Baré Alexandre (r0912072)

Assignment 3 : Face Recognition

Biometrics System Concepts

Katholiek Universiteit Leuven
Faculty of Engineering Science
Academic Year 2021-2022

1 Tasks of Choice

I opted for the task 3 (2pt.) and 4 (1pt.).
Task 3 will be described in the last section of the report.

For task 4, I chose to experiment with a different distance calculation layer in the siamese deep learning model. The initial distance layer was based on the euclidean distance and I will compare it with the cosine distance between the feature representations.

$$cosine\_distance(\mathbf{v_1}, \mathbf{v_2}) = 1 - cosine\_similarity(\mathbf{v_1}, \mathbf{v_2}) \tag{1}$$

$$= 1 - \frac{\mathbf{v_1}.\mathbf{v_2}}{||\mathbf{v_1}||_2 \; ||\mathbf{v_2}||_2} \tag{2}$$

The implementation of this distance measure can be found in *siamese.py*. The effective comparison of the DL systems with the euclidean distance ('DL-Euclidean') and with the cosine distance ('DL-Cosine') will appear within the answers to questions 2 to 6 in the next sections. We refer to both at the same with the designation 'DL'.

2 Q1

See *assignment_3_2022.ipynb*.

I compute a distance matrix where each cell corresponds to the distance *dist_metric* between a pair of samples.

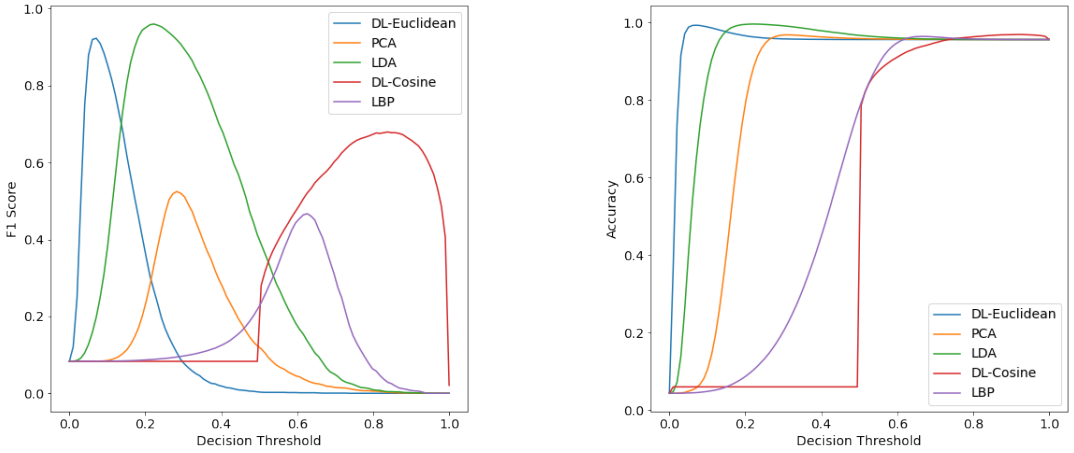
I also compute the similarity matrix which differs from the distance matrix by the fact that the measure between two representations $\mathbf{v_1}$ and $\mathbf{v_2}$ is the following similarity measure

$$sim(\mathbf{v_1}, \mathbf{v_2}) = \frac{1}{1 + dist\_metric(\mathbf{v_1}, \mathbf{v_2})}$$

I finally obtain the similarity score by extracting the non-diagonal elements (i.e. the unnormalized similarity scores between different images) of the similarity matrix and by normalizing them in the range $[0, 1]$.

3 Q2

The accuracy and F1 score are shown in figure 1.



((a)) F1 Score ((b)) Accuracy

FIGURE 1 – F1 Score & Accuracy

TABLE 1 – Optimal classification thresholds under different criteria

	Max *f1-score*	Max *accuracy*	Max *balanced accuracy*	Max $\sqrt{TPR * TNR}$	*EER*
LBP	0.63	0.67	0.53	0.53	0.49
PCA	0.28	0.31	0.22	0.21	0.19
LDA	0.22	0.22	0.16	0.16	0.16
DL-Euclidean	0.07	0.07	0.05	0.05	0.04
DL-Cosine	0.84	0.92	0.66	0.66	0.63

The point at which the F1 score is maximal makes a decent compromise between precision (the fraction of samples predicted as genuine that are truly genuine) and recall (the fraction of truly genuine samples that are correctly predicted as such). The F1 score focuses on the classification of the genuine class. Nevertheless, depending on the application, a balance between both classes could be sought instead.

