

Online Signature Biometrics Lab – Report

Nour Abdelhamid

Martín Bernardi

Mary Chris Go

Outline

1. Introduction
2. Data
3. Feature Extraction
4. Performance Evaluation

1. Introduction

The objective of this session is to DEVELOP and EVALUATE an online signature recognition algorithm. According to the theory sessions, signature recognition systems can be divided into two categories:

- **Off-line:** the input is a static image of the signature.
- **On-line:** the signature is acquired using a specific digital sensor which includes the static image and dynamic signals related to the way the signature was done: x,y coordinates, and pressure as a function of time.

Figure 1 shows a block diagram of a typical online signature recognition algorithm where $[x,y,p]$ are the captured signals by the sensor (Cartesian coordinates and pressure), f_t is the feature vector of the query signature to be compared with the f_c feature vector of the signature stored in the database (claimed identity).

In this session we will assume that the data is available (previously acquired) and we will focus on the development of two modules:

- Feature Extraction Module.
- Matcher.

You must complete the tasks proposed in this document and answer the questions included.

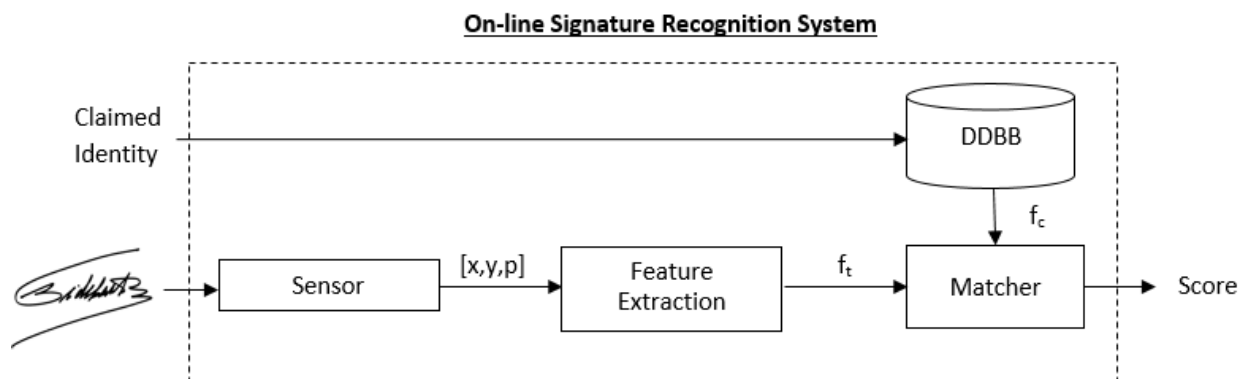


Figure 1. Block Diagram of a typical online signature recognition system

2. Data

For the practice, we will use 50 users from the BiosecurID database. Each of the users has 28 signatures acquired in 4 sessions with a time-lapse of 2 months. From the 28 signatures, 16 are genuine (4 per session) and 12 are forgers (3 per session). In this practice, we will only consider genuine signatures.

Each of the signatures is stored in a .mat file which contains three vectors of the same length with the x, y coordinates, and the pressure as functions of time.

The formatting of the files is uXXXXsYYYY_sgZZZZ.mat:

- XXXX: user number
- YYYY: session number
- ZZZZ: signature number

The GENUINE signatures of each session are those with ZZZZ=[0001,0002,0006,0007].

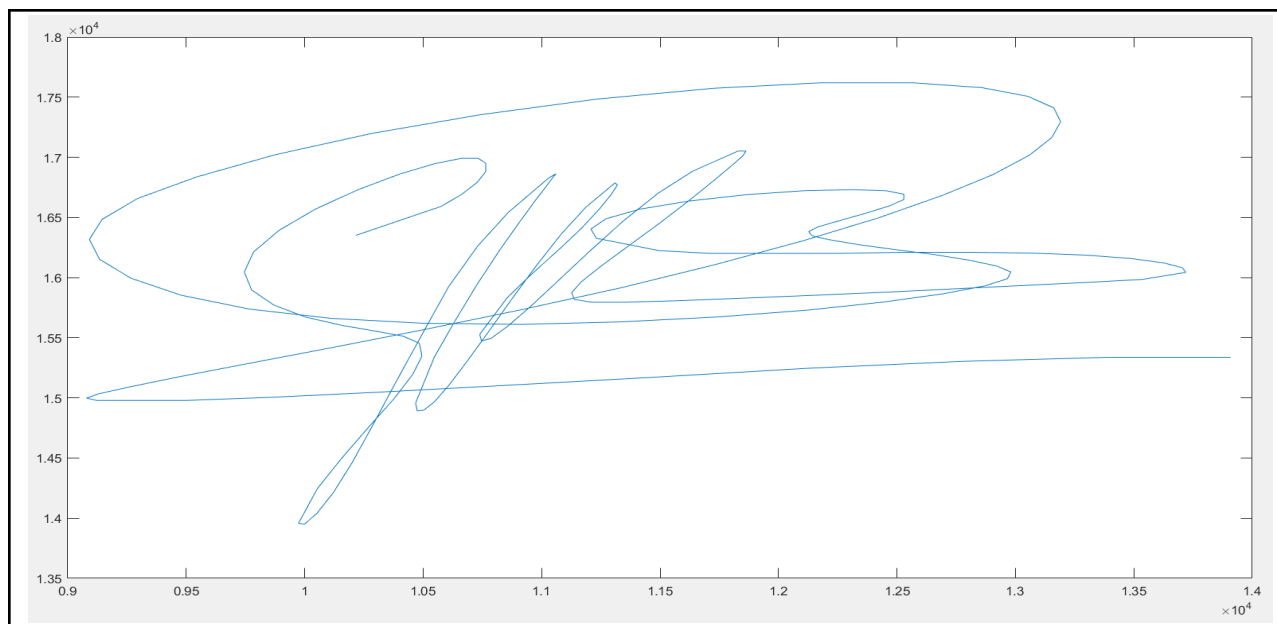
The signatures with ZZZZ=[0003,0004,0005] are the FORGERS and they will NOT be used in this practice.

