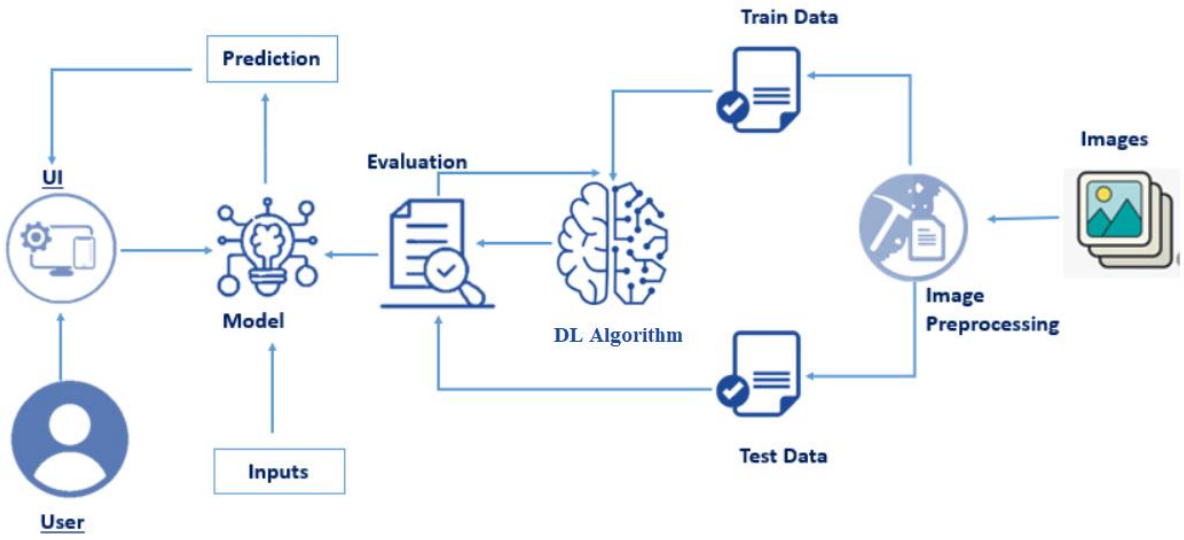
Fertilizers Recommendation System For Disease Prediction

Project Description:

Agriculture is the most important sector in today's life. Most plants are affected by a wide variety of bacterial and fungal diseases. Diseases on plants placed a major constraint on the production and a major threat to food security. Hence, early and accurate identification of plant diseases is essential to ensure high quantity and best quality. In recent years, the number of diseases on plants and the degree of harm caused has increased due to the variation in pathogen varieties, changes in cultivation methods, and inadequate plant protection techniques. An automated system is introduced to identify different diseases on plants by checking the symptoms shown on the leaves of the plant. Deep learning techniques are used to identify the diseases and suggest the precautions that can be taken for those diseases.

Architecture:



Software Requirements:

1. Jupyter Notebook(Anaconda 3)
2. Spyder(Anaconda 3)

Package Requirements:

1. Tensorflow
2. Keras
3. Python

Fruits Disease prediction model Building:

1. **Image Augmentation**

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train = ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range =
0.2,horizontal_flip = True)
test = ImageDataGenerator(rescale = 1./255)
x_train = train.flow_from_directory('E:\\fruit-dataset\\train',target_size =
(128,128),batch_size=8,class_mode = 'categorical')
x_test = test.flow_from_directory('E:\\fruit-dataset\\test',target_size =
(128,128),batch_size=8,class_mode = 'categorical')
x_train.class_indices
```

2. **Neural Network Model**

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten
model = Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(128 ,activation='relu'))
model.add(Dense(64,activation = 'relu'))
```

3. **Model Building**

```
model.add(Dense(6,activation = 'softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
model.summary()
import tensorflow as tf
tf.__version__
model.fit(x_train,steps_per_epoch=5384//8,validation_data=x_test,validation_steps=1686
//8,epochs=5)
```

