy: 0.8229
Epoch 179/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3621 - accuracy:
0.8315 - val_loss: 0.3738 - val_accuracy: 0.8370
Epoch 180/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3612 - accuracy:
0.8241 - val_loss: 0.3910 - val_accuracy: 0.8156
Epoch 181/275
51/51 [==============================] - 4s 79ms/step - loss: 0.3490 - accuracy:
0.8309 - val_loss: 0.4137 - val_accuracy: 0.8058
Epoch 182/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3797 - accuracy:
0.8413 - val_loss: 0.3471 - val_accuracy: 0.8401
Epoch 183/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3483 - accuracy:
0.8438 - val_loss: 0.3728 - val_accuracy: 0.8333
Epoch 184/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3549 - accuracy:
0.8358 - val_loss: 0.3295 - val_accuracy: 0.8431
Epoch 185/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3446 - accuracy:
0.8346 - val_loss: 0.3154 - val_accuracy: 0.8615
```

```
Epoch 186/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3384 - accuracy:
0.8493 - val_loss: 0.3113 - val_accuracy: 0.8591
Epoch 187/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3306 - accuracy:
0.8529 - val_loss: 0.3451 - val_accuracy: 0.8284
Epoch 188/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3315 - accuracy:
0.8487 - val_loss: 0.3053 - val_accuracy: 0.8560
Epoch 189/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3583 - accuracy:
0.8358 - val_loss: 0.3066 - val_accuracy: 0.8578
Epoch 190/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3489 - accuracy:
0.8444 - val_loss: 0.3030 - val_accuracy: 0.8609
Epoch 191/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3437 - accuracy:
0.8438 - val_loss: 0.3376 - val_accuracy: 0.8438
Epoch 192/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3331 - accuracy:
0.8444 - val_loss: 0.3175 - val_accuracy: 0.8517
Epoch 193/275
51/51 [==============================] - 4s 85ms/step - loss: 0.3434 - accuracy:
0.8450 - val_loss: 0.3171 - val_accuracy: 0.8529
Epoch 194/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3363 - accuracy:
0.8499 - val_loss: 0.2706 - val_accuracy: 0.8738
Epoch 195/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3190 - accuracy:
0.8499 - val_loss: 0.3384 - val_accuracy: 0.8401
Epoch 196/275
51/51 [==============================] - 4s 85ms/step - loss: 0.3409 - accuracy:
0.8536 - val_loss: 0.3104 - val_accuracy: 0.8591
Epoch 197/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3177 - accuracy:
0.8450 - val_loss: 0.2876 - val_accuracy: 0.8732
Epoch 198/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3313 - accuracy:
0.8413 - val_loss: 0.3026 - val_accuracy: 0.8585
Epoch 199/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3193 - accuracy:
0.8664 - val_loss: 0.3061 - val_accuracy: 0.8615
Epoch 200/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3073 - accuracy:
0.8591 - val_loss: 0.2902 - val_accuracy: 0.8762
Epoch 201/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3153 - accuracy:
0.8652 - val_loss: 0.2947 - val_accuracy: 0.8683
```

```
Epoch 202/275
51/51 [==============================] - 4s 83ms/step - loss: 0.3238 - accuracy:
0.8640 - val_loss: 0.3117 - val_accuracy: 0.8591
Epoch 203/275
51/51 [==============================] - 4s 79ms/step - loss: 0.3114 - accuracy:
0.8652 - val_loss: 0.2786 - val_accuracy: 0.8732
Epoch 204/275
51/51 [==============================] - 4s 84ms/step - loss: 0.3190 - accuracy:
0.8542 - val_loss: 0.2887 - val_accuracy: 0.8713
Epoch 205/275
51/51 [==============================] - 4s 77ms/step - loss: 0.3055 - accuracy:
0.8585 - val_loss: 0.2935 - val_accuracy: 0.8627
Epoch 206/275
51/51 [==============================] - 4s 85ms/step - loss: 0.3091 - accuracy:
0.8554 - val_loss: 0.3118 - val_accuracy: 0.8615
Epoch 207/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2931 - accuracy:
0.8640 - val_loss: 0.3336 - val_accuracy: 0.8499
Epoch 208/275
51/51 [==============================] - 4s 84ms/step - loss: 0.3248 - accuracy:
0.8578 - val_loss: 0.2543 - val_accuracy: 0.8934
Epoch 209/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2996 - accuracy:
0.8640 - val_loss: 0.2837 - val_accuracy: 0.8676
Epoch 210/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3004 - accuracy:
0.8664 - val_loss: 0.2886 - val_accuracy: 0.8732
Epoch 211/275
51/51 [==============================] - 4s 80ms/step - loss: 0.2786 - accuracy:
0.8738 - val_loss: 0.3059 - val_accuracy: 0.8621
Epoch 212/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3193 - accuracy:
0.8542 - val_loss: 0.2722 - val_accuracy: 0.8799
Epoch 213/275
51/51 [==============================] - 4s 76ms/step - loss: 0.3087 - accuracy:
0.8591 - val_loss: 0.2713 - val_accuracy: 0.8725
Epoch 214/275
51/51 [==============================] - 4s 78ms/step - loss: 0.3065 - accuracy:
0.8578 - val_loss: 0.2583 - val_accuracy: 0.8793
Epoch 215/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2997 - accuracy:
0.8646 - val_loss: 0.3191 - val_accuracy: 0.8536
Epoch 216/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2799 - accuracy:
0.8799 - val_loss: 0.2637 - val_accuracy: 0.8787
Epoch 217/275
51/51 [==============================] - 4s 86ms/step - loss: 0.2726 - accuracy:
0.8787 - val_loss: 0.2512 - val_accuracy: 0.8854
```

```
Epoch 218/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2669 - accuracy:
0.8842 - val_loss: 0.2483 - val_accuracy: 0.8897
Epoch 219/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2697 - accuracy:
0.8830 - val_loss: 0.2559 - val_accuracy: 0.8866
Epoch 220/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2790 - accuracy:
0.8805 - val_loss: 0.2572 - val_accuracy: 0.8903
Epoch 221/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2702 - accuracy:
0.8805 - val_loss: 0.2690 - val_accuracy: 0.8750
Epoch 222/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2734 - accuracy:
0.8817 - val_loss: 0.2393 - val_accuracy: 0.8958
Epoch 223/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2782 - accuracy:
0.8689 - val_loss: 0.3111 - val_accuracy: 0.8529
Epoch 224/275
51/51 [==============================] - 4s 79ms/step - loss: 0.2768 - accuracy:
0.8738 - val_loss: 0.2508 - val_accuracy: 0.8909
Epoch 225/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2824 - accuracy:
0.8793 - val_loss: 0.2537 - val_accuracy: 0.8824
Epoch 226/275
51/51 [==============================] - 4s 83ms/step - loss: 0.2590 - accuracy:
0.8891 - val_loss: 0.2757 - val_accuracy: 0.8873
Epoch 227/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2721 - accuracy:
0.8817 - val_loss: 0.2601 - val_accuracy: 0.8756
Epoch 228/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2741 - accuracy:
0.8756 - val_loss: 0.2648 - val_accuracy: 0.8836
Epoch 229/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2732 - accuracy:
0.8842 - val_loss: 0.2413 - val_accuracy: 0.8903
Epoch 230/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2555 - accuracy:
0.8903 - val_loss: 0.2401 - val_accuracy: 0.8897
Epoch 231/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2478 - accuracy:
0.8934 - val_loss: 0.2812 - val_accuracy: 0.8787
Epoch 232/275
51/51 [==============================] - 4s 83ms/step - loss: 0.2614 - accuracy:
0.8805 - val_loss: 0.2594 - val_accuracy: 0.8885
Epoch 233/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2598 - accuracy:
0.8860 - val_loss: 0.2829 - val_accuracy: 0.8744
```