QUESTION. Choose a signature (from a random user) and show (assuming that the sensor has 200 samples/second acquisition rate):

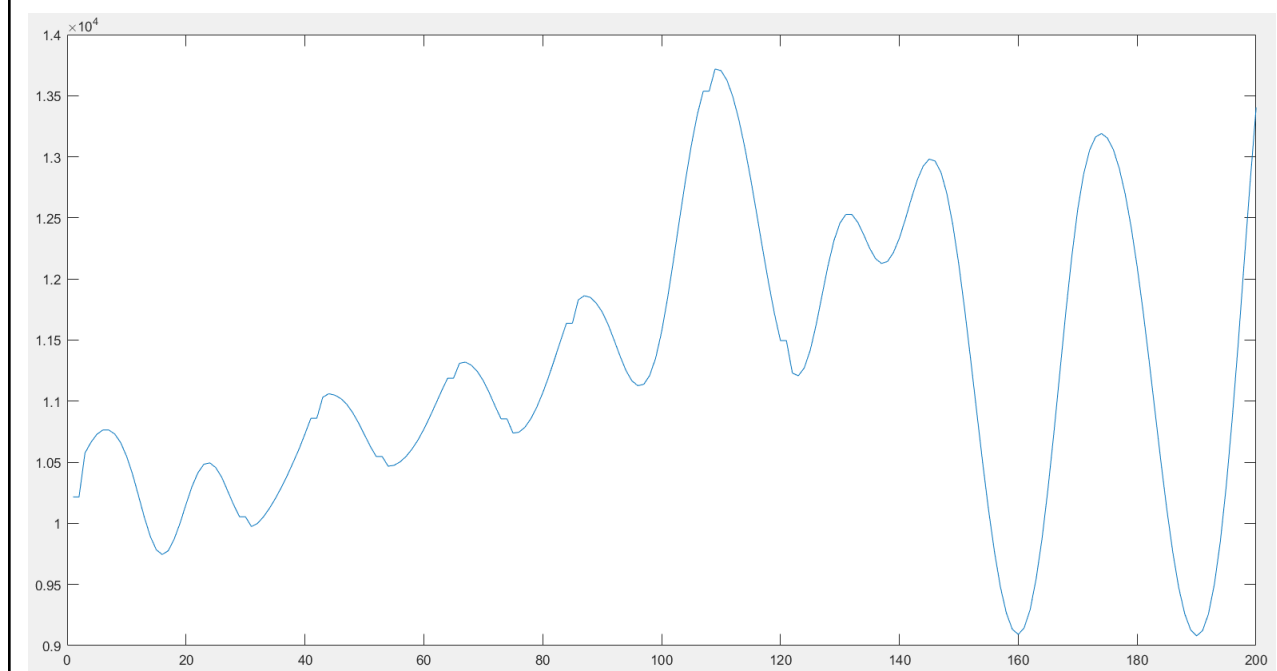
- Signal x as a function of signal y.
- Signal x as a function of time.
- Signal y as a function of time.
- Signal p as a function of time.

Using u1001s0001_sg0002.mat:

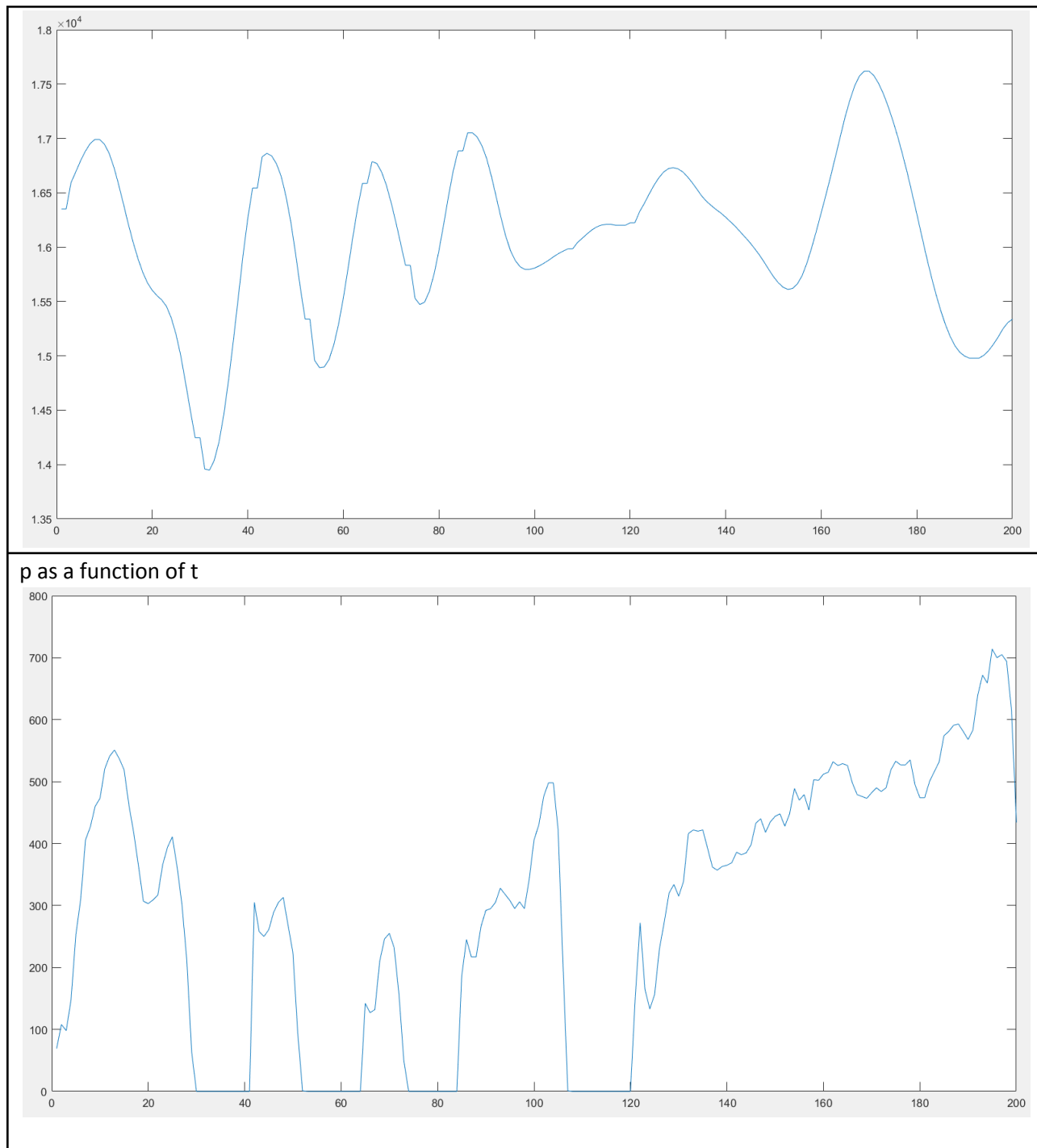
x as a function of y



x as a function of t



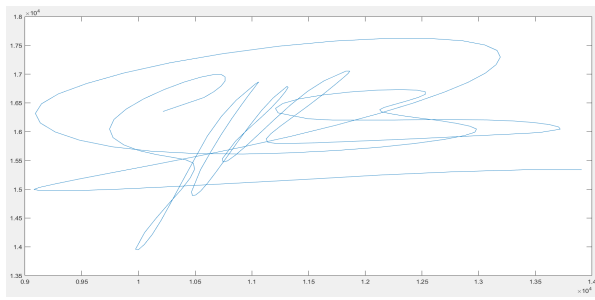
y as a function of t



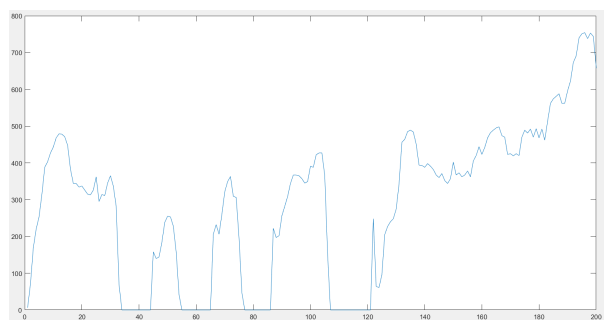
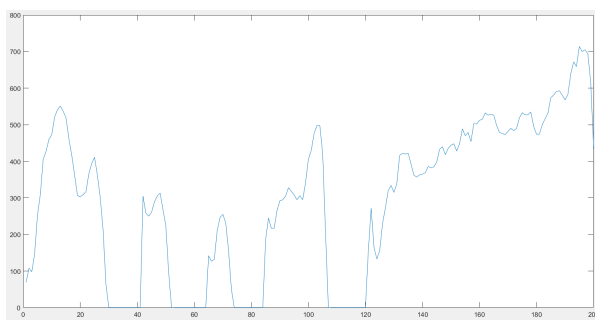
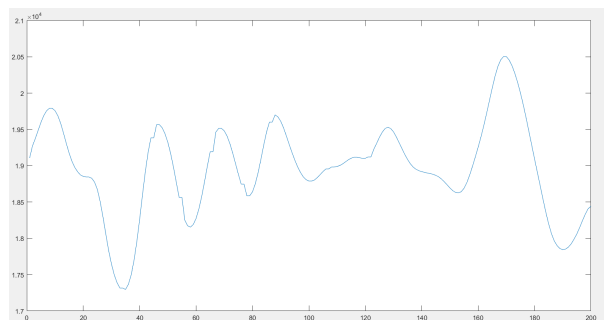
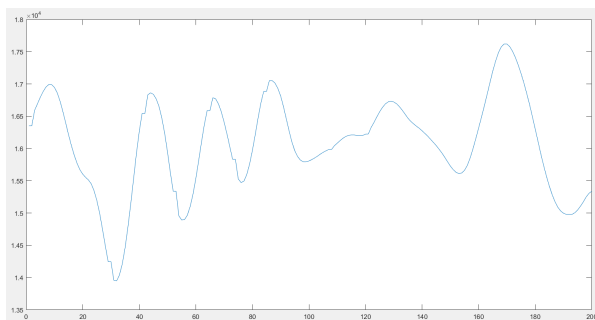
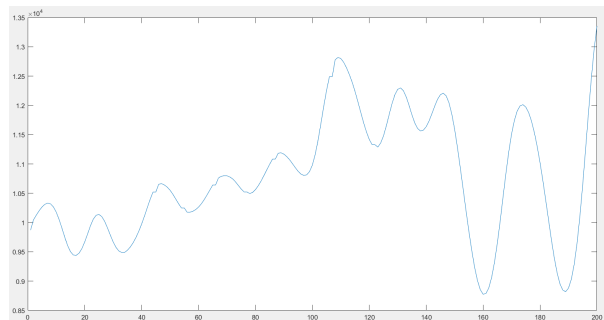
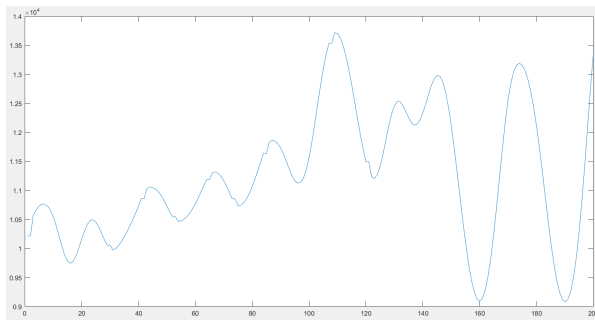
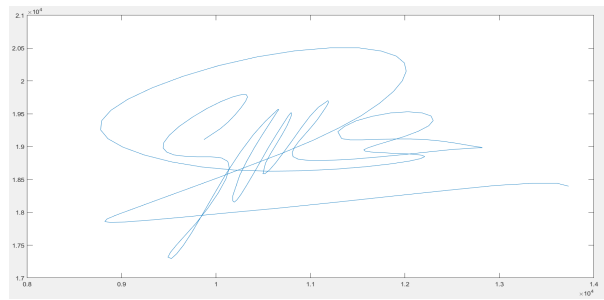
Repeat the task with another signature of the same user.