4. **Download Model**

```
model.save("fruit.h5")
```

5. **Model testing with sample input**

```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model =load_model("fruit.h5")
img = image.load_img("E:\\fruit-dataset\\test\\Apple___healthy\\0a285c8b-1c31-48d4-
89f2-af8b9edc36f6___RS_HL 5759.jpg",target_size=(128,128))
```

```
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)

index=['Apple___Black_rot',
 'Apple___healthy',
 'Corn_(maize)___Northern_Leaf_Blight',
 'Corn_(maize)___healthy',
 'Peach___Bacterial_spot',
 'Peach___healthy']

index[pred[0]]
```

Vegetable Disease prediction model Building:

1. **Image Augmentation**

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train = ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range =
0.2,horizontal_flip = True)
test = ImageDataGenerator(rescale = 1./255)
x_train = train.flow_from_directory('E:\\Veg-dataset\\train_set',target_size =
(128,128),batch_size=8,class_mode = 'categorical')
x_test = test.flow_from_directory('E:\\Veg-dataset\\test_set',target_size =
(128,128),batch_size=8,class_mode = 'categorical')
x_train.class_indices
```

2. **Neural Network Model**

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten
model = Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(128 ,activation='relu'))
model.add(Dense(64,activation = 'relu'))
model.add(Dense(30,activation = 'relu'))
```

3. **Model Building**

```
model.add(Dense(9,activation = 'softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
model.summary()
import tensorflow as tf
```

```
tf.__version__
model.fit(x_train,steps_per_epoch=11386//16,validation_data=x_test,validation_steps=34
16//8,epochs=5)
```

