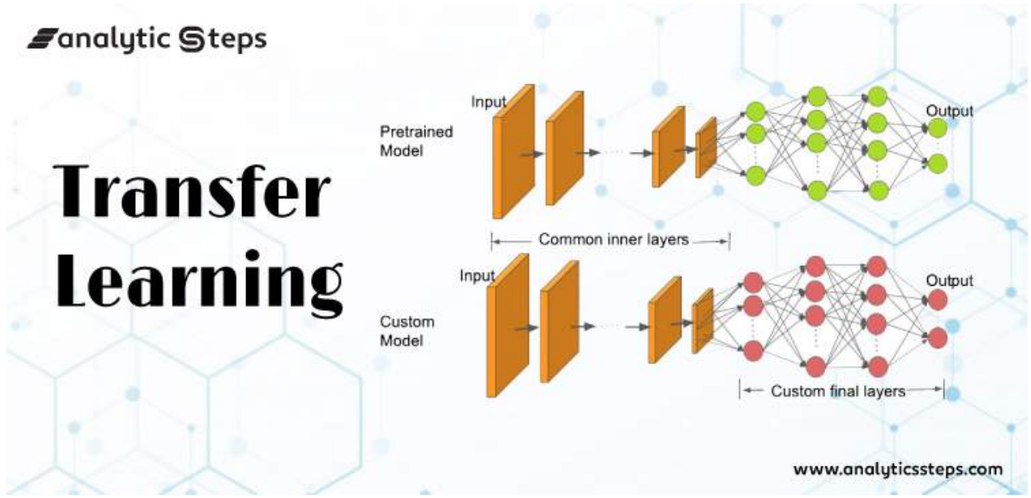



```
import os
from keras.layers import Input,Dense,Flatten
from keras.models import Model
from keras.applications.vgg16 import VGG16
from keras.applications.vgg16 import preprocess_input
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator
import numpy as np
import matplotlib.pyplot as plt
```

```
IMAGE_SIZE=[224,224]
```

```
train_path="/content/drive/MyDrive/maskdata"
test_path="/content/drive/MyDrive/maskdataset"
```

```
vgg=VGG16(input_shape=IMAGE_SIZE+[3],weights='imagenet',include_top=False)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5
58889256/58889256 [==============================] - 0s 0us/step
```

```
for layer in vgg.layers:
  layer.trainable=False
```

```
x=Flatten()(vgg.output)
```

```
prediction=Dense(2,activation='softmax')(x)
```

```
model=Model(inputs=vgg.input,outputs=prediction)
```

```
model.summary()
```

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 224, 224, 3)]     0

 block1_conv1 (Conv2D)       (None, 224, 224, 64)      1792

 block1_conv2 (Conv2D)       (None, 224, 224, 64)      36928

 block1_pool (MaxPooling2D)  (None, 112, 112, 64)      0

 block2_conv1 (Conv2D)       (None, 112, 112, 128)     73856

 block2_conv2 (Conv2D)       (None, 112, 112, 128)     147584

 block2_pool (MaxPooling2D)  (None, 56, 56, 128)       0

 block3_conv1 (Conv2D)       (None, 56, 56, 256)       295168

 block3_conv2 (Conv2D)       (None, 56, 56, 256)       590080

 block3_conv3 (Conv2D)       (None, 56, 56, 256)       590080

 block3_pool (MaxPooling2D)  (None, 28, 28, 256)       0

 block4_conv1 (Conv2D)       (None, 28, 28, 512)       1180160

 block4_conv2 (Conv2D)       (None, 28, 28, 512)       2359808

 block4_conv3 (Conv2D)       (None, 28, 28, 512)       2359808

 block4_pool (MaxPooling2D)  (None, 14, 14, 512)       0

 block5_conv1 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_conv2 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_conv3 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_pool (MaxPooling2D)  (None, 7, 7, 512)         0

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 2)                 50178

=================================================================
Total params: 14,764,866
Trainable params: 50,178
Non-trainable params: 14,714,688
_________________________________________________________________
```

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

```
#image augmentation
train_datagen=ImageDataGenerator(rescale=1/255,
                                 shear_range=0.2, #distort image in x axis
                                 zoom_range=0.2,
                                 horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1/255)
```