```
Epoch 234/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2923 - accuracy:
0.8652 - val_loss: 0.2601 - val_accuracy: 0.8848
Epoch 235/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2777 - accuracy:
0.8781 - val_loss: 0.2924 - val_accuracy: 0.8652
Epoch 236/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2737 - accuracy:
0.8805 - val_loss: 0.2763 - val_accuracy: 0.8689
Epoch 237/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2624 - accuracy:
0.8836 - val_loss: 0.2609 - val_accuracy: 0.8915
Epoch 238/275
51/51 [==============================] - 4s 79ms/step - loss: 0.2765 - accuracy:
0.8744 - val_loss: 0.2355 - val_accuracy: 0.8977
Epoch 239/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2348 - accuracy:
0.8977 - val_loss: 0.2315 - val_accuracy: 0.8958
Epoch 240/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2498 - accuracy:
0.8964 - val_loss: 0.2344 - val_accuracy: 0.8903
Epoch 241/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2450 - accuracy:
0.8940 - val_loss: 0.2406 - val_accuracy: 0.9001
Epoch 242/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2540 - accuracy:
0.8903 - val_loss: 0.3073 - val_accuracy: 0.8768
Epoch 243/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2927 - accuracy:
0.8719 - val_loss: 0.2275 - val_accuracy: 0.8977
Epoch 244/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2615 - accuracy:
0.8891 - val_loss: 0.2384 - val_accuracy: 0.8964
Epoch 245/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2281 - accuracy:
0.9075 - val_loss: 0.3007 - val_accuracy: 0.8621
Epoch 246/275
51/51 [==============================] - 4s 84ms/step - loss: 0.2512 - accuracy:
0.8909 - val_loss: 0.2247 - val_accuracy: 0.9013
Epoch 247/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2310 - accuracy:
0.8995 - val_loss: 0.2368 - val_accuracy: 0.8952
Epoch 248/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2525 - accuracy:
0.8958 - val_loss: 0.2181 - val_accuracy: 0.9032
Epoch 249/275
51/51 [==============================] - 4s 79ms/step - loss: 0.2363 - accuracy:
0.8995 - val_loss: 0.2124 - val_accuracy: 0.9044
```

```
Epoch 250/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2489 - accuracy:
0.8928 - val_loss: 0.2670 - val_accuracy: 0.8866
Epoch 251/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2539 - accuracy:
0.8915 - val_loss: 0.2229 - val_accuracy: 0.8952
Epoch 252/275
51/51 [==============================] - 4s 82ms/step - loss: 0.2612 - accuracy:
0.8909 - val_loss: 0.2742 - val_accuracy: 0.8781
Epoch 253/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2398 - accuracy:
0.9007 - val_loss: 0.2149 - val_accuracy: 0.9124
Epoch 254/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2229 - accuracy:
0.8989 - val_loss: 0.2228 - val_accuracy: 0.9062
Epoch 255/275
51/51 [==============================] - 4s 85ms/step - loss: 0.2363 - accuracy:
0.8964 - val_loss: 0.1997 - val_accuracy: 0.9161
Epoch 256/275
51/51 [==============================] - 4s 78ms/step - loss: 0.2470 - accuracy:
0.8940 - val_loss: 0.2568 - val_accuracy: 0.8854
Epoch 257/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2318 - accuracy:
0.9038 - val_loss: 0.2555 - val_accuracy: 0.8903
Epoch 258/275
51/51 [==============================] - 4s 80ms/step - loss: 0.2536 - accuracy:
0.8854 - val_loss: 0.2638 - val_accuracy: 0.8848
Epoch 259/275
51/51 [==============================] - 4s 80ms/step - loss: 0.2393 - accuracy:
0.9020 - val_loss: 0.2437 - val_accuracy: 0.8989
Epoch 260/275
51/51 [==============================] - 4s 79ms/step - loss: 0.2349 - accuracy:
0.8958 - val_loss: 0.1785 - val_accuracy: 0.9295
Epoch 261/275
51/51 [==============================] - 4s 85ms/step - loss: 0.2104 - accuracy:
0.9020 - val_loss: 0.2704 - val_accuracy: 0.8860
Epoch 262/275
51/51 [==============================] - 5s 89ms/step - loss: 0.2363 - accuracy:
0.9013 - val_loss: 0.2349 - val_accuracy: 0.8977
Epoch 263/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2418 - accuracy:
0.8946 - val_loss: 0.2116 - val_accuracy: 0.9081
Epoch 264/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2118 - accuracy:
0.9124 - val_loss: 0.2280 - val_accuracy: 0.9093
Epoch 265/275
51/51 [==============================] - 4s 81ms/step - loss: 0.2198 - accuracy:
0.9093 - val_loss: 0.2357 - val_accuracy: 0.8940
```

```
Epoch 266/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2357 - accuracy:
0.9093 - val_loss: 0.2161 - val_accuracy: 0.9130
Epoch 267/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2349 - accuracy:
0.8952 - val_loss: 0.2284 - val_accuracy: 0.8946
Epoch 268/275
51/51 [==============================] - 4s 81ms/step - loss: 0.2059 - accuracy:
0.9185 - val_loss: 0.2091 - val_accuracy: 0.9105
Epoch 269/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2201 - accuracy:
0.9069 - val_loss: 0.1932 - val_accuracy: 0.9130
Epoch 270/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2222 - accuracy:
0.9075 - val_loss: 0.2188 - val_accuracy: 0.9142
Epoch 271/275
51/51 [==============================] - 4s 86ms/step - loss: 0.2666 - accuracy:
0.8799 - val_loss: 0.2444 - val_accuracy: 0.8830
Epoch 272/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2196 - accuracy:
0.8983 - val_loss: 0.2355 - val_accuracy: 0.8946
Epoch 273/275
51/51 [==============================] - 4s 76ms/step - loss: 0.2021 - accuracy:
0.9130 - val_loss: 0.2146 - val_accuracy: 0.9136
Epoch 274/275
51/51 [==============================] - 4s 79ms/step - loss: 0.2095 - accuracy:
0.9185 - val_loss: 0.2535 - val_accuracy: 0.8897
Epoch 275/275
51/51 [==============================] - 4s 77ms/step - loss: 0.2004 - accuracy:
0.9136 - val_loss: 0.1868 - val_accuracy: 0.9265
```

[ ]:
```python
# checking the accuracy of the model
scores = model.evaluate(test_ds)
```

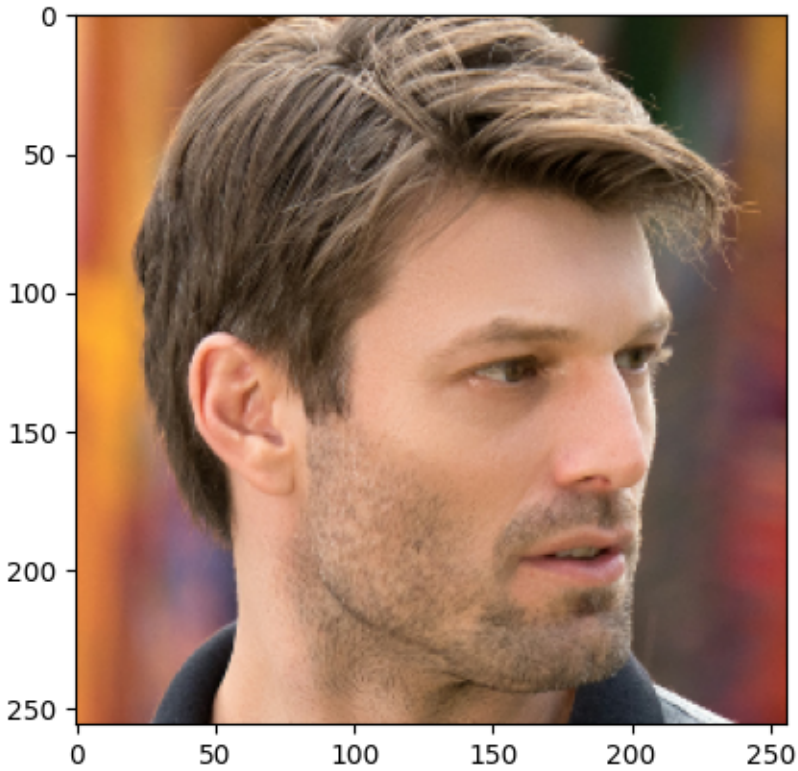
```
51/51 [==============================] - 1s 23ms/step - loss: 0.1868 - accuracy:
0.9265
```

[ ]:
```python
# accuracy of the model is 92.65%
```

[ ]:
```python
# predicting labels for the batch of images
import numpy as np

for image_batch, label_batch in imgdata.take(1):

    first_image = image_batch[0].numpy().astype('uint8')
    first_label = label_batch[0].numpy()
```

