Pattern recognition Spring Semester 2019: Final Report

1. Organization of the group

We created and used a WhatsApp group. When someone did something, he notified the other ones in order for them to know what was left to do. When we saw each other at the university, we talked about the project as well to clarify some points, but most of the discussions took place on WhatsApp.

Worked well	Did not work
When someone had an issue, some of the others always had ideas to solve it. We regularly kept each other informed of the progress of the work.	We didn't define specific tasks for each person at the beginning (since it's complicated to have a clear overview of all the tasks and the time they will require at the beginning). Therefore, it was sometimes difficult to know when we had to work since we wanted to leave a bit of work for everyone.

2. Feedback about the various tasks

Description and specificities of the SVM/MLP/CNN task

GitHub: https://github.com/pattern-recognition-mcs/JuLiJoManAnt_SVM_MLP

Our SVM task is divided into 3 parts:

- Create a non-optimized classifier
- Create and optimize a second classifier thanks to cross-validation in order to to find the best parameters
- Classify the test set using both classifiers in order to compare the accuracy difference between a non-optimized and an optimized classifier.

Our MLP relies on Pytorch, so it performs well. We split the training set into 2 parts: 85% of the data as training set and 15% as validation set. Then, we search for the best number of hidden dimensions and the best learning rate, before applying the MLP to the test set.

Finally, our CNN is a relatively conventional CNN, also relying on Pytorch. We search for the best parameters (learning rate), before applying it to the test set.

Worked well	Did not work
Good accuracy of each model CNN and SVM very efficient	Our MLP was a bit slow (particularly with the permuted MNIST set, since we had to test a lot of different parameter combinations)

Description and specificities of the keyword spotting task

GitHub: https://github.com/pattern-recognition-mcs/PatRec19_KWS_Data

The first part crops keywords out of the text pages based on the polygons of word segments that were initially provided. Getting keywords from a page takes between 60 and 110 seconds.

The pictures are then normalized since some words are bigger than others. To do so, we searched for the maximum height and width within the words, and made every other word fit. Since the background of a white scanned page is usually darker, we got rid of that background noise.

Then, values for multiple features were computed for each keyword (upper and lower contour, black and white transitions, number of black pixels, fraction of black pixels).

Finally, we used dynamic time warping to compute the similarity between keywords. We implemented two versions of DTW: one slow version and one faster version that uses NumPy.

Worked well	Did not work
DTW showing great accuracy	DTW pretty slow (even the faster version) Struggle with the Precision and Recall values

Description and specificities of the signature verification task

GitHub: https://github.com/pattern-recognition-mcs/signatureVerification

The signature properties used to do the verification task are the same as recommended : x, y, vx, vy, pressure. Once the properties are extracted from the input file, the mean DTW is computed for all authors between their own genuine signatures.
From there on, the DTW between these mean distances and the unknown signatures was also computed. It only remained to predict to which author those unknown signatures belonged by subtracting the mean distance with the distance of an unknown signature. If this difference is below a given threshold, that means that this signature is a genuine signature of the current author (the one whose mean distance was used). Since the threshold is an hyperparameter, it has been tweaked to find the best accuracy.

Worked well	Did not work
Our algorithm has a correct accuracy	Could be better if more performing properties were used.

3. General thoughts about the group exercise

There were positive and negative parts. The positive ones were that we could easily help each other and not lose as much time as we would have if we were working alone. On the other hand, the groups were pretty large this year. For that reason, each person could only work on small subparts of the projects, which made it harder to get a clear global understanding of how some tasks worked. Also, the organization part was more difficult than if the groups had been smaller.