

```
In [1]: pip install tensorflow
```

```
Requirement already satisfied: tensorflow in c:\users\mehra\anaconda3\lib\site-packages (2.13.0)
Requirement already satisfied: tensorflow-intel==2.13.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow) (2.13.0)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.1.21 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (23.5.26)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (0.4.0)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (3.7.0)
Requirement already satisfied: libclang>=13.0.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (16.0.6)
Requirement already satisfied: numpy<=1.24.3,>=1.22 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.24.3)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (3.3.0)
Requirement already satisfied: packaging in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (23.0)
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 c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (4.24.3)
Requirement already satisfied: setuptools in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (68.0.0)
Requirement already satisfied: six>=1.12.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (2.3.0)
Requirement already satisfied: typing-extensions<4.6.0,>=3.6.6 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (4.5.0)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.14.1)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (1.58.0)
Requirement already satisfied: tensorboard<2.14,>=2.13 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (2.13.0)
Requirement already satisfied: tensorflow-estimator<2.14,>=2.13.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (2.13.0)
Requirement already satisfied: keras<2.14,>=2.13.1 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (2.13.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow-intel==2.13.0->tensorflow) (0.31.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\mehra\anaconda3\lib\site-packages (from astunparse>=1.6.0->tensorflow-intel==2.13.0->tensorflow) (0.38.4)
Requirement already satisfied: google-auth<3,>=1.6.3 in c:\users\mehra\anaconda3\lib\site-packages (from tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2.23.0)
Requirement already satisfied: google-auth-oauthlib<1.1,>=0.5 in c:\users\mehra\anaconda3\lib\site-packages (from tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (1.0.0)
Requirement already satisfied: markdown>=2.6.8 in c:\users\mehra\anaconda3\lib\site-packages (from tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (3.4.1)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2.31.0)
Requirement already satisfied: tensorflow-data-server<0.8.0,>=0.7.0 in c:\users\mehra\anaconda3\lib\site-packages (from tensorflow<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (0.7.1)
Requirement already satisfied: werkzeug>=1.0.1 in c:\users\mehra\anaconda3\lib\site-packages (from tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2.2.3)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in c:\users\mehra\anaconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (5.3.1)
Requirement already satisfied: pyasn1-modules>=0.2.1 in c:\users\mehra\anaconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (0.2.8)
Requirement already satisfied: rsa<5,>=3.1.4 in c:\users\mehra\anaconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (4.9)
Requirement already satisfied: urllib3<2.0 in c:\users\mehra\anaconda3\lib\site-packages (from google-auth<3,>=1.6.3->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (1.26.16)
Requirement already satisfied: requests-oauthlib>=0.7.0 in c:\users\mehra\anaconda3\lib\site-packages (from google-auth-oauthlib<1.1,>=0.5->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\mehra\anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\mehra\anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (3.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\mehra\anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2023.7.22)
Requirement already satisfied: MarkupSafe>=2.1.1 in c:\users\mehra\anaconda3\lib\site-packages (from werkzeug>=1.0.1->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (2.1.1)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in c:\users\mehra\anaconda3\lib\site-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (0.4.8)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\mehra\anaconda3\lib\site-packages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<1.1,>=0.5->tensorboard<2.14,>=2.13->tensorflow-intel==2.13.0->tensorflow) (3.2.2)
Note: you may need to restart the kernel to use updated packages.
```

```
In [2]: pip install keras
```

```
Requirement already satisfied: keras in c:\users\mehra\anaconda3\lib\site-packages (2.13.1)
Note: you may need to restart the kernel to use updated packages.
```

```
In [3]: import keras
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
```

```
In [4]: classifier = Sequential() # Initialise the CNN
```

```
In [5]: # Ist step of Convoltional layer to get feature maps using feature detector  
  
classifier.add(Convolution2D(filters=32, # output feature maps  
                           kernel_size=(3,3), # matrix size for feature detector  
                           input_shape=(64, 64, 3), # input image shape, 3 is for rgb coloured image with 128*128 px  
                           kernel_initializer='he_uniform', # weights distriution  
                           activation='relu')) # activation function
```

```
In [6]: # 2nd Pooling Layer  
classifier.add(MaxPooling2D(pool_size=(2,2)))
```

```
In [7]: #2nd convolutional and pooling layer.  
classifier.add(Convolution2D(filters=32,  
                           kernel_size=(3,3),  
                           kernel_initializer='he_uniform',  
                           activation='relu'))  
classifier.add(MaxPooling2D(pool_size=(2,2)))
```

```
In [8]: # Step 3 - Flattening  
classifier.add(Flatten())
```

```
In [9]: #Step 4 full connection in which input we have from flattening  
  
classifier.add(Dense(units=128,kernel_initializer='glorot_uniform', activation='relu'))  
#step 5 output layer  
classifier.add(Dense(units=1,kernel_initializer='glorot_uniform',activation='sigmoid'))  
# Compiling the CNN  
classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
```

```
In [10]: from keras.preprocessing.image import ImageDataGenerator
```

```
In [11]: #applying all the transformation we want to apply to training data set  
train_datagen = ImageDataGenerator(rescale = 1./255,  
                                   shear_range = 0.2,  
                                   zoom_range = 0.2,  
                                   horizontal_flip = True)
```

```
In [12]: #Rescling the test data set images to use for validation.  
test_datagen= ImageDataGenerator(rescale=1./255)
```

```
In [13]: #Getting My training data ready for validation, so it will read all the data with the px size we gave.  
  
training_set= train_datagen.flow_from_directory(directory= ('D:/Deep Learning/Project/Strawberry Dataset/DATASET/Train_dataset'),  
                                                target_size=(64,64), # As we choose 64*64 for our convolution model  
                                                batch_size=20,  
                                                class_mode='binary' # for 2 class binary  
)
```