4. **Download Model**

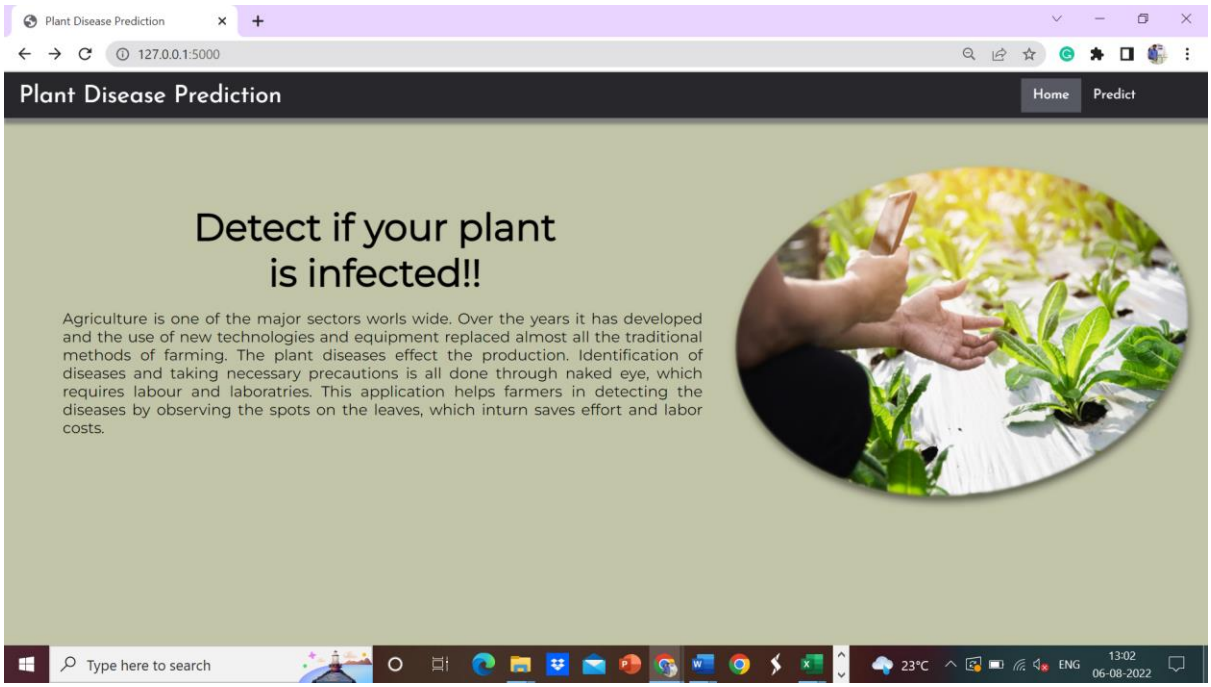
```
model.save("vegetable.h5")
```

5. **Model testing with sample input**

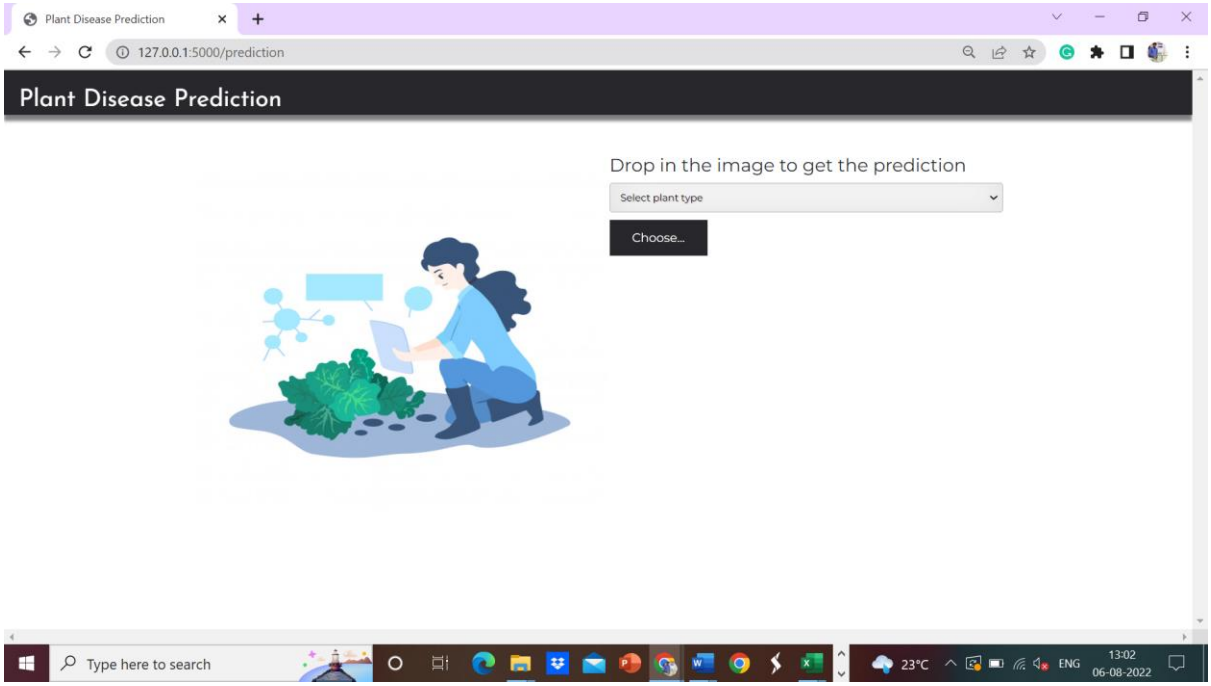
```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model =load_model("vegetable.h5")
img = image.load_img("E:\\Veg-dataset\\test_set\\Tomato___Bacterial_spot\\b8b09399-
338a-4e6d-b05b-30b7a24afc43___GCREC_Bact.Sp 3273.jpg",target_size=(128,128))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x),axis=1)

index=['Pepper,_bell___Bacterial_spot',
 'Pepper,_bell___healthy',
 'Potato___Early_blight',
 'Potato___Late_blight',
 'Potato___healthy',
 'Tomato___Bacterial_spot',
 'Tomato___Late_blight',
 'Tomato___Leaf_Mold',
 'Tomato___Septoria_leaf_spot']

index[pred[0]]
```