```
    print("first image to predict")
    plt.imshow(first_image)
    print("Actual label : ",class_names[first_label])


    batch_pred = model.predict(image_batch)
    print("Pred label : ",class_names[np.argmax(batch_pred[0])])
```

```
first image to predict
Actual label :  real
1/1 [==============================] - 0s 141ms/step
Pred label :  real
```

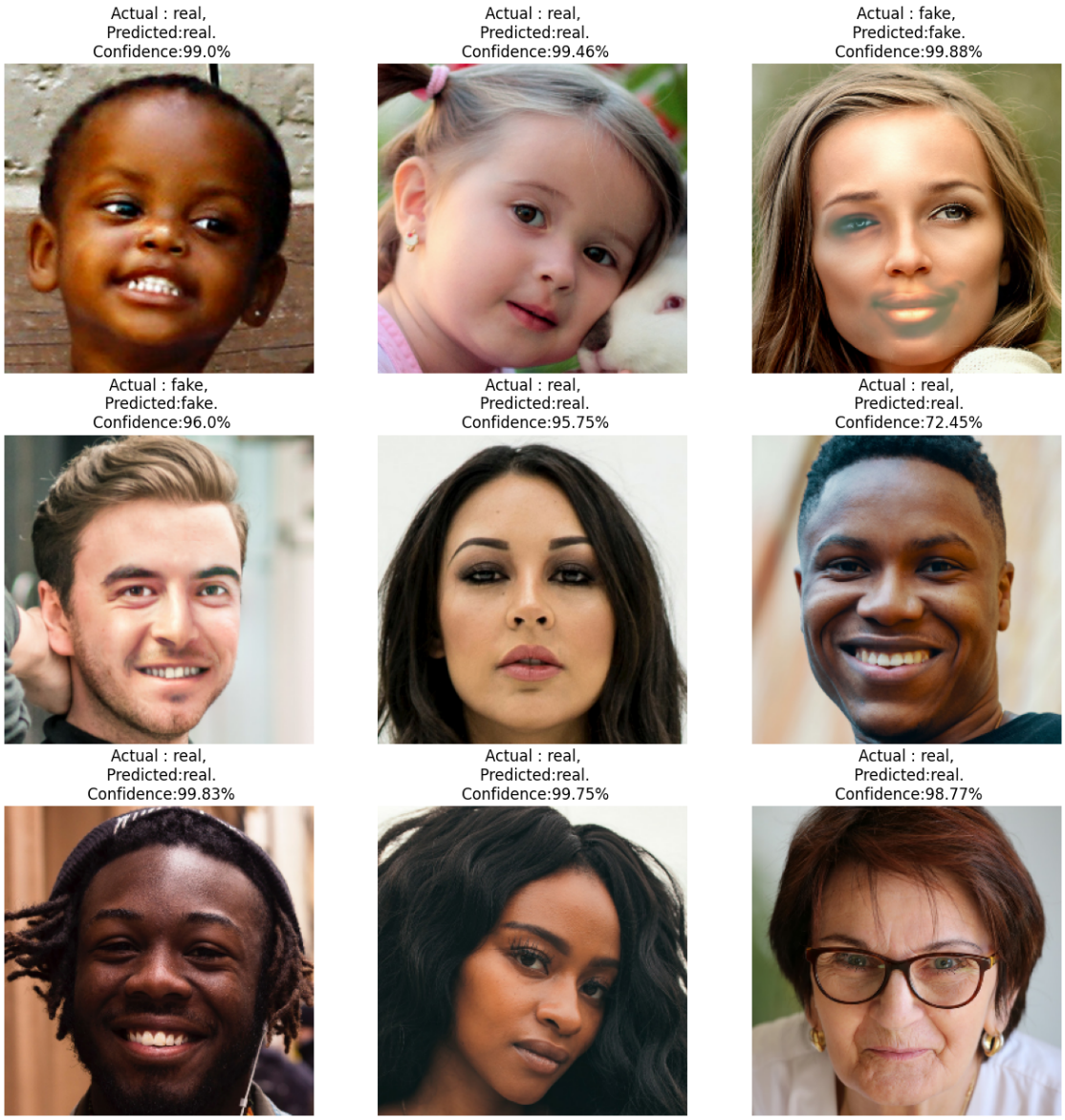


[ ]:
```
# image prediction function using the model
def pred(model, img):
    img_array = tf.keras.preprocessing.image.img_to_array(images[i].numpy())
    img_array = tf.expand_dims(img_array, 0)

    predictions = model.predict(img_array)

    predicted_class = class_names[np.argmax(predictions[0])]
    confidence = round(100 * (np.max(predictions[0])), 2)
```

```
    return predicted_class, confidence
```

[ ]:
```
# Displaying Sample Predictions with Confidence
plt.figure(figsize=(15, 15))

for images, labels in test_ds.take(1):
    for i in range(9):
        ax = plt.subplot(3,3, i+1)
        plt.imshow(images[i].numpy().astype("uint8"))

        predicted_class, confidence = pred(model, images[i].numpy())
        actual_class = class_names[labels[i]]

        plt.title(f"Actual : {actual_class},\n Predicted:{predicted_class}.\n␣
 ↪Confidence:{confidence}%")