Accuracy promotes a good specificity, i.e. a very low FPR = 1 - TNR, in our case because there are far more negative than positive pairs of samples in a verification system, i.e. there are far more possible imposters than samples of genuine identity. Accuracy is thus not an appropriate metric to find a good balance between sensitivity (i.e. convenience) and specificity (i.e. security).

Both F1-score and accuracy happen to recommend the same decision threshold in our case although in general F1-score is a more trustworthy metric that is not influenced by class imbalance. It however focuses on the correct classification of positive samples while accuracy focuses on the correct classification of negative samples due to high class imbalance.

Other optimal thresholds than those obtained by looking at the maximum f1-score and accuracy in figure 1 can be obtained as follows. All optimal thresholds are reported in tableau 1.

We can rely on the balanced accuracy, which is defined as the average of recall obtained on each class. Contrary to the traditional accuracy, it is not influenced by class imbalance and promotes thus evenly a good TPR (specificity) and TNR (sensitivity).

The geometric mean of the TPR and TNR can be an alternative to the harmonic mean used in the F1-score. The main difference relies in the fact that the extreme values are less penalized by the geometric mean. It nevertheless inherits the other disadvantages of F1-score.

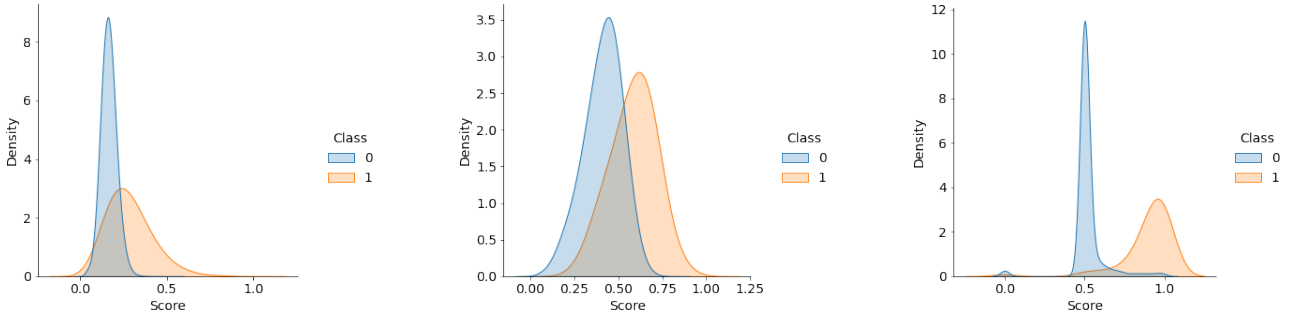
The equal error rate is also an optimal point where a perfect balance between TPR and TNR is sought. It corresponds indeed to the point at which TPR = TNR, i.e. FPR = FRR. The EER point is thus the fairest balance (in between a low FPR (i.e. high security) and a low FRR (i.e. high convenience).

The last three techniques try to find a balance between TPR and TNR. They often advise on thresholds that are close or equivalent. In practice however, the decision threshold depends on the requirement of the application at hand. The balance between FPR and FRR is indeed application-specific. Some applications may require a higher security (low FPR) such as to access your banking app and others may require a higher convenience of use (low FRR) such as to get access to your mobile phone.

