from google.colab import drive

drive.mount('/content/drive')

import os

import numpy as np

from PIL import Image

from skimage.feature import hog

from sklearn.neural\_network import MLPClassifier

from sklearn.metrics import accuracy\_score

from sklearn.preprocessing import LabelEncoder

def load\_images\_from\_folder(folder):

images = []

labels = []

for class\_folder in os.listdir(folder):

class\_path = os.path.join(folder, class\_folder)

if not os.path.isdir(class\_path):

continue

for filename in os.listdir(class\_path):

img\_path = os.path.join(class\_path, filename)

img = Image.open(img\_path).convert('L') # Convert to grayscale

img = img.resize((64, 64)) # Resize to a fixed size

# Extract HOG features

img\_array = np.array(img)

hog\_features = hog(img\_array, pixels\_per\_cell=(8, 8), cells\_per\_block=(2, 2), feature\_vector=True)

images.append(hog\_features)

labels.append(class\_folder)

return images, labels

dataset\_path = '/content/drive/MyDrive/your\_dataset\_folder' # Adjust this path accordingly

train\_folder = os.path.join(dataset\_path, 'train')

test\_folder = os.path.join(dataset\_path, 'test')

X\_train, y\_train = load\_images\_from\_folder(train\_folder)

X\_test, y\_test = load\_images\_from\_folder(test\_folder)

X\_train = np.array(X\_train)

y\_train = np.array(y\_train)

X\_test = np.array(X\_test)

y\_test = np.array(y\_test)

label\_encoder = LabelEncoder()

y\_train = label\_encoder.fit\_transform(y\_train)

y\_test = label\_encoder.transform(y\_test)

# Train an ANN model with Gradient Descent

ann = MLPClassifier(hidden\_layer\_sizes=(100,), max\_iter=300, solver='sgd', learning\_rate\_init=0.01, learning\_rate='adaptive', random\_state=42)

ann.fit(X\_train, y\_train)

y\_pred = ann.predict(X\_test)

accuracy = accuracy\_score(y\_test, y\_pred)

print(f"Accuracy: {accuracy \* 100:.2f}%")