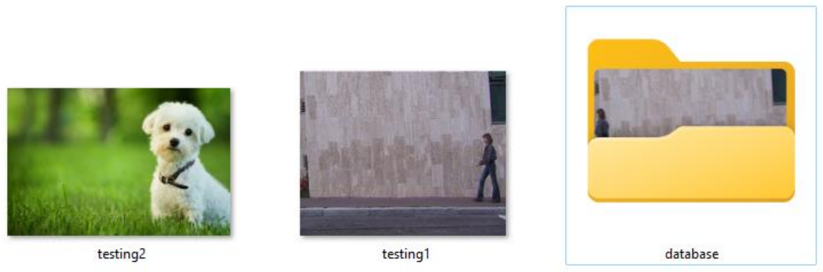
Assignment 2 – CNN Feature Distance based Clustering and Recognition

Cleveland State University

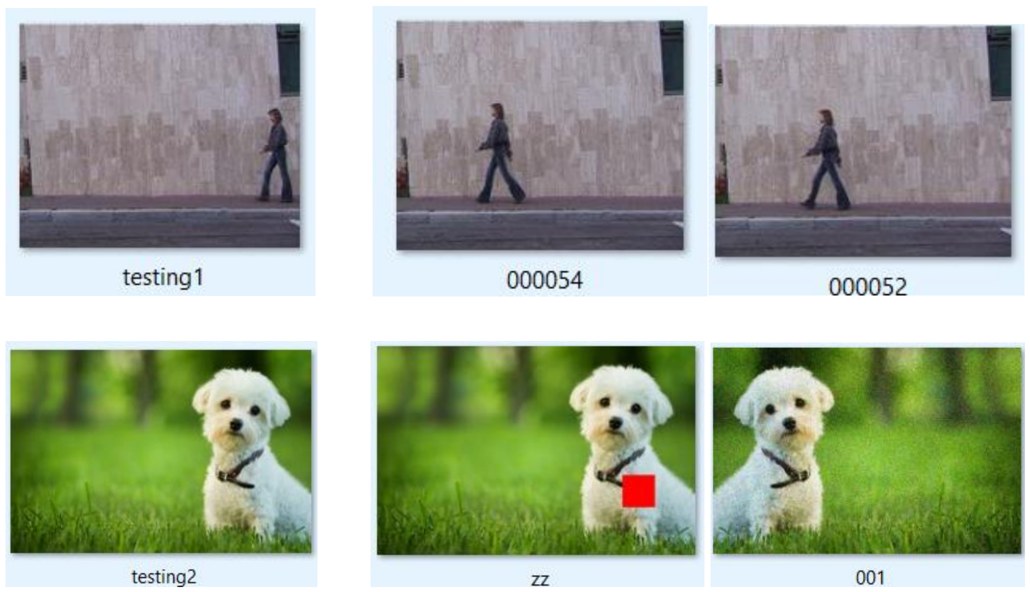
Due: 11:55pm of April 5th, 2024

In this assignment, we will implement the CNN feature distance-based clustering and recognition task. Suppose a database folder has two classes of images, when we have an interested testing image, the recognition task is to recognize which class it belongs to.



Please write Python and PyTorch code to implement the following task:

(1) Download a **ResNet18** pretrained model as the pretrained CNN model X for this assignment. The pretrained CNN model on ImageNet can be downloaded from the internet.
(2) Use X to extract features for each image of the database folder.
(3) Use **kmeans** clustering method to cluster the images in database into k=2 classes. You can use the built-in kmeans function from the sklearn package.
(4) After the kmeans clustering, you can obtain the feature center of each class, defined as C_1 and C_2.
(5) Use X to extract features for “testing1.png” and “testing2.png”, defined as f_1 and f_2.
(6) Compute the L_2 distances between f_1 and C_1, between f_1 and C_2, between f_2 and C_1, between f_2 and C_2.
(7) Based on the feature distance, we can recognize which class “testing1.png” and “testing2.png” belong to.
(8) Finally, please visualize the recognition results. For example, if “testing1.png” belongs to class 2, please show “testing1.png” and two randomly sampled images of class 2. See below.



Blackboard submission only needs 2 files: 1) Your Python3 code, 2) a demo video (<20MB) to show the successful running of your code with two input images ("testing1.png" and "testing2.png").

Requirement:

- You are required to use PyTorch programming to finish this assignment.
- Please use CPU based computation for this assignment to make it simple.

Note:

- Before feeding an image into the pretrained CNN model X, you need to resize it to a uniform size.
- TA will re-run your code, and it is expected to see visualized images to show/list the recognized examples for testing images. TA might change different "images" to test your code, so do not hard code your program.
- Your results should be similar but might be different with the above example results.