Project 2

Computer Vision (CSI4116-01)

Spring, 2022

Due date : 14th May, 23:59:59 KST

Overview

[Recognize faces using Eigenfaces]

You are given two face datasets, “train” and “test”.

You have to run PCA to recognize the faces in the ”test”.

The assignment consists of 3 steps.

1) Select the number of principal components you use for this data.
2) Reconstruct all images in the training set using the number of PCs.
3) Recognize images in the test dataset using a simple nearest neighbor algorithm.

Dataset

Two datasets are provided for this problem

- faces_training : a face database of 39 people
- faces_test : 5 faces to be recognized

Filename : faceXX.pgm / testXX.pgm

- XX indicates the identity number of the image.

data example

Step1: Principal Components Selection

1. Using SVD algorithm, compute principal components of the 'train' dataset.

2. Given a percentage as an input, select the number of principal components you use for this data.
   - e.g., given 0.80, the smallest d such that $\frac{\sum_{i=1}^{d} \lambda_i}{\sum_i \lambda_i} \geq 80\%$

Step1: Principal Components Selection

[Implementation Detail]

1. The percentage of the variance is given as a command line option
2. Output the selected dimension to the ‘output.txt’ file.

Input Example

```
$ python 2022XXXXXX.py 0.95
```

The percentage of the variance is the float value (0,1)
You don’t have to consider the case coming from inappropriate inputs

Output Example

```
##########  STEP 1  ##########
Input Percentage: 0.95
Selected Dimension: XX
```

The percentage given as an input
Selected PCs number

Step2: Image Reconstruction

1. Reconstruct all images in the training set using the selected number of principal components(step1). Then save the reconstructed images.

2. Compute the reconstruction error between original images and reconstructed images.

   Reconstruction error : Mean Squared Error

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - t_i)^2$$

Step2: Image Reconstruction

[Implementation Detail]

1. Save the reconstructed image as a format of ‘2022XXXXXX/faceXX.pgm’ (same with the original file).

2. Output the reconstruction loss of each train image to the ‘output.txt’ file.

Output Sample (face02.pgm)

Original

Reconstructed

Reconstructed iimage size is equal to the original image size

Printing Format (4 decimal places)

```
########## STEP 2 ##########
Reconstruction error
Average : XX.XXXX
01: XX.XXXX
02: XX.XXXX
03: XX.XXXX
04: XX.XXXX
...
```

Step3: Face Recognition

1. Recognize images in the test dataset using a simple nearest neighbor algorithm. We'll use 'l2 distance'. (Compute distance between two vectors.)

   l2 distance(Euclidean distance) :

$$d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

2. Find the closest identity of each image in the test dataset among the identities in the train dataset.

(https://en.wikipedia.org/wiki/Euclidean_distance)

Step3: Face Recognition

[Implementation Detail]

1. Output the identity number for each image in the test dataset to the ‘output.txt’ file.

Printing Format

```
##########  STEP 3  ##########
test01.pgm ==> faceXX.pgm
test02.pgm ==> faceXX.pgm
test03.pgm ==> faceXX.pgm
test04.pgm ==> faceXX.pgm
test05.pgm ==> faceXX.pgm
```

[test image name] ➔ [corresponding train image name]
We have to find the most nearest face among data in train datasets. You don’t have to consider the case where there are several faces which have same l2 distance.