        plt.axis("off")
```

```
1/1 [==============================] - 1s 602ms/step
1/1 [==============================] - 0s 27ms/step
1/1 [==============================] - 0s 24ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 28ms/step
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 28ms/step
```



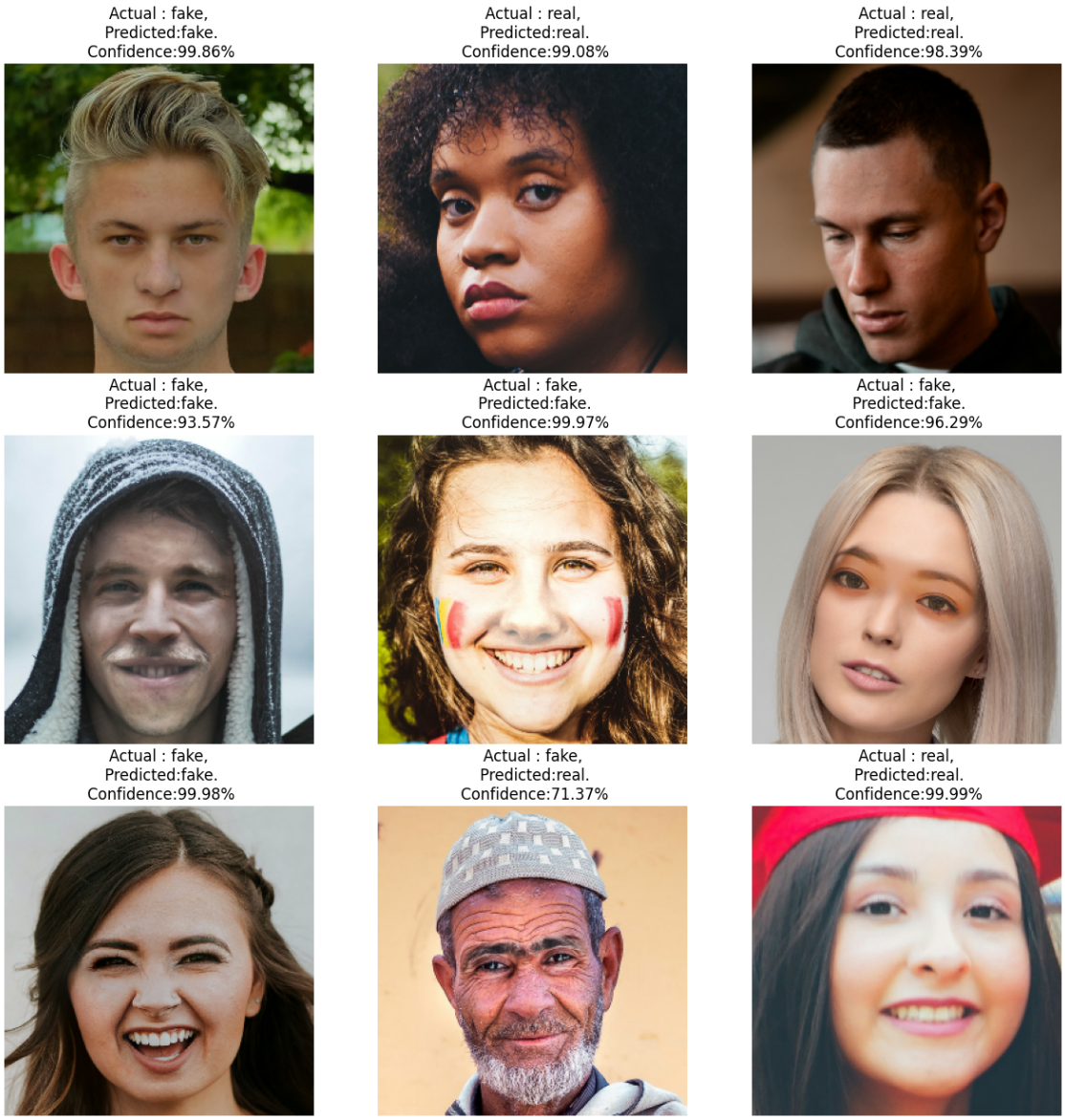
[ ]:
```
# saving the model
import pickle

with open('model_f_real_pickle_final','wb') as f:
  pickle.dump(model,f)
```

[ ]:
```
# to run the pickle(saved model)
# import pickle

with open('model_f_real_pickle_final','rb') as f:
  model_saved = pickle.load(f)
```

```
#to predict the model
#model_saved.predict("give input")
```

[ ]:
```
plt.figure(figsize=(15, 15))

for images, labels in test_ds.take(1):
    for i in range(9):
        ax = plt.subplot(3,3, i+1)
        plt.imshow(images[i].numpy().astype("uint8"))

        predicted_class, confidence = pred(model_saved, images[i].numpy())
        actual_class = class_names[labels[i]]

        plt.title(f"Actual : {actual_class},\n Predicted:{predicted_class}.\n␣
 ↪Confidence:{confidence}%")

        plt.axis("off")
```

```
1/1 [==============================] - 0s 214ms/step
1/1 [==============================] - 0s 54ms/step
1/1 [==============================] - 0s 29ms/step
1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 55ms/step
1/1 [==============================] - 0s 34ms/step
1/1 [==============================] - 0s 31ms/step
1/1 [==============================] - 0s 47ms/step
1/1 [==============================] - 0s 100ms/step
```



[ ]:
```
plt.figure(figsize=(15, 15))
ax = plt.subplot(3,3)
plt.imshow(imgdata[0].numpy().astype("uint8"))

predicted_class, confidence = pred(model_saved, imgdata[0].numpy())
actual_class = class_names[labels[i]]

plt.title(f"Actual : {actual_class},\n Predicted:{predicted_class}.\n␣
 ↪Confidence:{confidence}%")

plt.axis("off")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-9695e35acdf8> in <cell line: 1>()
----> 1 plt.figure(figsize=(15, 15))
      2 ax = plt.subplot(3,3)
      3 plt.imshow(imgdata[0].numpy().astype("uint8"))
      4
      5 predicted_class, confidence = pred(model_saved, imgdata[0].numpy())

NameError: name 'plt' is not defined
```

[1]:
```
!pip install gradio
```

```
Collecting gradio
  Downloading gradio-4.31.3-py3-none-any.whl (12.3 MB)
                            12.3/12.3 MB
46.0 MB/s eta 0:00:00
Collecting aiofiles<24.0,>=22.0 (from gradio)
  Downloading aiofiles-23.2.1-py3-none-any.whl (15 kB)
Requirement already satisfied: altair<6.0,>=4.2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (4.2.2)
Collecting fastapi (from gradio)
  Downloading fastapi-0.111.0-py3-none-any.whl (91 kB)
                            92.0/92.0 kB
12.4 MB/s eta 0:00:00
Collecting ffmpy (from gradio)
  Downloading ffmpy-0.3.2.tar.gz (5.5 kB)
  Preparing metadata (setup.py) … done
Collecting gradio-client==0.16.3 (from gradio)
  Downloading gradio_client-0.16.3-py3-none-any.whl (315 kB)
                            315.8/315.8

kB 33.6 MB/s eta 0:00:00
Collecting httpx>=0.24.1 (from gradio)
  Downloading httpx-0.27.0-py3-none-any.whl (75 kB)
                            75.6/75.6 kB
9.6 MB/s eta 0:00:00
Requirement already satisfied: huggingface-hub>=0.19.3 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.20.3)
Requirement already satisfied: importlib-resources<7.0,>=1.3 in
/usr/local/lib/python3.10/dist-packages (from gradio) (6.4.0)
Requirement already satisfied: jinja2<4.0 in /usr/local/lib/python3.10/dist-
packages (from gradio) (3.1.4)
Requirement already satisfied: markupsafe~=2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.1.5)
Requirement already satisfied: matplotlib~=3.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (3.7.1)
```

```
Requirement already satisfied: numpy~=1.0 in /usr/local/lib/python3.10/dist-
packages (from gradio) (1.25.2)
Collecting orjson~=3.0 (from gradio)
  Downloading
orjson-3.10.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (142
kB)
                           142.5/142.5
kB 18.0 MB/s eta 0:00:00
Requirement already satisfied: packaging in
/usr/local/lib/python3.10/dist-packages (from gradio) (24.0)
Requirement already satisfied: pandas<3.0,>=1.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.0.3)
Requirement already satisfied: pillow<11.0,>=8.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (9.4.0)
Requirement already satisfied: pydantic>=2.0 in /usr/local/lib/python3.10/dist-
packages (from gradio) (2.7.1)
Collecting pydub (from gradio)
  Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)
Collecting python-multipart>=0.0.9 (from gradio)
  Downloading python_multipart-0.0.9-py3-none-any.whl (22 kB)
Requirement already satisfied: pyyaml<7.0,>=5.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (6.0.1)
Collecting ruff>=0.2.2 (from gradio)
  Downloading ruff-0.4.4-py3-none-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(8.7 MB)
                           8.7/8.7 MB
63.6 MB/s eta 0:00:00
Collecting semantic-version~=2.0 (from gradio)
  Downloading semantic_version-2.10.0-py2.py3-none-any.whl (15 kB)
Collecting tomlkit==0.12.0 (from gradio)
  Downloading tomlkit-0.12.0-py3-none-any.whl (37 kB)
Collecting typer<1.0,>=0.12 (from gradio)
  Downloading typer-0.12.3-py3-none-any.whl (47 kB)
                           47.2/47.2 kB
5.9 MB/s eta 0:00:00
Requirement already satisfied: typing-extensions~=4.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (4.11.0)
Requirement already satisfied: urllib3~=2.0 in /usr/local/lib/python3.10/dist-
packages (from gradio) (2.0.7)
Collecting uvicorn>=0.14.0 (from gradio)
  Downloading uvicorn-0.29.0-py3-none-any.whl (60 kB)
                           60.8/60.8 kB
6.4 MB/s eta 0:00:00
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-
packages (from gradio-client==0.16.3->gradio) (2023.6.0)
Collecting websockets<12.0,>=10.0 (from gradio-client==0.16.3->gradio)
  Downloading websockets-11.0.3-cp310-cp310-manylinux_2_5_x86_64.manylinux1_x86_
```

