Project 3

CSE 402 - Biometrics and Pattern Recognition

Instructor: Dr. Arun Ross

Due Date: Dec 10, 2022 (11:00pm ET)

Total Points: 100

Note:

1. **You are permitted to discuss the following questions with others in the class. However, you *must* write up your *own* solutions to these questions. Any indication to the contrary will be considered an act of academic dishonesty. Copying from *any source* constitutes academic dishonesty.**
2. **A neatly typed report is expected (alternately, you can neatly handwrite the report and then scan it). The report, in PDF format, must be uploaded in D2L by December 10, 11:00 pm. Late submissions will not be graded. In your submission, please include the names of individuals you discussed this assignment with and the list of external resources (e.g., websites, other books, articles, etc.) that you used to complete the assignment (if any).**
3. **When solving equations or reducing expressions you must explicitly show every step in your computation and/or include the code that was used to perform the computation. Missing steps or code will lead to a deduction of points. Including the code without including the outputs will lead to deduction of points.**
4. **Code developed as part of this assignment must be (a) included as an appendix to your report or inline with your solution (in the report), and also (b) archived in a single zip file and uploaded in D2L. Including the code without the outputs or including the outputs without the code will result in deduction of points.**
5. **Please submit the report (PDF) and the code (Zip file) as two separate files in D2L.**

---

1. Select one of your own frontal face photo (“selfie"). Crop this photo so that only the face is seen. Next, rescale the cropped face photo to a size of 100 × 100. Finally, convert the cropped and scaled photo from color to grayscale. Use this 100 × 100 grayscale image to do the following.

   (a) [5 points] Apply the following operations to the grayscale image: (a) increase brightness; (b) improve contrast; (c) smoothen the image using a Gaussian Filter. You can use any set of parameter values for each of these operations. Display the original grayscale image (say, F_0) along with the 3 modified images (say, F_B, F_C, F_G) in your report.

(b) [10 points] For each of the 4 images, compute the corresponding LBP image with P=8, R=1. You can set the border pixels of each LBP image to 0. Display the 4 LBP images (say, L_0, L_B, L_C, L_G) in your report.

(c) [5 points] Write a program that compares L_0 with each of L_B, L_C and L_G (so there will be 3 comparison scores). This must be done as follows: compare each pixel in one image with the corresponding pixel in the other image, take the absolute difference between the two, sum all the absolute differences, and then divide this sum by the total number of pixels. Report the 3 scores.

(d) [5 points] Also report the scores obtained when comparing F_0 with each of F_B, F_C and F_G. Use the same procedure as above to compute the comparison scores.

(e) [5 points] Discuss your observations. In particular, does applying the LBP operator reduce the intra-class variation due to changes in pixel intensity?

2. You are given a set of 50 face images pertaining to 10 different subjects (5 images per subject).

(a) [10 points] Compute the eigen-faces (i.e., basis images) using 30 face images (first 3 images per subject). **Show** the mean face as well as the eigen-faces corresponding to the top 50 eigen-values.

(b) [10 points] Using the mean face and the top 50 eigen-faces (i.e., eigen-vectors), compute the eigen-coefficients (i.e., the 50-dimensional feature vector) for each of the 50 images in the dataset, including the 30 face images you had used in 2a.

(c) [10 points] Generate genuine scores and impostor scores by computing the Euclidean distance between the feature vectors of every pair of face images. **Plot** the histograms of genuine and impostor scores in the same graph. Use a different color for each histogram.

(d) [5 points] **Plot** the ROC curve summarizing the matching performance using these scores. You can use the matlab code available here to plot the ROC curve. The code can be invoked as **roc(gen, imp, 'd')**. Here, **gen** and **imp** are the set of genuine and impostor scores, respectively. 'd' denotes that the scores are distance (or dissimilarity) scores.

(e) [10 points] Repeat the above after selecting the top (i) 10, (ii) 20, (iii) 40 eigen-faces. **Plot** the ROC curves for each case. **Comment** on the change in matching performance as you vary the number of eigen-faces used to generate the feature vector.

3. Select 10 of your own **frontal** face photos ("selfies"). Crop each photo so that only the face is seen. Next, rescale each cropped face photo to a size of 30 × 30. Finally, convert the cropped and scaled photos from color to grayscale.

- [5 points] Display the 10 grayscale images.
- [5 points] Compute the eigen-coefficients of the 10 faces using the top 25 eigen-faces (eigen-vectors) computed in 2a.
- [5 points] Compute the genuine scores between every pair of faces (there will be 45 genuine scores).

- [5 points] Plot the histogram of 45 genuine scores on the same graph as 2c. For this part, you can re-plot the histograms from the previous question and use them here.
- [5 points] Comment on the accuracy of the face matcher.

4. **[Bonus Question: 10 Points]** Select 10 of your own **non-frontal** face photos (5 right profile and 5 left profile). Crop each photo so that only the face/head is seen. Next, rescale each cropped face photo to a size of 30×30. Finally, convert the cropped and scaled photos from color to grayscale.

   - Display the 10 grayscale images.
   - Compute the eigen-coefficients of the 10 faces using the top 25 eigen-faces (eigen-vectors) computed in 2a.
   - Compute the genuine scores between every pair of faces (there will be 45 genuine scores).
   - Plot the histogram of 45 genuine scores on the same graph as 2c. For this part, you can re-plot the histograms from the previous question (2c) and use them here.
   - Comment on the accuracy of the face matcher.
