Graphs in Machine Learning
Homework 3 - Large Scale Graph Learning

Mohammed Amine Kheldouni

December 26^{th}, 2018

Online SSL

Question 2.1

In this question, we completed the code for incremental $k-$centers which updates the centroids and the multiplicity V. Please refer to the class *incremental_k_centers* for the implementation.

Question 2.2

We implemented the *online_ssl_compute_solution* by building a $k-NN$ similarity graph, using the centroids and the multiplicity V. Once our Laplacian matrix had been constructed, we performed a hard HFS to predict the last face's label.

Question 2.3

We added some pre-processing steps to our images by performing a resizing (96,96) and a bilateral filter on the grey image. The advantage of using a bilateral filtering is that it is highly effective in noise removal while keeping edges sharp, which helps face recognition.

Another parameter tuning we applied was increasing the variance ($\sigma^2 = 300$) of our $k-NN$ graph so that the weights stay in a common range of values (not too small nor too high). In addition, we set $k = 6$ during the experiments.

Our first experiment is drawn from a binary online classification between my sister's face (Nassima) and mine (Amine). In a second experiment, I manually added two profiles from the previous assignment (TD2) and named them Sarah and Simon. The first figure below presents the four total labels in our dataset. For faces labeled *Sarah* or *Simon*, we had to build a crop function to get the boxes as if those people were in front of the camera.
The results of the online SSL face recognition are shown below.

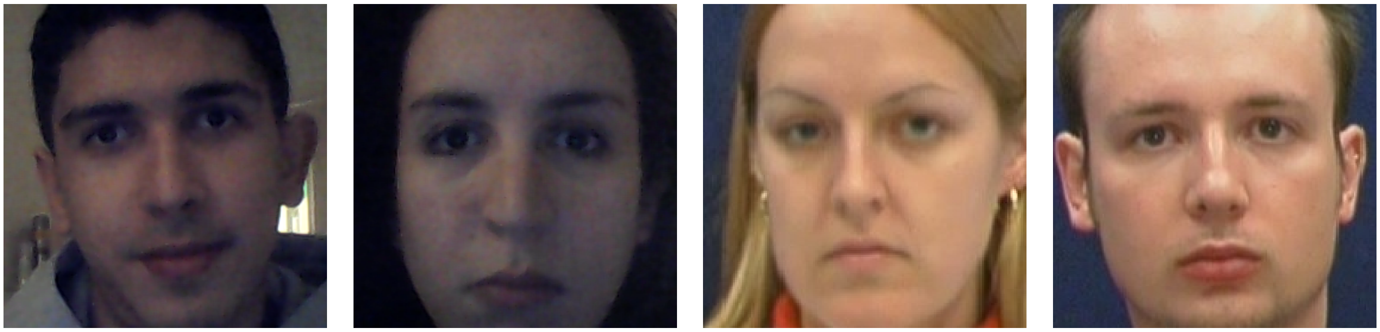

FIGURE 1 – Figure of the profiles for face recognition. From *left* to *right* : Amine, Nassima, Sarah and Simon.