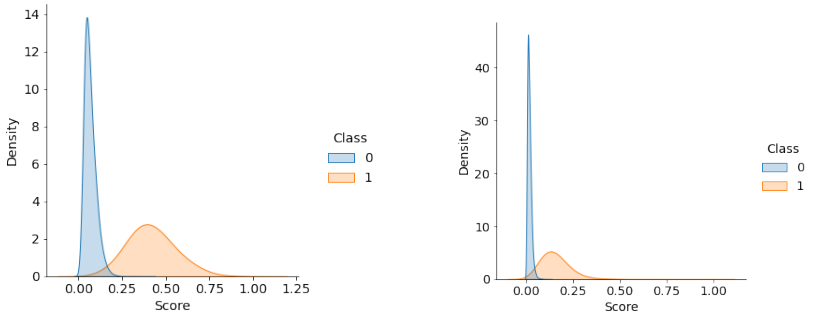
4 Q3

The different figures below are put in approximate decreasing order of the overlap of the class distributions. It makes PCA and LBP the least discriminatory techniques, then DL-Cosine, LDA and DL-Euclidean. Note that the major difference between these two sets of techniques is that the second set (DL-Cosine, LDA, DL-Euclidean) are based on supervised learning algorithms. They thus take into account the training examples' class to build discriminatory image representations.

On the one hand, we see that the distribution of the impostor class for PCA and especially for LBP spreads over a wider range of scores. Consequently, we should expect a higher FAR for low decision thresholds. On the other hand, we see that the impostor class for LDA and DL is limited to a very narrow region. As a consequence, the FAR will most probably be very low for most decision thresholds. On the contrary, the genuine distribution spread over a broader range of va-



((a)) PCA ((b)) LBP ((c)) DL-Cosine



((d)) LDA ((e)) DL-Euclidean

FIGURE 2 – Genuine (class 1) and Impostor (class 0) Score Distributions

lues for the five techniques. As a result, the FRR will probably be much more dependent of the decision threshold we select.

Looking at the genuine and impostor distributions that do not overlap much for LDA and DL, we should expect that with an optimal decision threshold, the FAR and FRR will be very low. Indeed, the two classes are for the most part of the score range very distinctive. The contrary can be deduced from the overlapping distributions for LBP and PCA.

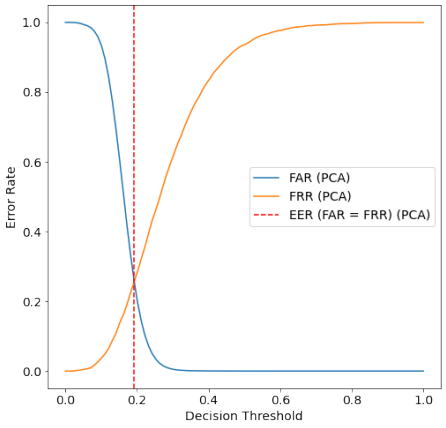
It makes sense as PCA focuses on projecting the images in a low-dimensional space where the principal components maximise the encoded variance of the original images. It thus computes the projections/features to minimize the reconstruction error of these images rather than to focus on class-specific features. The highest variation in images typically come from differences in illumination, which is what we mostly observe as being encoded in the first principal components.

LBP descriptors only encode for texture and misses thus rich information from the geometry of the face, the structure of the face components, ... That is why we also observe a substantial overlap between the genuine and impostor distributions.

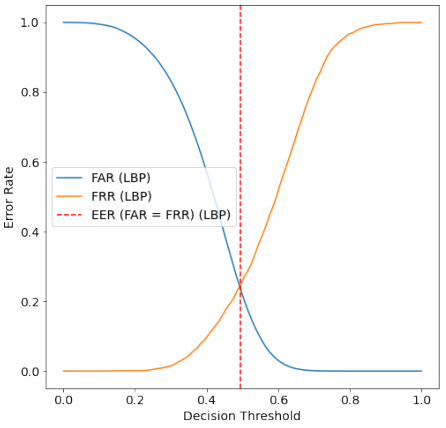
Similarly to PCA, LDA applies a linear transformation to the face images to reduce the dimensionality of the problem. Contrary to PCA, it maximises the ratio of the between-class variance over the within-class variance. Taking into account class information allows thus to project the images into a more class-discriminatory subspace.

Finally, our Deep Learning approaches learn a feature embedding via a neural network trained to minimize the categorical cross entropy with the purpose to maximise the classification accuracy. Convolutional Neural Networks are particularly popular in computer vision due to weight sharing. It allows the network to apply the same local transformation in every region of the image. Stacking convolutional layer together progressively increases the effective receptive field of the transformations such that the deeper we move into the network, the more global image features are. It encodes particularly well class-specific features, which explains the smaller overlap between class distributions. We however already sees a difference in favour of the euclidean distance compared to the cosine distance. The overlap is indeed larger with the cosine distance.