```
64.manylinux_2_17_x86_64.manylinux2014_x86_64.whl (129 kB)
                              129.9/129.9
kB 12.1 MB/s eta 0:00:00
Requirement already satisfied: entrypoints in
/usr/local/lib/python3.10/dist-packages (from altair<6.0,>=4.2.0->gradio) (0.4)
Requirement already satisfied: jsonschema>=3.0 in
/usr/local/lib/python3.10/dist-packages (from altair<6.0,>=4.2.0->gradio)
(4.19.2)
Requirement already satisfied: toolz in /usr/local/lib/python3.10/dist-packages
(from altair<6.0,>=4.2.0->gradio) (0.12.1)
Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages
(from httpx>=0.24.1->gradio) (3.7.1)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-
packages (from httpx>=0.24.1->gradio) (2024.2.2)
Collecting httpcore==1.* (from httpx>=0.24.1->gradio)
  Downloading httpcore-1.0.5-py3-none-any.whl (77 kB)
                              77.9/77.9 kB
8.7 MB/s eta 0:00:00
Requirement already satisfied: idna in /usr/local/lib/python3.10/dist-
packages (from httpx>=0.24.1->gradio) (3.7)
Requirement already satisfied: sniffio in /usr/local/lib/python3.10/dist-
packages (from httpx>=0.24.1->gradio) (1.3.1)
Collecting h11<0.15,>=0.13 (from httpcore==1.*->httpx>=0.24.1->gradio)
  Downloading h11-0.14.0-py3-none-any.whl (58 kB)
                              58.3/58.3 kB
6.5 MB/s eta 0:00:00
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from huggingface-hub>=0.19.3->gradio) (3.14.0)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-
packages (from huggingface-hub>=0.19.3->gradio) (2.31.0)
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.10/dist-
packages (from huggingface-hub>=0.19.3->gradio) (4.66.4)
Requirement already satisfied: contourpy>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-
packages (from matplotlib~=3.0->gradio) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (1.4.5)
Requirement already satisfied: pyparsing>=2.3.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-
packages (from pandas<3.0,>=1.0->gradio) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-
```

```
packages (from pandas<3.0,>=1.0->gradio) (2024.1)
Requirement already satisfied: annotated-types>=0.4.0 in
/usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio) (0.6.0)
Requirement already satisfied: pydantic-core==2.18.2 in
/usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio) (2.18.2)
Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.10/dist-
packages (from typer<1.0,>=0.12->gradio) (8.1.7)
Collecting shellingham>=1.3.0 (from typer<1.0,>=0.12->gradio)
  Downloading shellingham-1.5.4-py2.py3-none-any.whl (9.8 kB)
Requirement already satisfied: rich>=10.11.0 in /usr/local/lib/python3.10/dist-
packages (from typer<1.0,>=0.12->gradio) (13.7.1)
Collecting starlette<0.38.0,>=0.37.2 (from fastapi->gradio)
  Downloading starlette-0.37.2-py3-none-any.whl (71 kB)
                             71.9/71.9 kB
7.8 MB/s eta 0:00:00
Collecting fastapi-cli>=0.0.2 (from fastapi->gradio)
  Downloading fastapi_cli-0.0.3-py3-none-any.whl (9.2 kB)
Collecting ujson!=4.0.2,!=4.1.0,!=4.2.0,!=4.3.0,!=5.0.0,!=5.1.0,>=4.0.1 (from
fastapi->gradio)
  Downloading
ujson-5.10.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (53 kB)
                             53.6/53.6 kB
6.8 MB/s eta 0:00:00
Collecting email_validator>=2.0.0 (from fastapi->gradio)
  Downloading email_validator-2.1.1-py3-none-any.whl (30 kB)
Collecting dnspython>=2.0.0 (from email_validator>=2.0.0->fastapi->gradio)
  Downloading dnspython-2.6.1-py3-none-any.whl (307 kB)
                             307.7/307.7
kB 33.2 MB/s eta 0:00:00
Requirement already satisfied: attrs>=22.2.0 in
/usr/local/lib/python3.10/dist-packages (from
jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (23.2.0)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in
/usr/local/lib/python3.10/dist-packages (from
jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (2023.12.1)
Requirement already satisfied: referencing>=0.28.4 in
/usr/local/lib/python3.10/dist-packages (from
jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (0.35.1)
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.10/dist-
packages (from jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (0.18.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-
packages (from python-dateutil>=2.7->matplotlib~=3.0->gradio) (1.16.0)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from
rich>=10.11.0->typer<1.0,>=0.12->gradio) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from
```

```
rich>=10.11.0->typer<1.0,>=0.12->gradio) (2.16.1)
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-
packages (from anyio->httpx>=0.24.1->gradio) (1.2.1)
Collecting httptools>=0.5.0 (from uvicorn>=0.14.0->gradio)
  Downloading httptools-0.6.1-cp310-cp310-manylinux_2_5_x86_64.manylinux1_x86_64
.manylinux_2_17_x86_64.manylinux2014_x86_64.whl (341 kB)
                           341.4/341.4
kB 34.1 MB/s eta 0:00:00
Collecting python-dotenv>=0.13 (from uvicorn>=0.14.0->gradio)
  Downloading python_dotenv-1.0.1-py3-none-any.whl (19 kB)
Collecting uvloop!=0.15.0,!=0.15.1,>=0.14.0 (from uvicorn>=0.14.0->gradio)
  Downloading
uvloop-0.19.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.4
MB)
                           3.4/3.4 MB
90.6 MB/s eta 0:00:00
Collecting watchfiles>=0.13 (from uvicorn>=0.14.0->gradio)
  Downloading
watchfiles-0.21.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(1.3 MB)
                           1.3/1.3 MB
74.5 MB/s eta 0:00:00
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->huggingface-
hub>=0.19.3->gradio) (3.3.2)
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.10/dist-
packages (from markdown-it-py>=2.2.0->rich>=10.11.0->typer<1.0,>=0.12->gradio)
(0.1.2)
Building wheels for collected packages: ffmpy
  Building wheel for ffmpy (setup.py) … done
  Created wheel for ffmpy: filename=ffmpy-0.3.2-py3-none-any.whl size=5584
sha256=994e36316325478752fadad393a3a448ca046cc1865fcadb6fa19a1c9cdd3328
  Stored in directory: /root/.cache/pip/wheels/bd/65/9a/671fc6dcde07d4418df0c592
f8df512b26d7a0029c2a23dd81
Successfully built ffmpy
Installing collected packages: pydub, ffmpy, websockets, uvloop, ujson, tomlkit,
shellingham, semantic-version, ruff, python-multipart, python-dotenv, orjson,
httptools, h11, dnspython, aiofiles, watchfiles, uvicorn, starlette, httpcore,
email_validator, typer, httpx, gradio-client, fastapi-cli, fastapi, gradio
  Attempting uninstall: typer
    Found existing installation: typer 0.9.4
    Uninstalling typer-0.9.4:
      Successfully uninstalled typer-0.9.4
```