A comparison of two signatures of the same user:

u1001s0001_sg0002



u1001s0001_sg0001



QUESTION. Are the different signals reasonable? Do they have the same length? Why?

From the comparison of two genuine signatures, the slight differences are reasonable. Since both are from the same user, the graphs follow the same trend but there are few deviations that are visible in some turning points of the curve. The two signatures have a different time sample of 202 and 201, respectively. These deviations are acceptable since people can not write the same signature each time exactly the same way. If these signatures were compared to the forgery samples, we will see big differences not just in the pattern but the duration of the time as well.

3. Feature Extraction

The comparison of signals with different lengths is not trivial. Therefore, we will extract 4 global parameters of each of the signatures. So, each signature will be represented by a feature vector with a fixed size equal to 4. These parameters are:

- Total duration of the signature: T
- Number of *pen-up* (number of times the pen was lifted). It means the number of times (not the number of samples) that p is equal to 0.
- Duration of *pen-down* (signal p is different to 0) T_d divided by the total duration T : T_d/T
- Average pressure in *pen-down* (signal p is different to 0).

You have to develop 4 functions to extract each of the parameters:

- $T = T_{total}(x)$
- $N_{pu} = N_{penups}(p)$
- $T_{pd} = T_{pendown}(p)$
- $P_{pd} = P_{pendown}(p)$.

According to those functions, we will develop a new function with input data (x, y, p) of a given signature and output data the feature vector containing the 4 parameters ($FeatVect = featureExtractor(x, y, p)$).

Based on your function `featureExtractor` you have to develop a program (`ProcessBiosecurID.m`) to extract all the feature vectors from the database and store it in a matrix with 3 dimensions:

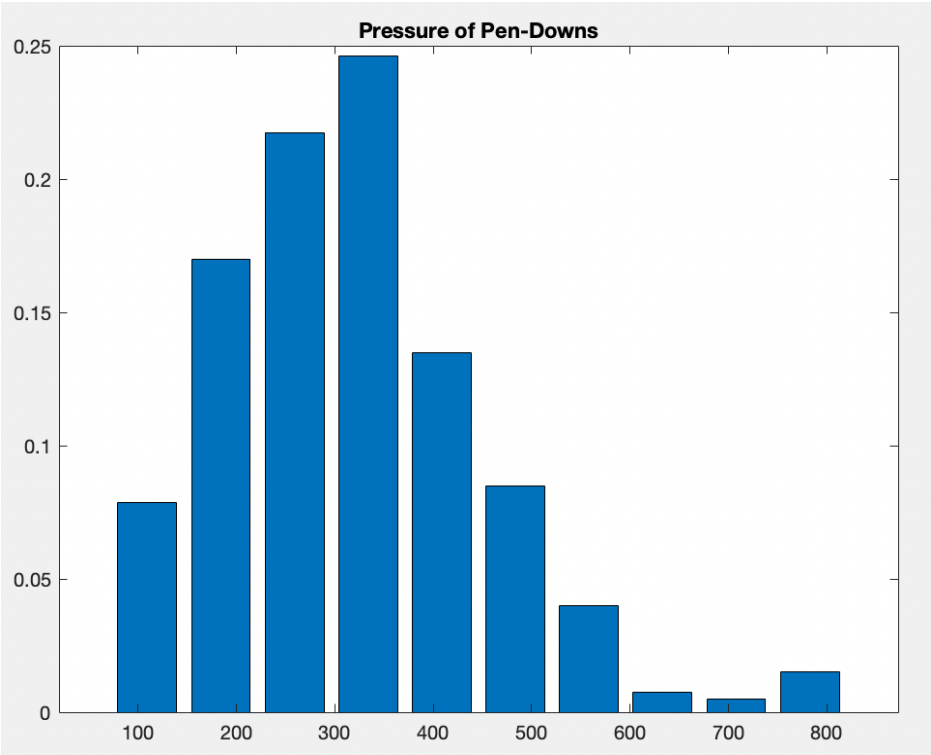
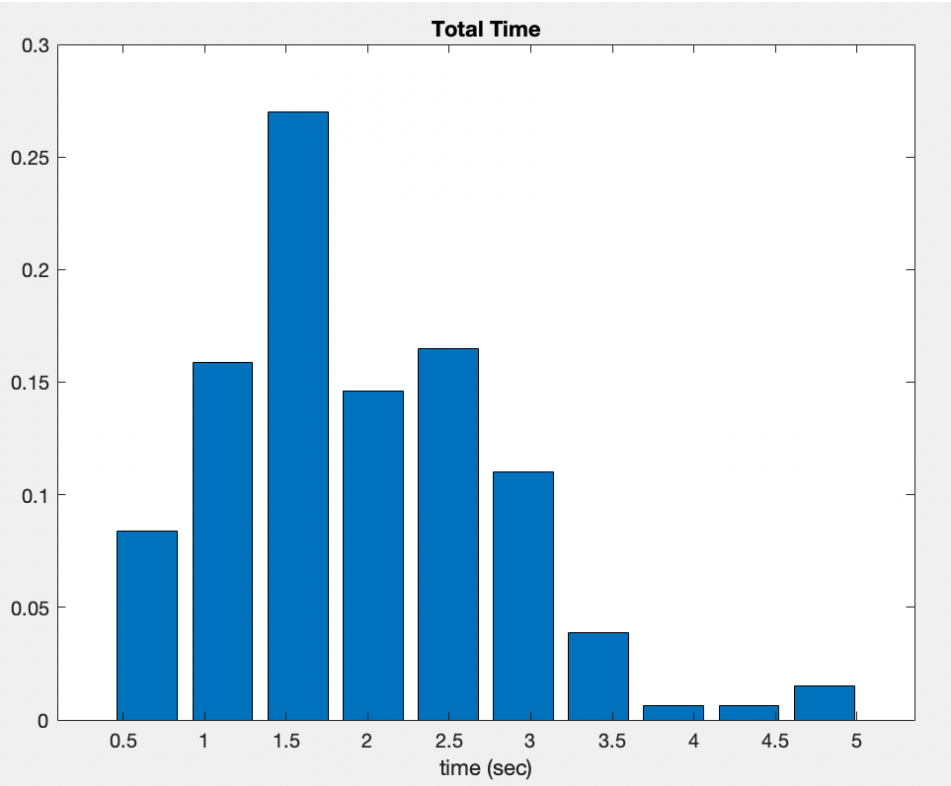
- Dimension 1: number of user (1:50)
- Dimension 2: number of signature (1:16)
- Dimension 3: number of parameter (1:4)