Found 2193 images belonging to 2 classes.

```
In [14]: #Getting My test data ready for validation, so it will read all the data with the px size we gave.  
  
test_set= test_datagen.flow_from_directory(directory= 'D:/Deep Learning/Project/Strawberry Dataset/DATASET/Test_dataset',  
                                            target_size=(64,64), # As we choose 64*64 for our convolution model  
                                            batch_size=20,  
                                            class_mode='binary' # for 2 class binary  
)
```

Found 2193 images belonging to 2 classes.

```
In [28]: # Training the model  
history = classifier.fit(training_set, epochs=10, validation_data=test_set)
```

```
# Accessing the history parameters
print(history.params)

Epoch 1/10
110/110 [=====] - 67s 609ms/step - loss: 0.1739 - accuracy: 0.9270 - val_loss: 0.1101 - val_accuracy: 0.9549
Epoch 2/10
110/110 [=====] - 77s 706ms/step - loss: 0.1611 - accuracy: 0.9339 - val_loss: 0.1015 - val_accuracy: 0.9599
Epoch 3/10
110/110 [=====] - 71s 647ms/step - loss: 0.1558 - accuracy: 0.9384 - val_loss: 0.1075 - val_accuracy: 0.9585
Epoch 4/10
110/110 [=====] - 54s 497ms/step - loss: 0.1468 - accuracy: 0.9403 - val_loss: 0.1187 - val_accuracy: 0.9567
Epoch 5/10
110/110 [=====] - 48s 437ms/step - loss: 0.1463 - accuracy: 0.9412 - val_loss: 0.1238 - val_accuracy: 0.9489
Epoch 6/10
110/110 [=====] - 48s 436ms/step - loss: 0.1348 - accuracy: 0.9485 - val_loss: 0.0968 - val_accuracy: 0.9580
Epoch 7/10
110/110 [=====] - 55s 502ms/step - loss: 0.1424 - accuracy: 0.9412 - val_loss: 0.0837 - val_accuracy: 0.9667
Epoch 8/10
110/110 [=====] - 62s 563ms/step - loss: 0.1265 - accuracy: 0.9526 - val_loss: 0.0872 - val_accuracy: 0.9681