OUTPUT SCREENSHOTS

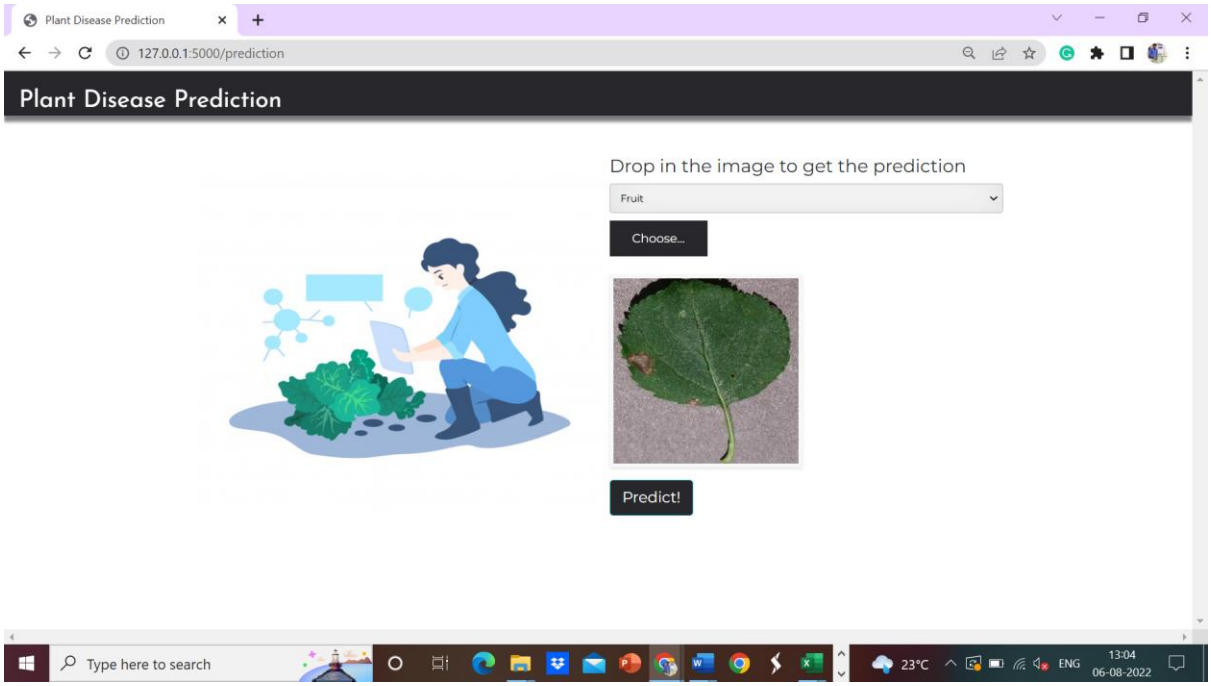
Home Page

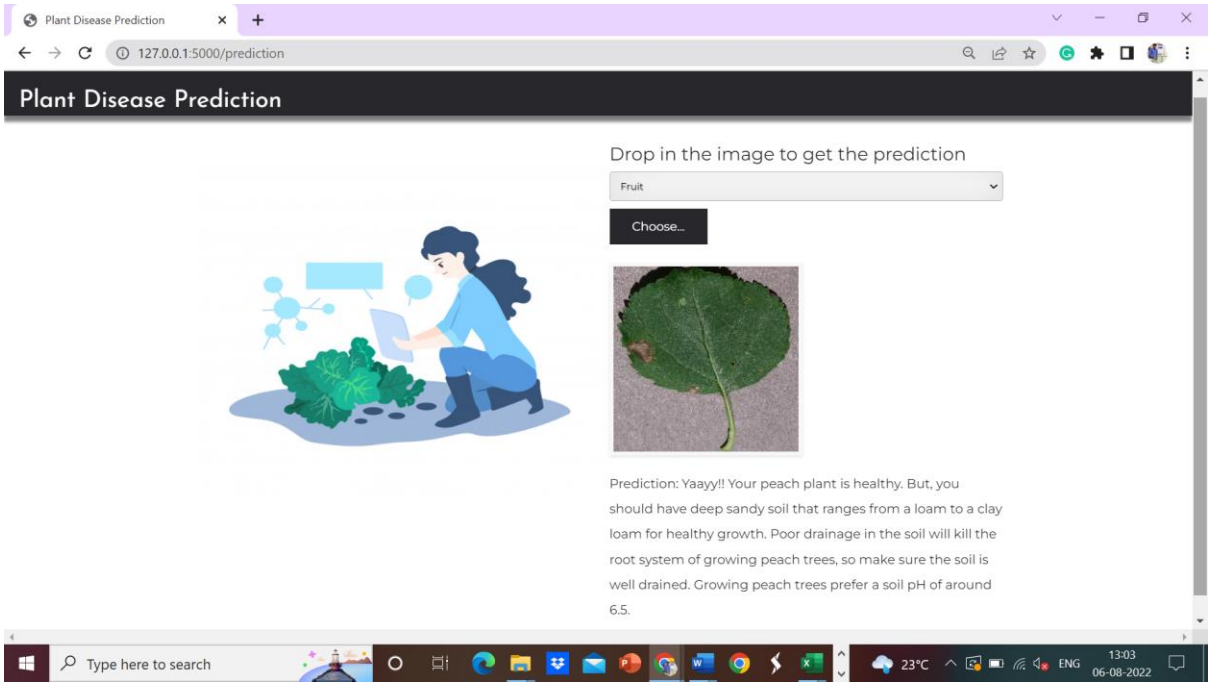


Prediction Interface

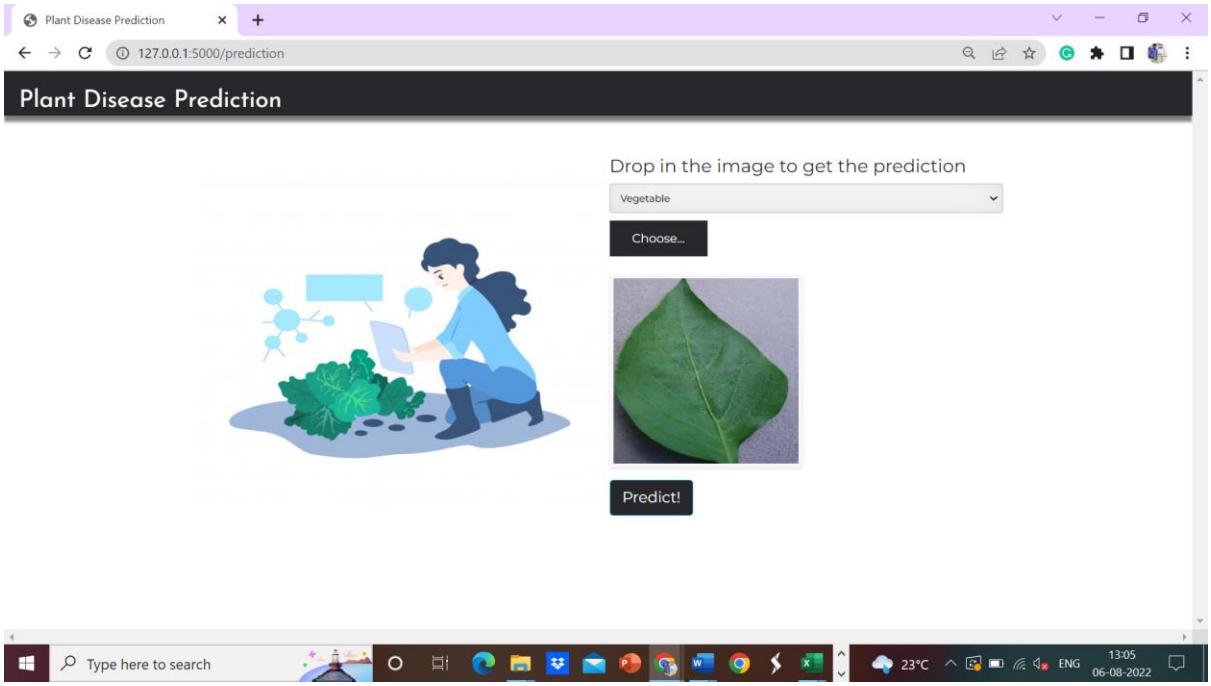