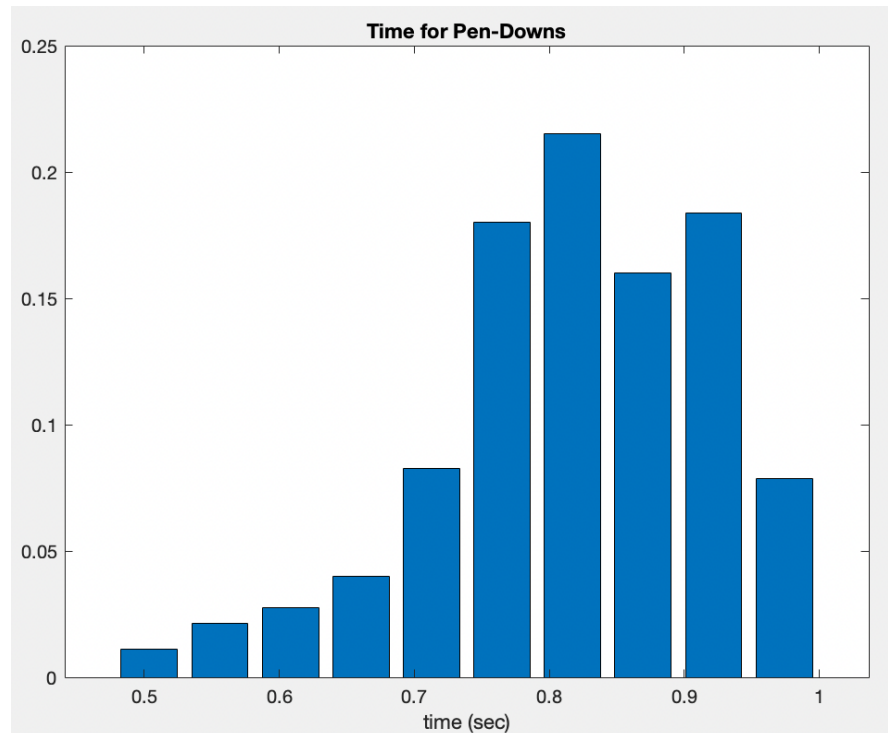
You have to save this matrix into the file `BiosecurIDparameters.mat`

Once you have the file `BiosecurIDparameters.mat`, you have to plot the distributions normalized between 0 and 1 (dividing by the total number of points of the distribution) for each of the 4 parameters.

You can use the Matlab functions `hist.m` and `histc.m`

QUESTION: Plot the 5 distributions.





4. Performance Evaluation

We will evaluate the performance of our system according to the number of signatures N in the enrollment set ($N=1$, $N=4$, and $N=12$).

The similarity score between a query/test signature and the enrollment signatures (signatures in the database) will be the Euclidean distance between feature vectors (vectors with 4 parameters). The final score will be the average score of the N comparisons (comparison between the query/test sample and the N enrollment samples).

You have to develop the function $\text{Score} = \text{Matcher}(\text{test}, \text{Model})$ where:

- Score: is the final score of the comparison.
- test: is the feature vector of the query/test signature (1×4)
- Model: is a matrix containing the feature vectors of the signatures enrolled in the database. Therefore, this matrix contains $N \times 4$ values in which N is the number of signatures enrolled for the claimed identity.

There are two cases to be analyzed:

Genuine Scores: scores obtained when you compare a signature with his real enrolled identity (claimed identity = enrolled identity). So these users should be accepted by the system. For each user you will use N signatures as enrolled samples and the rest for testing:

- For $N=1$ we will have $SG=15$ genuine scores.

- For N=4 we will have SG=12 genuine scores.
- For N=12 we will have SG=4 genuine scores.

For each of the scenarios (N=1,4,12) you have to save all the genuine scores into a matrix (with dimension 50xSG). Each of the three matrixes will be stored into a .mat file with name: GenuineScores_N.mat.

Impostor Scores: scores obtained when you compare a signature with the enrolled samples of other users (claimed identity \neq enrolled identity). So these users should be rejected by the system. In this case, we will compare one signature of each user (the first one) with the models of the rest of the users (excluding the genuine case). Therefore, we will obtain SI=49 impostor scores for each user and each scenario (N=1,4,12).

For each scenario (N=1,4,12) these impostor scores will be saved into a matrix with dimensions 50xSI (50x49). Each of the three matrices will be stored into a .mat file with name: ImpostorScores_N.mat.

Once we obtain the genuine and impostor scores, we will evaluate the performance of our system for each of the three scenarios (N=1,4,12) as a function of: FAR/FRR, EER and DET curves.