```
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
spacy 3.7.4 requires typer<0.10.0,>=0.3.0, but you have typer 0.12.3 which is
incompatible.
weasel 0.3.4 requires typer<0.10.0,>=0.3.0, but you have typer 0.12.3 which is
incompatible.
Successfully installed aiofiles-23.2.1 dnspython-2.6.1 email_validator-2.1.1
fastapi-0.111.0 fastapi-cli-0.0.3 ffmpy-0.3.2 gradio-4.31.3 gradio-client-0.16.3
h11-0.14.0 httpcore-1.0.5 httptools-0.6.1 httpx-0.27.0 orjson-3.10.3
pydub-0.25.1 python-dotenv-1.0.1 python-multipart-0.0.9 ruff-0.4.4 semantic-
version-2.10.0 shellingham-1.5.4 starlette-0.37.2 tomlkit-0.12.0 typer-0.12.3
ujson-5.10.0 uvicorn-0.29.0 uvloop-0.19.0 watchfiles-0.21.0 websockets-11.0.3
```

[2]:
```
!pip install facenet_pytorch
```

```
Collecting facenet_pytorch
  Downloading facenet_pytorch-2.6.0-py3-none-any.whl (1.9 MB)
                           1.9/1.9 MB
13.6 MB/s eta 0:00:00
Requirement already satisfied: numpy<2.0.0,>=1.24.0 in
/usr/local/lib/python3.10/dist-packages (from facenet_pytorch) (1.25.2)
Collecting Pillow<10.3.0,>=10.2.0 (from facenet_pytorch)
  Downloading pillow-10.2.0-cp310-cp310-manylinux_2_28_x86_64.whl (4.5 MB)
                           4.5/4.5 MB
35.3 MB/s eta 0:00:00
Requirement already satisfied: requests<3.0.0,>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from facenet_pytorch) (2.31.0)
Requirement already satisfied: torch<2.3.0,>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from facenet_pytorch) (2.2.1+cu121)
Requirement already satisfied: torchvision<0.18.0,>=0.17.0 in
/usr/local/lib/python3.10/dist-packages (from facenet_pytorch) (0.17.1+cu121)
Requirement already satisfied: tqdm<5.0.0,>=4.0.0 in
/usr/local/lib/python3.10/dist-packages (from facenet_pytorch) (4.66.4)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.0.0->facenet_pytorch) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3.0.0,>=2.0.0->facenet_pytorch) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from
requests<3.0.0,>=2.0.0->facenet_pytorch) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from
```

```
requests<3.0.0,>=2.0.0->facenet_pytorch) (2024.2.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from torch<2.3.0,>=2.2.0->facenet_pytorch) (3.14.0)
Requirement already satisfied: typing-extensions>=4.8.0 in
/usr/local/lib/python3.10/dist-packages (from
torch<2.3.0,>=2.2.0->facenet_pytorch) (4.11.0)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages
(from torch<2.3.0,>=2.2.0->facenet_pytorch) (1.12)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-
packages (from torch<2.3.0,>=2.2.0->facenet_pytorch) (3.3)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
(from torch<2.3.0,>=2.2.0->facenet_pytorch) (3.1.4)
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages
(from torch<2.3.0,>=2.2.0->facenet_pytorch) (2023.6.0)
Collecting nvidia-cuda-nvrtc-cu12==12.1.105 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cuda_nvrtc_cu12-12.1.105-py3-none-manylinux1_x86_64.whl
(23.7 MB)
Collecting nvidia-cuda-runtime-cu12==12.1.105 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cuda_runtime_cu12-12.1.105-py3-none-manylinux1_x86_64.whl
(823 kB)
Collecting nvidia-cuda-cupti-cu12==12.1.105 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cuda_cupti_cu12-12.1.105-py3-none-manylinux1_x86_64.whl
(14.1 MB)
Collecting nvidia-cudnn-cu12==8.9.2.26 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cudnn_cu12-8.9.2.26-py3-none-manylinux1_x86_64.whl (731.7
MB)
Collecting nvidia-cublas-cu12==12.1.3.1 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cublas_cu12-12.1.3.1-py3-none-manylinux1_x86_64.whl (410.6
MB)
Collecting nvidia-cufft-cu12==11.0.2.54 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cufft_cu12-11.0.2.54-py3-none-manylinux1_x86_64.whl (121.6
MB)
Collecting nvidia-curand-cu12==10.3.2.106 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_curand_cu12-10.3.2.106-py3-none-manylinux1_x86_64.whl
(56.5 MB)
Collecting nvidia-cusolver-cu12==11.4.5.107 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_cusolver_cu12-11.4.5.107-py3-none-manylinux1_x86_64.whl
(124.2 MB)
Collecting nvidia-cusparse-cu12==12.1.0.106 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
```

```
  Using cached nvidia_cusparse_cu12-12.1.0.106-py3-none-manylinux1_x86_64.whl
(196.0 MB)
Collecting nvidia-nccl-cu12==2.19.3 (from torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_nccl_cu12-2.19.3-py3-none-manylinux1_x86_64.whl (166.0 MB)
Collecting nvidia-nvtx-cu12==12.1.105 (from
torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_nvtx_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (99 kB)
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-
packages (from torch<2.3.0,>=2.2.0->facenet_pytorch) (2.2.0)
Collecting nvidia-nvjitlink-cu12 (from nvidia-cusolver-
cu12==11.4.5.107->torch<2.3.0,>=2.2.0->facenet_pytorch)
  Using cached nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl
(21.1 MB)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from
jinja2->torch<2.3.0,>=2.2.0->facenet_pytorch) (2.1.5)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-
packages (from sympy->torch<2.3.0,>=2.2.0->facenet_pytorch) (1.3.0)
Installing collected packages: Pillow, nvidia-nvtx-cu12, nvidia-nvjitlink-cu12,
nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cuda-runtime-
cu12, nvidia-cuda-nvrtc-cu12, nvidia-cuda-cupti-cu12, nvidia-cublas-cu12,
nvidia-cusparse-cu12, nvidia-cudnn-cu12, nvidia-cusolver-cu12, facenet_pytorch
  Attempting uninstall: Pillow
    Found existing installation: Pillow 9.4.0
    Uninstalling Pillow-9.4.0:
      Successfully uninstalled Pillow-9.4.0
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
imageio 2.31.6 requires pillow<10.1.0,>=8.3.2, but you have pillow 10.2.0 which
is incompatible.
Successfully installed Pillow-10.2.0 facenet_pytorch-2.6.0 nvidia-cublas-
cu12-12.1.3.1 nvidia-cuda-cupti-cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105
nvidia-cuda-runtime-cu12-12.1.105 nvidia-cudnn-cu12-8.9.2.26 nvidia-cufft-
cu12-11.0.2.54 nvidia-curand-cu12-10.3.2.106 nvidia-cusolver-cu12-11.4.5.107
nvidia-cusparse-cu12-12.1.0.106 nvidia-nccl-cu12-2.19.3 nvidia-nvjitlink-
cu12-12.4.127 nvidia-nvtx-cu12-12.1.105
```

[1]:
```
!pip install grad-cam
```

```
Collecting grad-cam
  Downloading grad-cam-1.5.0.tar.gz (7.8 MB)
                           7.8/7.8 MB
27.2 MB/s eta 0:00:00
  Installing build dependencies … done
```