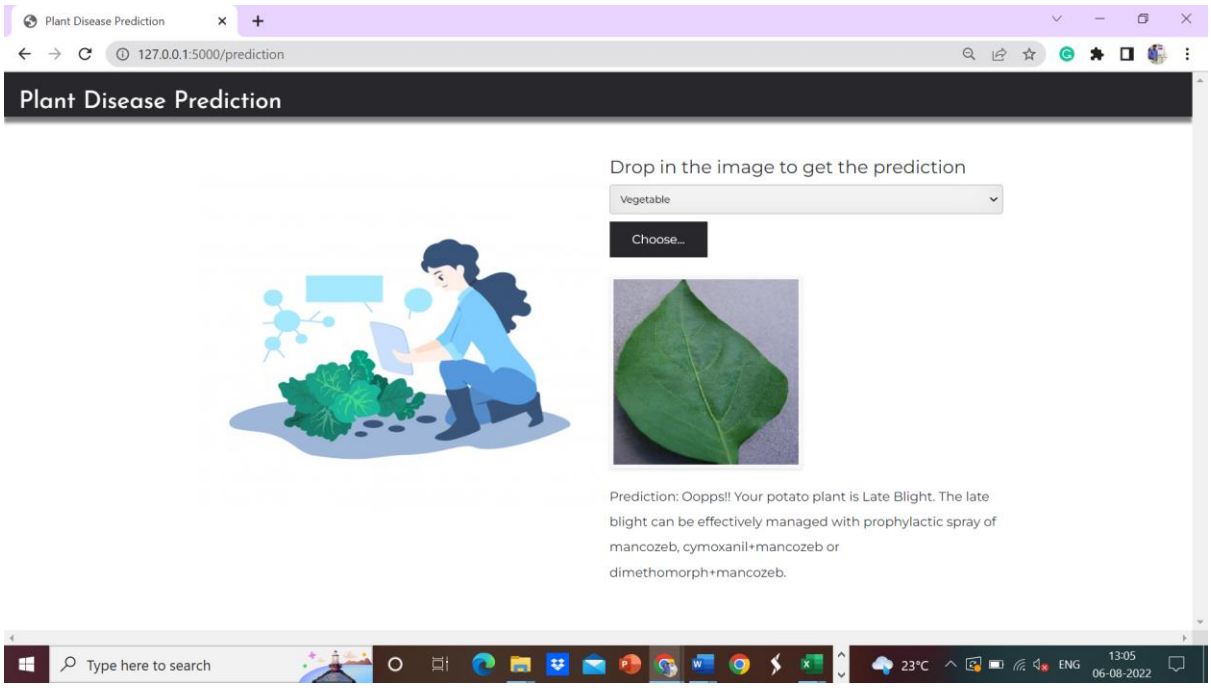
Fruit Leaf Disease Prediction





Vegetable Leaf Disease Prediction





Conclusion

The proposed fruit and vegetable leaf disease prediction model predict the disease and recommend the recommendation to the farmer. The proposed approach will give approximately 97.8 percentage of the accuracy level in terms of classification and prediction.