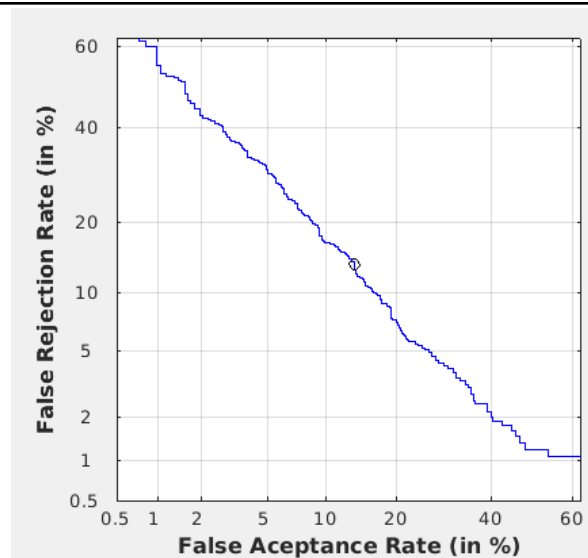
To obtain these performance metrics you will have available the next functions:

[EER]=Eval_Det(GenuineScores, ImpostorScores, 'b')

- EER: the value of the Equal Error Rate (error when FAR and FRR are equal)
- GenuineScores: the scores from target or genuine comparisons. These scores are obtained after applying the following normalization: $\text{GenuineScores} = 1./(\text{GenuineScores_N} + 0.00000001)$
- ImpostorScores: the scores from non target or impostor comparisons. These scores are obtained after applying the following normalization: $\text{ImpostorScores} = 1./(\text{ImpostorScores_N} + 0.00000001)$

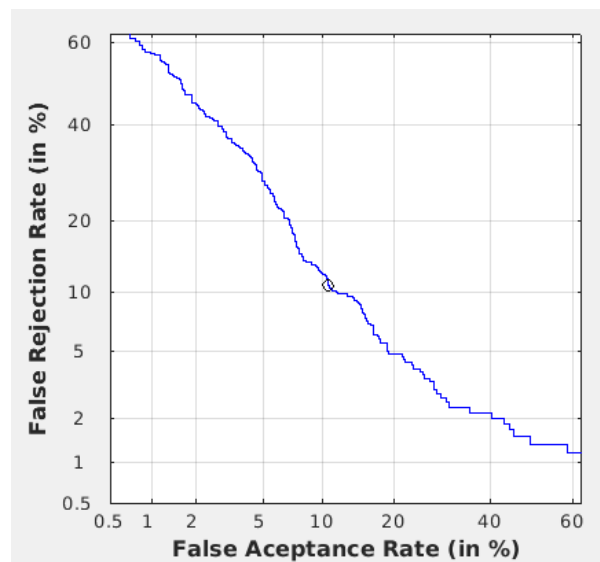
QUESTION. Plot the performance graphics (DET curves) using the genuine and impostors score stored in their respective matrixes (for each of the scenarios N=1,4,12). Indicate the EER value.

N=1



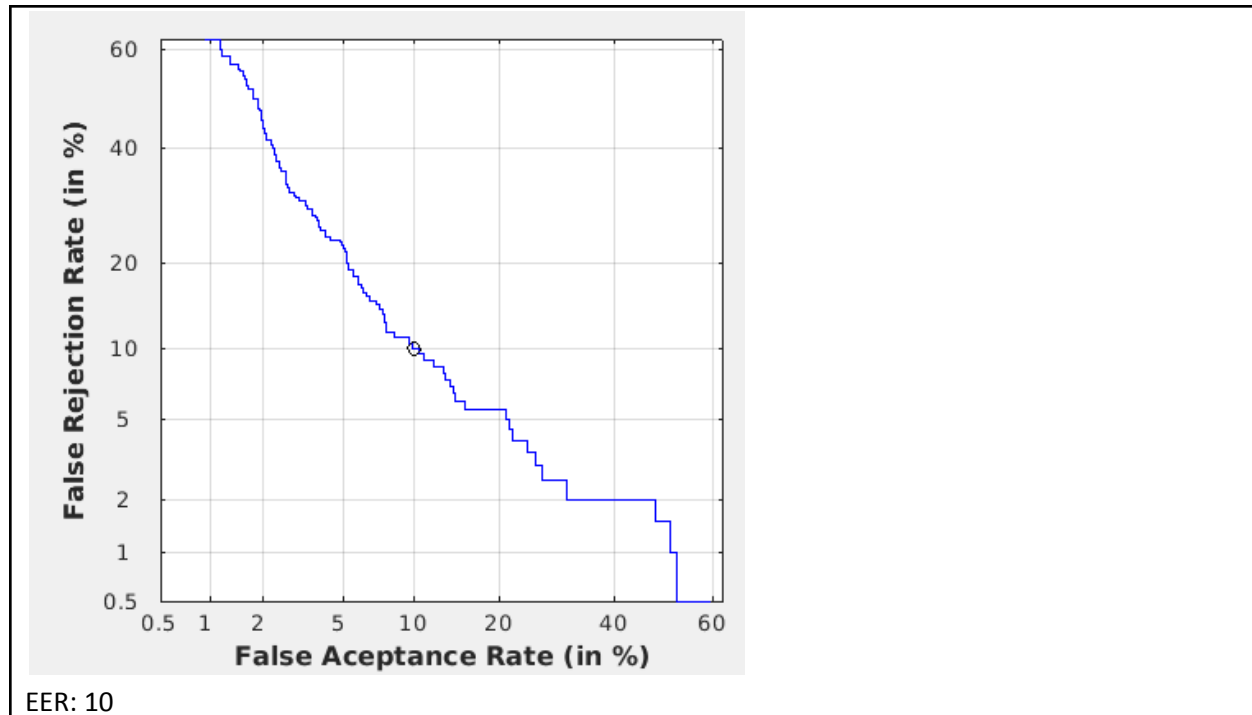
EER: 13.4721

N=4



EER: 10.8027

N=12



QUESTION. According to the results, are they reasonable? What metrics are more illustrative? When do you obtain the best performance?

