

```

import numpy as np

import os

from matplotlib import pyplot as plt

import cv2

import random

import pickle

#from google.colab.patches import cv2_imshow

file_list = []

class_list = []

DATADIR = "data"

# All the categories you want your neural network to detect

CATEGORIES = ["Apple___Black_rot",
"Apple___healthy", "Corn_(maize)___healthy", "Corn_(maize)___Northern_Leaf_Blight", "Peach___B
acterial_spot", "Peach___healthy", "Pepper,_bell___Bacterial_spot", "Pepper,_bell___healthy", "Potat
o___Early_blight", "Potato___healthy", "Potato___Late_blight", "Tomato___Bacterial_spot", "Tomato
___Late_blight", "Tomato___Leaf_Mold"]

# The size of the images that your neural network will use

IMG_SIZE = 150


training_data = []


def create_training_data():
    for category in CATEGORIES :
        path = os.path.join(DATADIR, category)
        class_num = CATEGORIES.index(category)
        for img in os.listdir(path):
            try :
                img_array = cv2.imread(os.path.join(path, img), 1)

                #ret,img_array =
cv2.threshold(img_array,170,155,cv2.THRESH_BINARY)

```

```
        #img_array=cv2.equalizeHist(img_array)

        #img_array = cv2.Canny(img_array, threshold1=50, threshold2=10)

        #img_array = cv2.medianBlur(img_array,1)

        new_array = cv2.resize(img_array, (IMG_SIZE, IMG_SIZE))

        training_data.append([new_array, class_num])

    except Exception as e:

        pass
```

```
create_training_data()
```

```
random.shuffle(training_data)
```

```
X = [] #features
```

```
y = [] #labels
```

```
for features, label in training_data:
```

```
    X.append(features)
```

```
    y.append(label)
```

```
X = np.array(X)
```

```
# Creating the files containing all the information about your model
```

```
pickle_out = open("X.pickle", "wb")
```

```
pickle.dump(X, pickle_out)
```

```
pickle_out.close()
```

```
pickle_out = open("y.pickle", "wb")
```

```
pickle.dump(y, pickle_out)
```

```
pickle_out.close()
```

```
pickle_in = open("X.pickle", "rb")
```

```
X = pickle.load(pickle_in)
```