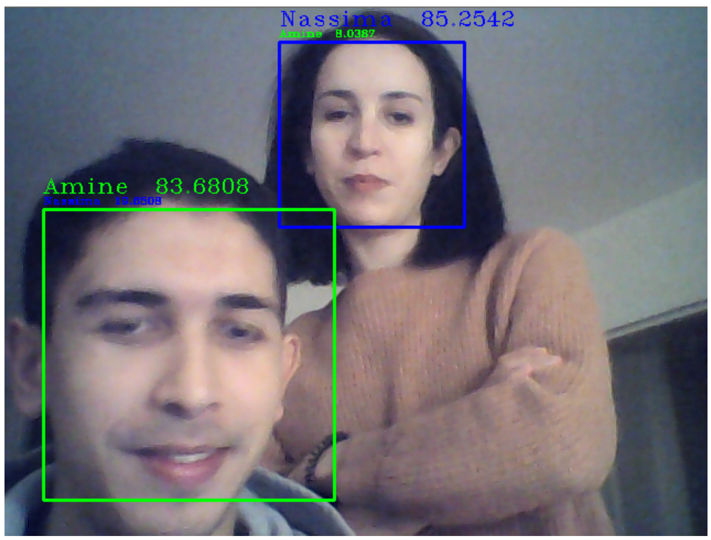

FIGURE 2 – *Nassima* and *Amine*

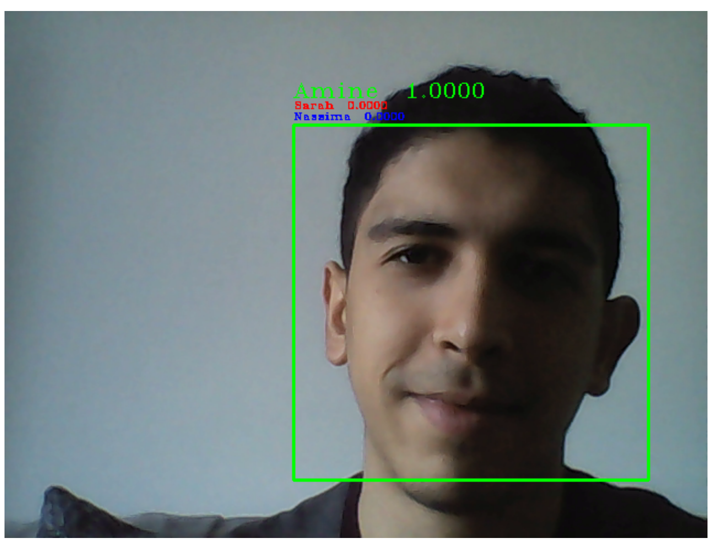

FIGURE 3 – *Amine* vs *Nassima* and *Sarah*

FIGURE 4 – **Left :** *Nassima* vs *Sarah*. **Right :** *Amine* vs *Simon*

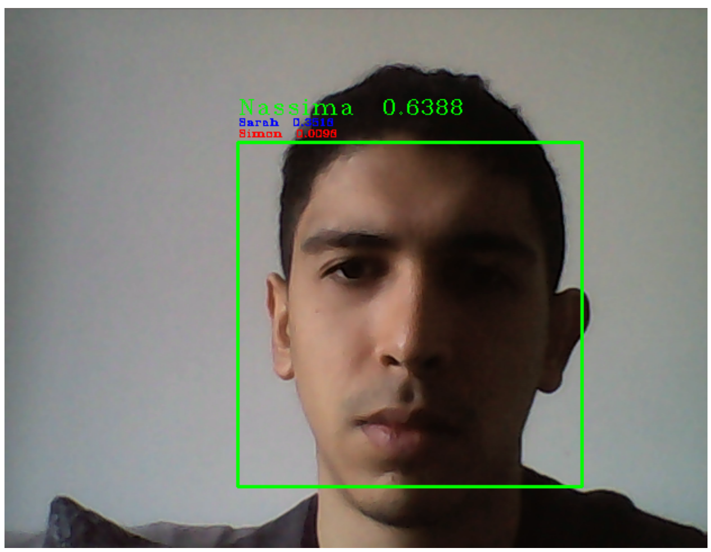

FIGURE 5 – *Amine*'s face labeled as *Nassima* when we do not treat the *unknown* case.

Comments

Unfortunately, in the experiments where my sister appears, the scores tend to explode (but still perform a good and accurate labeling of the captured faces) because of a bad tuning. However, our online SSL module performs really good face recognition. For the last example, when the only given labels are $\{Nassima, Sarah, Simon\}$, I am recognized as being my sister (closest face). To avoid this issue, we handle the *unknown* face when the score is not satisfying.

We noticed that tuning the variance and the k parameter while building the similarity matrix was prominent. A small value of the variance would definitely lower the scoring which implies, as we will see in the next question, a more frequent *unknown* labeling for captured faces (increasing the discrimination).

In addition, adding more images to profiles improved the accuracy of the labeling (varying the brightness, the contrast, the location ...). Therefore, we saved approximately 50 faces in *Amine*'s profile, 70 in *Nassima*'s and a hundred for the two others.

Question 2.4

If an unknown person is captured by the camera, it is labeled at the best scoring label (which maximizes f). However, sometimes we want the model to return *unknown* label whenever this value is too small. Therefore, we operate a thresholding at $\tau = 10^{-5}$ for f. Hence the following results when I capture my face with the camera whereas my profile is not a given input to the online SSL model.

FIGURE 6 – Capturing *Amine*'s face by the camera, whereas the only labels are $\{Sarah, Nassima\}$

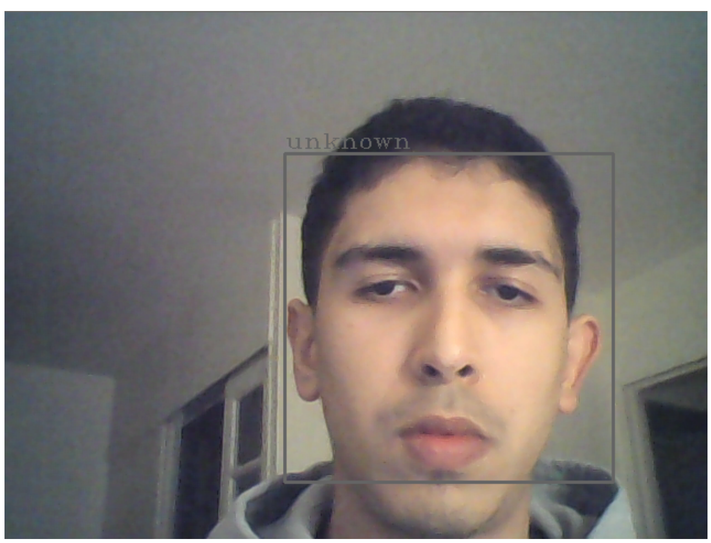

FIGURE 7 – Capturing *Amine*'s face by the camera, whereas the only label is $\{Sarah\}$

Large Scale Label Propagation

Question 3.1

In this new section, we implemented an iterative HFS for large scale data. Our graph is modeled as a sparse matrix in CSC format. We apply multiplications vector-wise to compute f using the Harmonic property that the HFS solution wants to satisfy :

$$f = \frac{\sum_{i \sim j} f(x_j) w_{ij}}{\sum\limits_{i \sim j} w_{ij}}$$

Question 3.2

We could consider a vector-wise regularization where we substract a regularization factor γ (for instance $\gamma = 10^{-2}$) to the i^{th} element of the i^{th} column of the weights matrix W_i. This way, there is a proper analogy between the regularized iterative HFS we described, and the regularized hard HFS we implemented in the previous assignment adding a $\gamma \mathcal{I}$ regularization term to the Laplacian.