```
training_set=train_datagen.flow_from_directory(train_path,
                                               target_size=(224,224),
                                               batch_size=32,
```

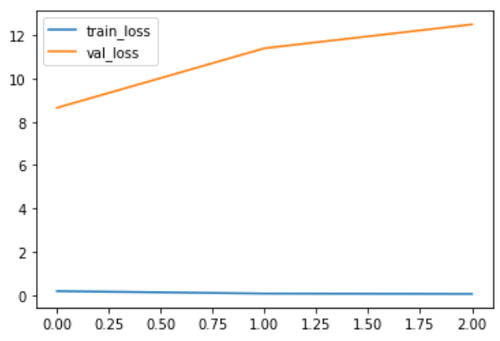
```
                                                class_mode="categorical")
test_set=test_datagen.flow_from_directory(test_path,
                                          target_size=(224,224),
                                          batch_size=32,
                                          class_mode="categorical")
```

```
Found 6955 images belonging to 2 classes.
Found 102 images belonging to 1 classes.
```

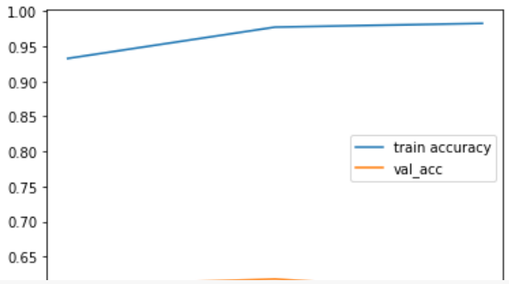
```
#fit model
model1=model.fit_generator(training_set,
                           validation_data=test_set,
                           epochs=3,
                           steps_per_epoch=len(training_set),
                           validation_steps=len(test_set)
                           )
```

```
iput-14-df0ec29b45cc>:2: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
odel.fit_generator(training_set,

==>.....................] - ETA: 1:07:54 - loss: 0.4739 - accuracy: 0.8148/usr/local/lib/python3.8/dist-packages/PIL/Image.py:975: UserWarning: Palette images with Transparency expressed in bytes should be converted to RGB
 warn(
==============================] - 4771s 22s/step - loss: 0.1787 - accuracy: 0.9326 - val_loss: 8.6471 - val_accuracy: 0.6078

==============================] - 4684s 21s/step - loss: 0.0666 - accuracy: 0.9773 - val_loss: 11.3985 - val_accuracy: 0.6176

==============================] - 4555s 21s/step - loss: 0.0507 - accuracy: 0.9826 - val_loss: 12.4999 - val_accuracy: 0.5980
```

```
#loss
plt.plot(model1.history["loss"],label="train_loss")
plt.plot(model1.history["val_loss"],label="val_loss")
plt.legend()
plt.show()
```




```
#accuracy
plt.plot(model1.history["accuracy"],label="train accuracy")
plt.plot(model1.history["val_accuracy"],label="val_acc")
plt.legend()
plt.show()
```



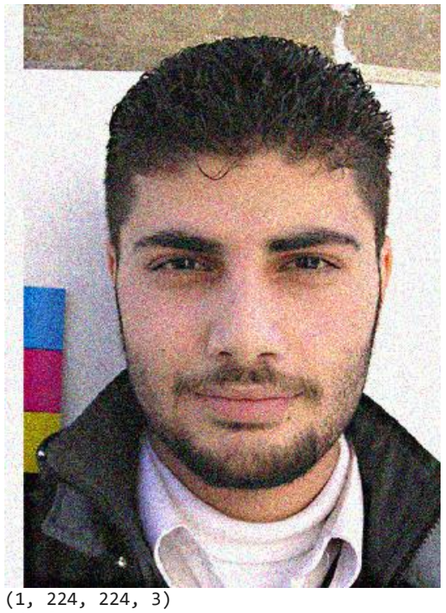

```
import tensorflow as tf
from keras.models import load_model
model.save("facfeatures_new_model.h5")
```

```
import cv2
import numpy as np
from google.colab.patches import cv2_imshow
def predict_new(imgpath,model):
  image=cv2.imread(imgpath)
  cv2_imshow(image)
  image_resized=cv2.resize(image,(224,224))
  image=np.expand_dims(image_resized,axis=0)
  print(image.shape)

  pred=model.predict(image)
  x=np.argmax(pred)
  if x==0:
    print("wearing mask")
  else:
    print("not wearing mask")
```

```
predict_new("/content/drive/MyDrive/maskdata/with_mask/with_mask_1877.jpg",model)
```

```
(1, 224, 224, 3)
1/1 [==============================] - 1s 948ms/step
wearing mask
```

```
predict_new("/content/drive/MyDrive/maskdata/without_mask/without_mask_1222.jpg",model)
```



```
(1, 224, 224, 3)
1/1 [==============================] - 1s 1s/step
not wearing mask
```

Could not connect to the reCAPTCHA service. Please check your internet connection and reload to get a reCAPTCHA challenge.