

Final Project - Data Collection

This is a **project group assignment**. No more than 4 individuals.

Due: Friday, October 7 @ 11:59 PM

Problem 1

Create a final project group with no more than 4 individuals.

1. Create your Group Team on GitHub.

- Select a name for your team. This will be used throughout the semester including announcement of contest winners.
- Only 1 person per team should **create** a GitHub Team, and the other members will **join** the team

I encourage you to meet with your teammates regularly.

Problem 2

2.1 Instructions

Each group will collect part of the training dataset. In order to collect the data, you will take pictures of all 10 classes.

Each team member should take a picture of a total of **10 trials** per class, giving a total of 100 images per team member. So, for a group with 4 members, there should be a total of 400 images.

I recommend you to save your files using a **coding system**, e.g. **ID-trial-label**.

- First assign a number from 1 to 4 to each team member, this is the ID. Then, for example, when team member with ID 4 is recording hers/his/their 5th picture of class label 8, the file name should read "4-5-8.jpg".

Create a shared folder (you have access to Google Drive with your UFL account) where you will place all the images. Once completed, everyone can download this folder and have a local copy of the group's data.

2.2 Create your Data for Submission

Create a dictionary with the data input array and target label array.

Before you start, change the `mydir` variable below to the folder directory with your .jpg images.

```
In [22]: import matplotlib.pyplot as plt
```

```
import numpy as np
import os
from PIL import Image # you may need to install Pillow in your environment (https://
import re # you may need to install regex in your environment (https://anaconda.org/

# Folder where all recordings are located
# mydir = 'change-this-path-to-the-local-directory-where-your-images-are-located'
mydir='D:\\Srikantnagfml\\dataset'
```

```
In [20]: # If the code above returns errors, it is likely that you have missing libraries
# Install the missing libraries and then run, restart the kernel and run the imports

# # Installs Pillow
# %conda install -c anaconda pillow

# # Install Regex
# %conda install -c conda-forge regex
```

Now, there are two options to create your data:

Option 1 -- Slowest

Use the code below to plot one file at a time, and manually label each recording.

This code will output and save the data files in the desired format for assignment submission.

```
In [ ]: labels = np.array([])

i=0
for file in os.listdir(mydir):
    # Will only read .jpg files (you can change this to other formats)
    # You may add more readable format with e.g.
    # if file.endswith('.jpg') or file.endswith('.jpeg') or file.endswith('.png'):
    if file.endswith('.jpg'):
        filename = mydir+'/'+file

        # Loads image, converts to grayscale and resizes it to a 300x300 image
        y = np.array(Image.open(filename).convert('RGB').convert('L').resize((300,300)))

        # Resizes 300x300 image to 90,000x1 array
        col_y = y.ravel()[:,np.newaxis]

        # Saves
        if i==0:
            data = col_y
        else:
            data = np.hstack((data, col_y))

        # Plots image
        plt.figure(figsize=(5,5))
        plt.imshow(y, cmap='gray')
        plt.xticks([],plt.yticks([]))
        plt.show();

        # Manually fills in the target label
        l = input('Type the label present in this image (0,1,2,3,4,5,6,7,8 or 9) and
        labels = np.hstack((labels, l))

        i+=1
```

```

if np.sum(labels=='')>0:
    print('-----')
    print('-----NEEDS ATTENTION-----')
    print('-----')
    print('ATTENTION, ', np.sum(labels==''), ' LABEL/S IS/ARE MISSING')

else:
    print('-----')
    print('-----DONE-----')
    print('-----')
    labels = np.array([int(i) for i in labels] )
    print('There are ', data.shape[1], ' images')
    print('There are ', labels.shape[0], ' labels')

# Saves the files to your current directory
np.save('data', data)
np.save('labels', labels)

```

Option 2 -- Fastest

Using the **coding system** from data collection to automatically create and save your data.

The code below will help process the data (using the coding system), and it will output and save the data files in the desired format for assignment submission.

In [23]:

```

labels = np.array([])

i=0
for file in os.listdir(mydir):
    # Will only read .jpg files (you can change this to other formats)
    # You may add more readable format with e.g.
    # if file.endswith('.jpg') or file.endswith('.jpeg') or file.endswith('.png'):
    if file.endswith(".jpg"):
        filename = mydir+'/'+file
        # Loads image, converts to grayscale and resizes it to a 300x300 image
        y = np.array(Image.open(filename).convert('RGB').convert('L').resize((300,300)))

        # Resizes 300x300 image to 90,000x1 array
        col_y = y.ravel()[:,np.newaxis]

        # Saves
        if i==0:
            data = col_y
        else:
            data = np.hstack((data, col_y))

        # Creates labels from filename
        #Find location of label
        match=re.split('[-.]',file)
        labels = np.hstack((labels, int(match[2])))

        i+=1

print('-----')
print('-----DONE-----')
print('-----')
print('There are ', data.shape[1], ' images')
print('There are ', labels.shape[0], ' labels')

# Saves the files to your current directory

```

```
np.save('data', data)
np.save('labels', labels)
```

```
-----
-----DONE-----
-----

There are 400 images
There are 400 labels
```

```
In [24]: import numpy as np
```

```
In [27]: K = np.load('C:\\Users\\kart\\OneDrive\\Desktop\\FML\\final data files\\data.npy')
```

```
In [28]: K
```

```
Out[28]: array([[145, 143, 166, ..., 181, 228, 226],
               [155, 148, 163, ..., 187, 228, 225],
               [160, 150, 159, ..., 194, 227, 225],
               ...,
               [151, 154, 163, ..., 105, 185, 221],
               [154, 152, 164, ..., 108, 186, 220],
               [150, 147, 160, ..., 113, 187, 220]], dtype=uint8)
```

2.3 Gather All Files for Submission

To receive full credit in this question, you should submit to Canvas:

1. Compressed folder (.zip) with the images collected from all team members. (100 recordings per student should be included.)
2. Single file "data.npy" with all the data from the group.
3. Single file "labels.npy" with all the associated target values from the group.

Submit your Solution

Confirm that you've successfully completed the assignment.

add and commit the final version of your work, and push your code/data to your GitHub repository -- **you may run into memory issues. If this happens, disregard this step and only submit the data files to Canvas**

Submit the URL of your GitHub Repository along with all data as your assignment submission on Canvas.