Directions

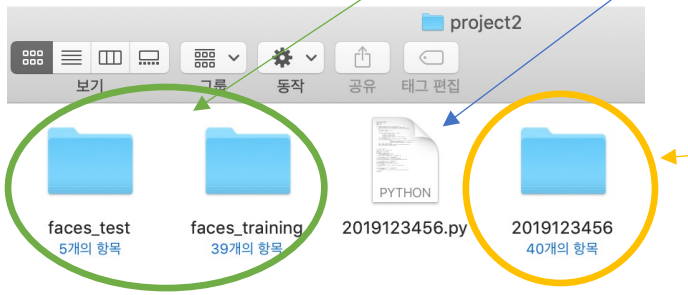
- You cannot use other external libraries except
  - numpy
  - opencv for image reading and writing
  - os, glob, sys

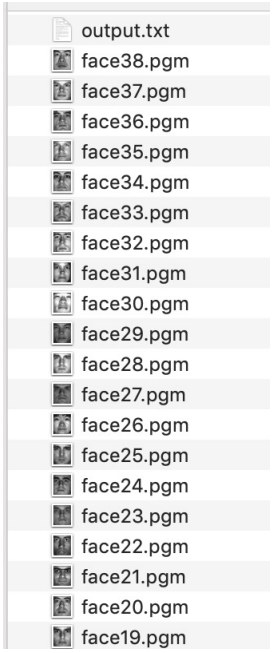
Output

faces_test and faces_training are here!

your code is here!

your output!





- When you **run** your python code, 2022XXXXXX(your student ID) directory is crea ted and inside the directory, ‘output.txt’, ‘faceXX.pgm’ (1~39) must be included.
- You should read images from relative path: ‘./faces_training’ and ‘./faces_test’.
- The overall format of ‘output.txt’ is on the right.
- The output values change according to the input percentage value.
- We will grade scores based on the ‘output.txt’ file and images. So make sure to comply with the given output format.
- **Not following these will result 0.**

output.txt

```
########## STEP 1 ##########
Input Percentage: 0.95
Selected Dimension: XX

########## STEP 2 ##########
Reconstruction error
Average : XX.XXXX
01: XX.XXXX
02: XX.XXXX
03: XX.XXXX
04: XX.XXXX
05: XX.XXXX
06: XX.XXXX
07: XX.XXXX
08: XX.XXXX
09: XX.XXXX
10: XX.XXXX
11: XX.XXXX
12: XX.XXXX
13: XX.XXXX
14: XX.XXXX
15: XX.XXXX
16: XX.XXXX
17: XX.XXXX
18: XX.XXXX
19: XX.XXXX
20: XX.XXXX
21: XX.XXXX
22: XX.XXXX
23: XX.XXXX
24: XX.XXXX
25: XX.XXXX
26: XX.XXXX
27: XX.XXXX
28: XX.XXXX
29: XX.XXXX
30: XX.XXXX
31: XX.XXXX
32: XX.XXXX
33: XX.XXXX
34: XX.XXXX
35: XX.XXXX
36: XX.XXXX
37: XX.XXXX
38: XX.XXXX
39: XX.XXXX

########## STEP 3 ##########
test01.pgm ==> faceXX.pgm
test02.pgm ==> faceXX.pgm
test03.pgm ==> faceXX.pgm
test04.pgm ==> faceXX.pgm
test05.pgm ==> faceXX.pgm
```

Submission

- Deliverables : 2022XXXXXX.py (Your student ID)
  - You should **only submit** *{student_id}.py* file
  - You **should not** include your result files.
  - Your python code must output the required files in the '2022XXXXXX' directory after running your code as mentioned in previous slides.
  - **Not following these will result 0.**

Grading Environment & Directions

- Language : Python
- We will grade your project in Linux(Ubuntu 18.04)
- Python3 (>= 3.7)

- This is an individual project
- You should strictly follow the input/output format
- Never copy code
- You will receive 0 points if you cheat
- If you have any questions, use the Q&A board of LearnUs.

Grading Policy

- Total 100 pts
- Details
  - step1 : 20 pts
  - step2 : 40 pts
  - step3 : 40 pts
  - We will grade implementation scores based on the ‘output.txt’ file and images. Make sure to comply with the given output format. Not following the output format will receive 0.

Due Date

- 14th May, 23:59:59 KST
- We'll receive late submission for 3 days (17th May).
- However, 20% reduction of total score per day is applied.