Epoch 9/10
110/110 [=====] - 49s 442ms/step - loss: 0.1592 - accuracy: 0.9375 - val_loss: 0.0981 - val_accuracy: 0.9612
Epoch 10/10
110/110 [=====] - 48s 437ms/step - loss: 0.1432 - accuracy: 0.9430 - val_loss: 0.0795 - val_accuracy: 0.9708
{'verbose': 1, 'epochs': 10, 'steps': 110}
```

```
In [184]: import numpy as np
from keras.preprocessing import image
from PIL import Image
from IPython.display import display
test_image = image.load_img("D:/Deep Learning/Project/Strawberry Dataset/DATASET/Test_dataset/Test/gray_mold590.jpg", target_size = (64, 64))
display(test_image)
# Loading the image and converting the pixels into array whcih will be used as input to predict.
test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis = 0)
result = classifier.predict(test_image)
training_set.class_indices
if result[0][0] == 1:
    prediction = 'Healthy Strawberry'
else:
    prediction = 'Unhealthy Strawberry'
print(prediction)
```



```
1/1 [=====] - 0s 31ms/step
Unhealthy Strawberry
```

```
In [185]: scores = classifier.evaluate(test_set)

110/110 [=====] - 20s 184ms/step - loss: 0.0795 - accuracy: 0.9708
```

```
In [186]: scores
```

```
Out[186]: [0.07948241382837296, 0.9708162546157837]
```

```
In [187]: history
```

```
Out[187]: <keras.src.callbacks.History at 0x2148cdf2110>
```

```
In [188]: history.history.keys()

Out[188]: dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

```
In [189]: history.history['loss'][:5] # show Loss for first 5 epochs
```

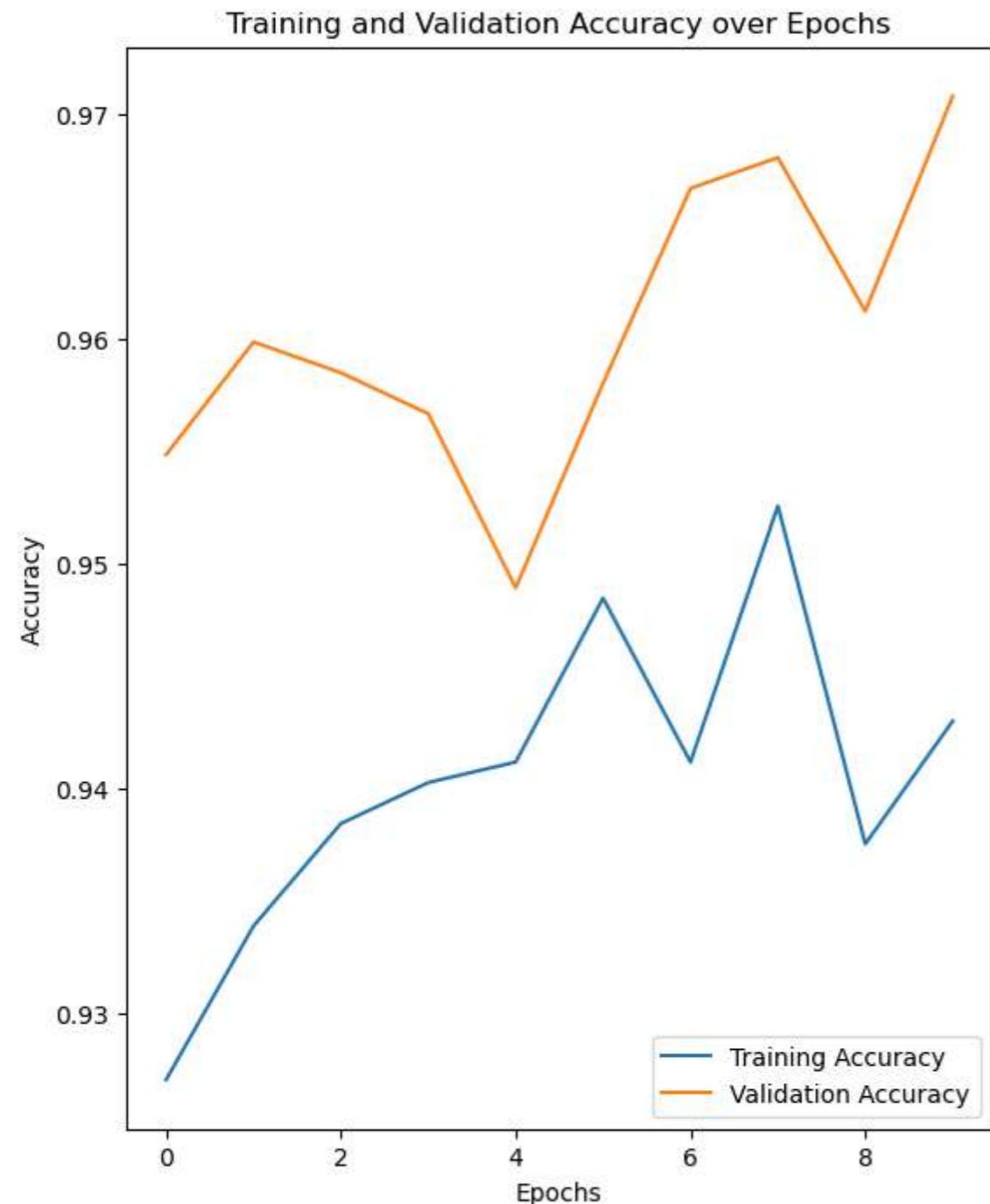
```
Out[189]: [0.17385216057300568,  
 0.1610853672027588,  
 0.1557706743478775,  
 0.14678439497947693,  
 0.1462993621826172]
```