The results are reasonable. The EER is more illustrative than since it linearizes the relationship between the FAR and FRR which makes a good visualization. The EER is much lower than a random classifier (EER=50%) and it is lower when more signatures are enrolled, although there is not a big difference.

With all the previous exercises done correctly you can obtain a mark up to 6 points out of 10.

Extra work 1: If you want to obtain a mark up to **8 points out of 10** you should complete one of the following points:

- Think and give a reasonable explanation of some additional features you can extract from the signatures. Program them, and repeat the point 4. *Performance Evaluation* in order to prove their improvement in the system performance.
- Obtain a list of the most discriminative features and based on that make combinations of features in order to obtain a better performance.
- Make an evaluation using the skilled forgeries signatures and compared the results with the random forgeries.

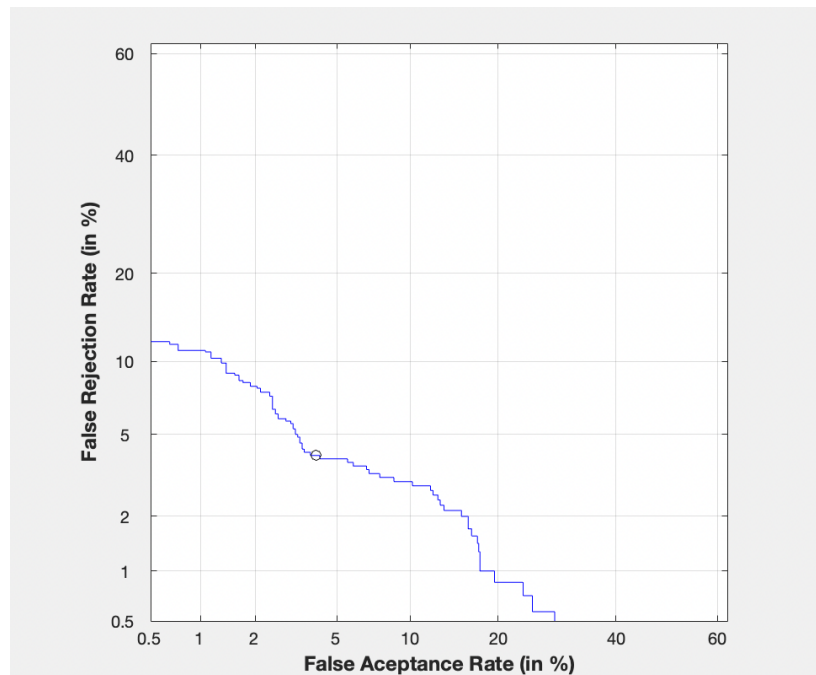
Extra work 2: If you want to obtain a mark up to **10 points out of 10** you should do the following (directly, without doing the Extra work 1):

- Develop an online signature recognition system based on local features (time functions) and Dynamic Time Warping for the Matcher. Repeat the same experimental protocol followed in the practice but using this new signature recognition system.
- You should use the following local features (time functions): x, y, pressure, (and their corresponding first and second derivative).
- You can use the DTW matcher available in Matlab. Take into account that it only allows comparing time functions of different lengths 1 to 1, i.e., x1 vs x2, y1 vs y2, etc. Therefore, you should compare time functions 1 to 1 and finally obtain the average between all-time functions in order to obtain the final score of the comparison between two signatures.
- The equation to obtain the score of the 1vs1 time function comparison is $score = e^{-D/k}$, where D is the minimum accumulative distance obtain after using DTW in Matlab, and K is the number of aligned time samples.

5. Extra Work 2

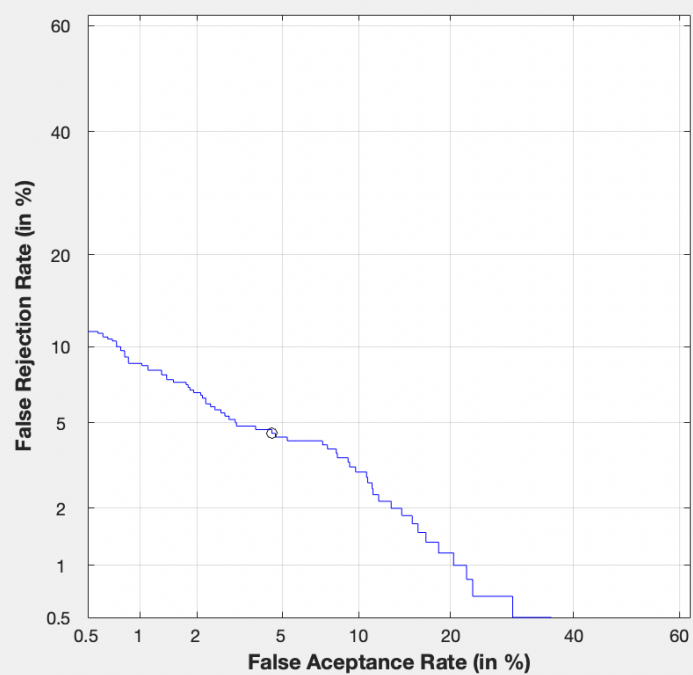
By using the DTW matcher available in Matlab, we managed to retrieve an EER of 4% for all $N_s = 1, 4, 12$. Below are the graphs for each Evaluation.