```
  Getting requirements to build wheel … done
  Preparing metadata (pyproject.toml) … done
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages
(from grad-cam) (1.25.2)
Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages
(from grad-cam) (10.2.0)
Requirement already satisfied: torch>=1.7.1 in /usr/local/lib/python3.10/dist-
packages (from grad-cam) (2.2.1+cu121)
Requirement already satisfied: torchvision>=0.8.2 in
/usr/local/lib/python3.10/dist-packages (from grad-cam) (0.17.1+cu121)
Collecting ttach (from grad-cam)
  Downloading ttach-0.0.3-py3-none-any.whl (9.8 kB)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages
(from grad-cam) (4.66.4)
Requirement already satisfied: opencv-python in /usr/local/lib/python3.10/dist-
packages (from grad-cam) (4.8.0.76)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-
packages (from grad-cam) (3.7.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-
packages (from grad-cam) (1.2.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from torch>=1.7.1->grad-cam) (3.14.0)
Requirement already satisfied: typing-extensions>=4.8.0 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (4.11.0)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages
(from torch>=1.7.1->grad-cam) (1.12)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-
packages (from torch>=1.7.1->grad-cam) (3.3)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
(from torch>=1.7.1->grad-cam) (3.1.4)
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages
(from torch>=1.7.1->grad-cam) (2023.6.0)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.1.105 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (12.1.105)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.1.105 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (12.1.105)
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.1.105 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (12.1.105)
Requirement already satisfied: nvidia-cudnn-cu12==8.9.2.26 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (8.9.2.26)
Requirement already satisfied: nvidia-cublas-cu12==12.1.3.1 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (12.1.3.1)
Requirement already satisfied: nvidia-cufft-cu12==11.0.2.54 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam)
(11.0.2.54)
Requirement already satisfied: nvidia-curand-cu12==10.3.2.106 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam)
(10.3.2.106)
```

```
Requirement already satisfied: nvidia-cusolver-cu12==11.4.5.107 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam)
(11.4.5.107)
Requirement already satisfied: nvidia-cusparse-cu12==12.1.0.106 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam)
(12.1.0.106)
Requirement already satisfied: nvidia-nccl-cu12==2.19.3 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (2.19.3)
Requirement already satisfied: nvidia-nvtx-cu12==12.1.105 in
/usr/local/lib/python3.10/dist-packages (from torch>=1.7.1->grad-cam) (12.1.105)
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-
packages (from torch>=1.7.1->grad-cam) (2.2.0)
Requirement already satisfied: nvidia-nvjitlink-cu12 in
/usr/local/lib/python3.10/dist-packages (from nvidia-cusolver-
cu12==11.4.5.107->torch>=1.7.1->grad-cam) (12.4.127)
Requirement already satisfied: contourpy>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-
packages (from matplotlib->grad-cam) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (1.4.5)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (24.0)
Requirement already satisfied: pyparsing>=2.3.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib->grad-cam) (2.8.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->grad-cam) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn->grad-cam) (1.4.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn->grad-cam) (3.5.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-
packages (from python-dateutil>=2.7->matplotlib->grad-cam) (1.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.7.1->grad-cam)
(2.1.5)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-
packages (from sympy->torch>=1.7.1->grad-cam) (1.3.0)
Building wheels for collected packages: grad-cam
  Building wheel for grad-cam (pyproject.toml) … done
  Created wheel for grad-cam: filename=grad_cam-1.5.0-py3-none-any.whl
size=38071
sha256=c9370c444e5b4dc93c26e0b6711547041d02b9733d2bc808d8bd2c57ef522317
  Stored in directory: /root/.cache/pip/wheels/5b/e5/3d/8548241d5cffe53ad1476c56
```

```
6a61ad9bf09cc61a9430f09726
Successfully built grad-cam
Installing collected packages: ttach, grad-cam
Successfully installed grad-cam-1.5.0 ttach-0.0.3
```

[2]:
```python
import gradio as gr
import torch
import torch.nn.functional as F
from facenet_pytorch import MTCNN, InceptionResnetV1
import numpy as np
from PIL import Image
import cv2
from pytorch_grad_cam import GradCAM
from pytorch_grad_cam.utils.model_targets import ClassifierOutputTarget
from pytorch_grad_cam.utils.image import show_cam_on_image
import warnings
warnings.filterwarnings("ignore")
```

[3]:
```python
## Download and load the model
```

[4]:
```python
DEVICE = 'cuda:0' if torch.cuda.is_available() else 'cpu'

mtcnn = MTCNN(
    select_largest=False,
    post_process=False,
    device=DEVICE
).to(DEVICE).eval()
```

[5]:
```python
model = InceptionResnetV1(
    pretrained="vggface2",
    classify=True,
    num_classes=1,
    device=DEVICE
)
```

```
  0%|          | 0.00/107M [00:00<?, ?B/s]
```

[6]:
```python
!pip install gdown

# Download the model from Google Drive
!gdown --id 1_WJ4f6i0SttNLHdGq8nO6saoF4eJwGyt -O model.pt

# Load the model from the local file system
checkpoint = torch.load("model.pt", map_location=torch.device('cpu'))

# Load the state dictionary into the model
model.load_state_dict(checkpoint['model_state_dict'])
```

```
model.to(DEVICE)
model.eval()
```

```
Requirement already satisfied: gdown in /usr/local/lib/python3.10/dist-packages
(5.1.0)
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.10/dist-
packages (from gdown) (4.12.3)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from gdown) (3.14.0)
Requirement already satisfied: requests[socks] in
/usr/local/lib/python3.10/dist-packages (from gdown) (2.31.0)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages
(from gdown) (4.66.4)
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.10/dist-
packages (from beautifulsoup4->gdown) (2.5)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests[socks]->gdown) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (2024.2.2)
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6 in
/usr/local/lib/python3.10/dist-packages (from requests[socks]->gdown) (1.7.1)
/usr/local/lib/python3.10/dist-packages/gdown/__main__.py:132: FutureWarning:
Option `--id` was deprecated in version 4.3.1 and will be removed in 5.0. You
don't need to pass it anymore to use a file ID.
  warnings.warn(
Downloading…
From (original):
https://drive.google.com/uc?id=1_WJ4f6i0SttNLHdGq8nO6saoF4eJwGyt
From (redirected): https://drive.google.com/uc?id=1_WJ4f6i0SttNLHdGq8nO6saoF4eJw
Gyt&confirm=t&uuid=8d0c2af9-723b-45f7-a514-14745cf6449b
To: /content/model.pt
100% 282M/282M [00:10<00:00, 27.8MB/s]
```

[6]:
```
InceptionResnetV1(
  (conv2d_1a): BasicConv2d(
    (conv): Conv2d(3, 32, kernel_size=(3, 3), stride=(2, 2), bias=False)
    (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
    (relu): ReLU()
  )
  (conv2d_2a): BasicConv2d(
    (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), bias=False)
    (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
```

```
track_running_stats=True)
    (relu): ReLU()
  )
  (conv2d_2b): BasicConv2d(
    (conv): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
bias=False)
    (bn): BatchNorm2d(64, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
    (relu): ReLU()
  )
  (maxpool_3a): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
  (conv2d_3b): BasicConv2d(
    (conv): Conv2d(64, 80, kernel_size=(1, 1), stride=(1, 1), bias=False)
    (bn): BatchNorm2d(80, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
    (relu): ReLU()
  )
  (conv2d_4a): BasicConv2d(
    (conv): Conv2d(80, 192, kernel_size=(3, 3), stride=(1, 1), bias=False)
    (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
    (relu): ReLU()
  )
  (conv2d_4b): BasicConv2d(
    (conv): Conv2d(192, 256, kernel_size=(3, 3), stride=(2, 2), bias=False)
    (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
    (relu): ReLU()
  )
  (repeat_1): Sequential(
    (0): Block35(
      (branch0): BasicConv2d(
        (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
```

```
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (branch2): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(96, 256, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (1): Block35(
      (branch0): BasicConv2d(
        (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
```

```
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (branch2): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(96, 256, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (2): Block35(
      (branch0): BasicConv2d(
        (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
```

```
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (branch2): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(96, 256, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (3): Block35(
      (branch0): BasicConv2d(
        (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
```