```
In [190]: pip install numpy matplotlib
```

```
Requirement already satisfied: numpy in c:\users\mehra\anaconda3\lib\site-packages (1.24.3)  
Requirement already satisfied: matplotlib in c:\users\mehra\anaconda3\lib\site-packages (3.7.1)  
Requirement already satisfied: contourpy>=1.0.1 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (1.0.5)  
Requirement already satisfied: cycler>=0.10 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (0.11.0)  
Requirement already satisfied: fonttools>=4.22.0 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (4.25.0)  
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (1.4.4)  
Requirement already satisfied: packaging>=20.0 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (23.0)  
Requirement already satisfied: pillow>=6.2.0 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (9.4.0)  
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (3.0.9)  
Requirement already satisfied: python-dateutil>=2.7 in c:\users\mehra\anaconda3\lib\site-packages (from matplotlib) (2.8.2)  
Requirement already satisfied: six>=1.5 in c:\users\mehra\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)  
Note: you may need to restart the kernel to use updated packages.
```

```
In [191]: import matplotlib.pyplot as plt
```

```
In [192]: history = history  
  
# Accessing the training and validation accuracy  
train_acc = history.history['accuracy']  
val_acc = history.history['val_accuracy']  
  
# Number of epochs  
EPOCHS = len(train_acc)  
  
# Plotting the accuracy  
plt.figure(figsize=(14, 8))  
plt.subplot(1, 2, 1)  
plt.plot(range(EPOCHS), train_acc, label='Training Accuracy')  
plt.plot(range(EPOCHS), val_acc, label='Validation Accuracy')  
plt.legend(loc='lower right')  
plt.xlabel('Epochs')  
plt.ylabel('Accuracy')  
plt.title('Training and Validation Accuracy over Epochs')  
plt.show()
```



```
In [193]: classifier.save('strawberry_disease_inception_model.keras')
```

```
In [ ]:
```