N=1



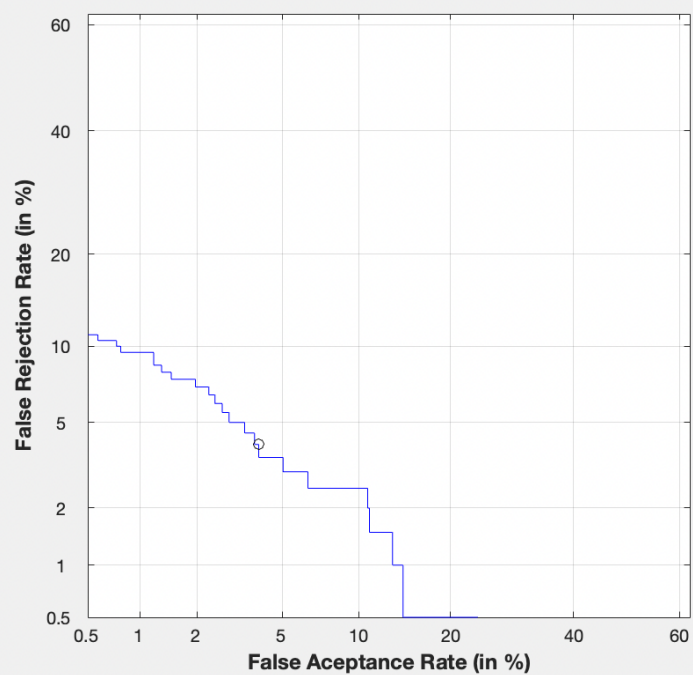
EER: 4

N=4



EER: 4.5

N=12



EER: 4

6. Extra Work 1.1

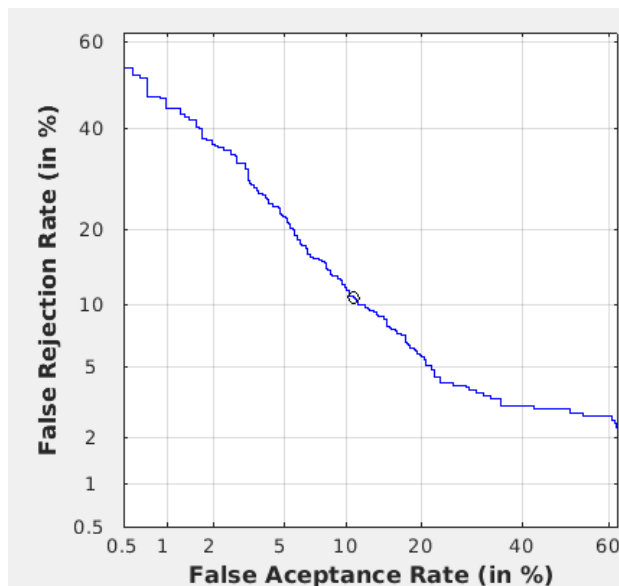
The features used so far are fairly basic, in the sense that the actual shape of the signature is not used when comparing the samples. To improve the performance we decided to incorporate spatial features to our model.

Proposed features:

- Count the number of peaks of the $y(t)$ and $x(t)$ functions to create two additional features. This means that the number of times the pen was moved left/right and up/down is taken into account. As noise can generate additional peaks, the signals are filtered with a running average before peak detection.
- As a measure of the general size of the signature, the standard deviation of the $y(t)$ and $x(t)$ are used as two more features.
- Another measure used is the average speed of the pen movement. For this, the signals $y(t)$ and $x(t)$ are differentiated against time and added after squaring to obtain a signal that represents speed over time. To avoid noise in the speed measurements, a lowpass filter was used before differentiation.

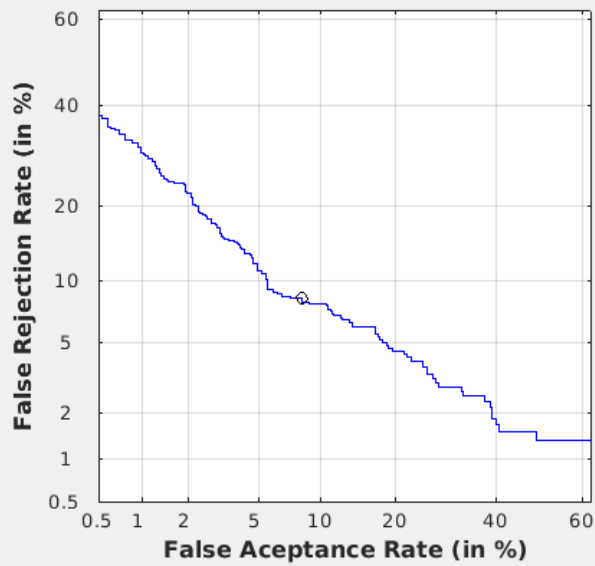
As a result, the accuracy of the classifications improved significantly. Also tests were done when some of the additional features were not calculated and we observed that all of them contribute to improve the model, although the average speed of pen movement does give significant differences.

N=1



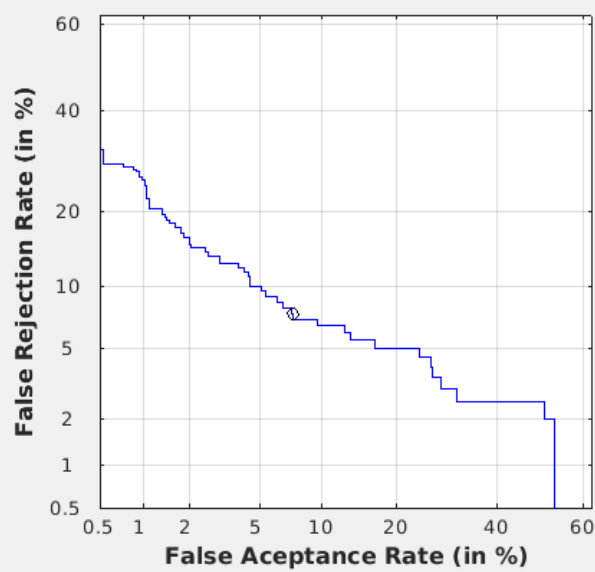
EER: 10.8000

N=4



EER: 8.3333

N=12



EER: 7.5000