```
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (branch2): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(96, 256, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (4): Block35(
      (branch0): BasicConv2d(
        (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
```

```
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (branch2): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(256, 32, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
          (bn): BatchNorm2d(32, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(96, 256, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
  )
  (mixed_6a): Mixed_6a(
    (branch0): BasicConv2d(
      (conv): Conv2d(256, 384, kernel_size=(3, 3), stride=(2, 2), bias=False)
      (bn): BatchNorm2d(384, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU()
    )
    (branch1): Sequential(
      (0): BasicConv2d(
        (conv): Conv2d(256, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (1): BasicConv2d(
```

```
        (conv): Conv2d(192, 192, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (2): BasicConv2d(
        (conv): Conv2d(192, 256, kernel_size=(3, 3), stride=(2, 2), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
    )
    (branch2): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
  )
  (repeat_2): Sequential(
    (0): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
```

```
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (1): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (2): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
```

```
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (3): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
```

```
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (4): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (5): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
```

```
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (6): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
```

```
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (7): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (8): Block17(
```

```
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (9): Block17(
      (branch0): BasicConv2d(
        (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(896, 128, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
```

```
        (1): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(1, 7), stride=(1, 1),
padding=(0, 3), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(128, 128, kernel_size=(7, 1), stride=(1, 1),
padding=(3, 0), bias=False)
          (bn): BatchNorm2d(128, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(256, 896, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
  )
  (mixed_7a): Mixed_7a(
    (branch0): Sequential(
      (0): BasicConv2d(
        (conv): Conv2d(896, 256, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (1): BasicConv2d(
        (conv): Conv2d(256, 384, kernel_size=(3, 3), stride=(2, 2), bias=False)
        (bn): BatchNorm2d(384, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
    )
    (branch1): Sequential(
      (0): BasicConv2d(
        (conv): Conv2d(896, 256, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (1): BasicConv2d(
        (conv): Conv2d(256, 256, kernel_size=(3, 3), stride=(2, 2), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
```

```
    )
    (branch2): Sequential(
      (0): BasicConv2d(
        (conv): Conv2d(896, 256, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (1): BasicConv2d(
        (conv): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1,
1), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (2): BasicConv2d(
        (conv): Conv2d(256, 256, kernel_size=(3, 3), stride=(2, 2), bias=False)
        (bn): BatchNorm2d(256, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
    )
    (branch3): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
  )
  (repeat_3): Sequential(
    (0): Block8(
      (branch0): BasicConv2d(
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1),
padding=(0, 1), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
```

```
        )
        (2): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1),
padding=(1, 0), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (1): Block8(
      (branch0): BasicConv2d(
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1),
padding=(0, 1), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1),
padding=(1, 0), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (2): Block8(
      (branch0): BasicConv2d(
```

```
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1),
padding=(0, 1), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1),
padding=(1, 0), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (3): Block8(
      (branch0): BasicConv2d(
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
```

```
          (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1),
padding=(0, 1), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1),
padding=(1, 0), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
      (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
    (4): Block8(
      (branch0): BasicConv2d(
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (branch1): Sequential(
        (0): BasicConv2d(
          (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1),
bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (1): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1),
padding=(0, 1), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
        (2): BasicConv2d(
          (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1),
padding=(1, 0), bias=False)
          (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
          (relu): ReLU()
        )
      )
```

```
      (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
      (relu): ReLU()
    )
  )
  (block8): Block8(
    (branch0): BasicConv2d(
      (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
      (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
      (relu): ReLU()
    )
    (branch1): Sequential(
      (0): BasicConv2d(
        (conv): Conv2d(1792, 192, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (1): BasicConv2d(
        (conv): Conv2d(192, 192, kernel_size=(1, 3), stride=(1, 1), padding=(0,
1), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
      (2): BasicConv2d(
        (conv): Conv2d(192, 192, kernel_size=(3, 1), stride=(1, 1), padding=(1,
0), bias=False)
        (bn): BatchNorm2d(192, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
        (relu): ReLU()
      )
    )
    (conv2d): Conv2d(384, 1792, kernel_size=(1, 1), stride=(1, 1))
  )
  (avgpool_1a): AdaptiveAvgPool2d(output_size=1)
  (dropout): Dropout(p=0.6, inplace=False)
  (last_linear): Linear(in_features=1792, out_features=512, bias=False)
  (last_bn): BatchNorm1d(512, eps=0.001, momentum=0.1, affine=True,
track_running_stats=True)
  (logits): Linear(in_features=512, out_features=1, bias=True)
)
```

[7]:
```
!export CUDA_VISIBLE_DEVICES=0
```

[8]:
```
def predict(input_image: Image.Image):
    """Predict the label of the input_image"""
```

```
    # Assuming you've already imported necessary libraries and defined DEVICE

    # Face detection and preprocessing
    face = mtcnn(input_image)
    if face is None:
        raise Exception('No face detected')
    face = face.unsqueeze(0)  # Add the batch dimension
    face = F.interpolate(face, size=(256, 256), mode='bilinear',␣
↪align_corners=False)

    # Convert the face into a numpy array for visualization
    prev_face = face.squeeze(0).permute(1, 2, 0).cpu().detach().int().numpy()
    prev_face = prev_face.astype('uint8')

    # Device and data type conversion
    face = face.to(DEVICE)
    face = face.to(torch.float32)
    face = face / 255.0
    face_image_to_plot = face.squeeze(0).permute(1, 2, 0).cpu().detach().int().
↪numpy()

    # Grad-CAM visualization
    target_layers = [model.block8.branch1[-1]]
    use_cuda = torch.cuda.is_available()
    cam = GradCAM(model=model, target_layers=target_layers)
    targets = [ClassifierOutputTarget(0)]

    grayscale_cam = cam(input_tensor=face, targets=targets, eigen_smooth=True)
    grayscale_cam = grayscale_cam[0, :]
    visualization = show_cam_on_image(face_image_to_plot, grayscale_cam,␣
↪use_rgb=True)
    face_with_mask = cv2.addWeighted(prev_face, 1, visualization, 0.5, 0)

    # Classification and confidence scores
    with torch.no_grad():
        output = torch.sigmoid(model(face).squeeze(0))
        prediction = "real" if output.item() < 0.5 else "fake"
        real_prediction = 1 - output.item()
        fake_prediction = output.item()

    confidences = {
        'real': real_prediction,
        'fake': fake_prediction
    }
    return confidences, face_with_mask
```

[11]:
```
from PIL import Image

# Load the input image
input_image_path = "/content/fake_frame_1.png"
input_image = Image.open(input_image_path)

# Call the predict function
confidences, face_with_mask = predict(input_image)

# Extract the results
real_confidence = confidences['real']
fake_confidence = confidences['fake']

print(f" (Real Confidence: {real_confidence:.4f}, Fake Confidence:␣
 ↪{fake_confidence:.4f})")
```

```
 (Real Confidence: 0.0001, Fake Confidence: 0.9999)
```

[10]:
```
interface = gr.Interface(
    fn=predict,
    inputs=[
        gr.Image(label="Input Image", type="pil")
    ],
    outputs=[
        gr.Label(label="Class"),
        gr.Image(label="Face with Explainability", type="pil")
    ],
).launch()
```

```
Setting queue=True in a Colab notebook requires sharing enabled. Setting
`share=True` (you can turn this off by setting `share=False` in `launch()`
explicitly).

Colab notebook detected. To show errors in colab notebook, set debug=True in
launch()
Running on public URL: https://4231e6425ab580de58.gradio.live

This share link expires in 72 hours. For free permanent hosting and GPU
upgrades, run `gradio deploy` from Terminal to deploy to Spaces
(https://huggingface.co/spaces)
```

```
<IPython.core.display.HTML object>
```

[ ]:
```

```