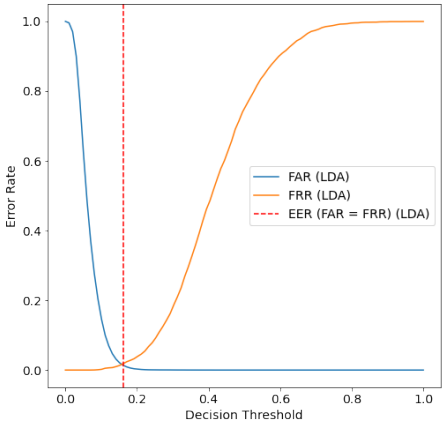
5 Q4



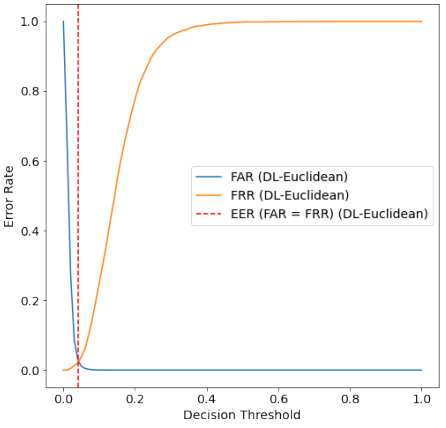
((a)) PCA



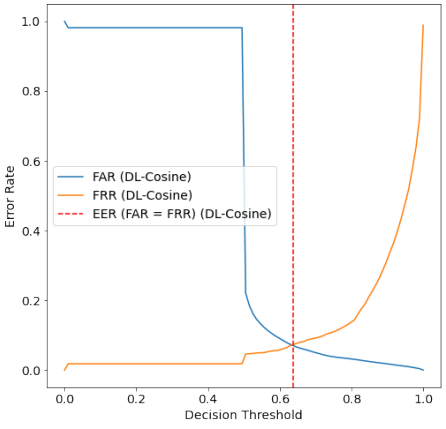
((b)) LBP



((c)) LDA



((d)) DL-Euclidean

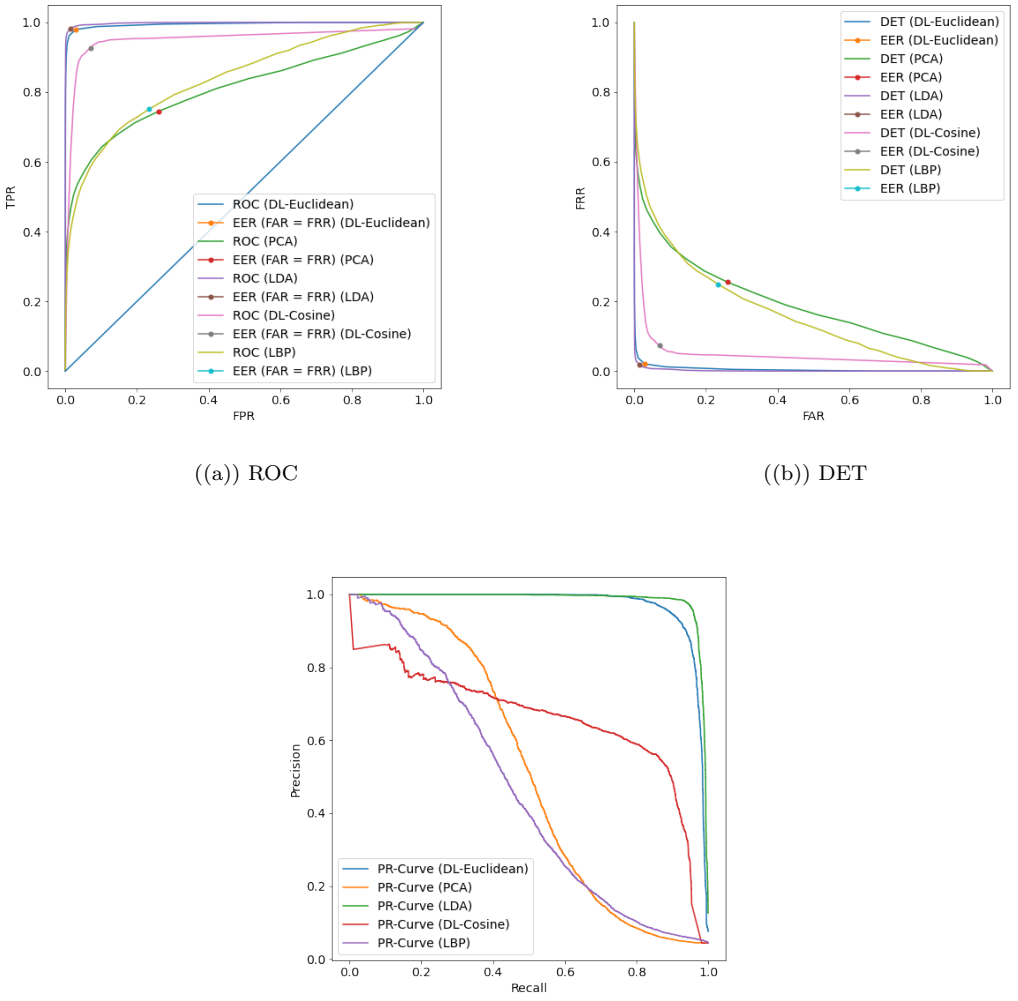


((e)) DL-Cosine

FIGURE 3 – FAR-FRR Curves

In figure 3, we see that FAR curves are very steep for DL, LDA, less for PCA and much less for LBP. It confirms my intuition on FAR from observing the imposter class distributions in Q2. FAR is much less threshold dependent for DL and LDA. For every feature extraction technique, FRR curves are however much less steep. It corroborates the intuition in Q2 that FRR is much more threshold-dependent because the genuine class distribution is much wider. As a consequence of those observations, the equal error rate is much higher for PCA and LBP than for LDA and DL.

To compare more objectively the five techniques, let us have a look at the DET, ROC and Precision-Recall curves side by side (see figure 4).



((a)) ROC

((b)) DET

((c)) Precision-Recall

FIGURE 4 – ROC, DET & PR Curves

As expected, the area under the ROC curve is quite big for LDA (0.9985), DL-Euclidean (0.9954) and DL-Cosine (0.9506) whereas it is way lower for LBP (0.8401) and PCA (0.8109). The ROC-AUC provides an aggregate measure of performance across all possible classification thresholds. It can be interpreted as the probability that the model scores a random genuine sample more highly than a random impostor. The closer it is to 1 (100% of correct classification), the better it will classify irrespective of the decision threshold. However, in real-world applications, the classification threshold remains an important factor that greatly differs from a case to another.

For instance, the ROC curve of the LBP system is only above the one of the LDA system for higher FPR values. As a result if a decision threshold with low FPR value can be accepted for some application, the TPR will be higher with PCA despite having a lower ROC-AUC with it than with LBP. Both however have overall very bad ROC curves that would not suit any practical biometric verification system.

Nonetheless, looking at the ROC curves for LDA and DL, we should expect a much better class discriminator. A first observation is that the system based on DL-Cosine hardly gets a TPR higher than 0.95. It is thus limited in terms of convenience of use. We can not spot a region where the DL systems perform better than the LDA system although at both extremes of the FPR values, the two systems converge (which system is chosen then becomes irrelevant). We know from figure 3 that high decision thresholds result in a low FAR but among these thresholds, the system can differ a great deal in FRR. Comparatively, very low decision thresholds result in a high FAR and low FRR. Regions of convergence of the different ROCs (corresponding to extreme values of the decision threshold) are thus not relevant in most applications.

For DET curves, we observe again a very flat slope on a broad range of the horizontal axis for LDA and DL. It means that the FAR (resp. security) can be reduced (resp. increased) without increasing (resp. decreasing) much the FRR (resp. biometric system convenience of use). This observation makes sense with what we have discussed for ROC curves as FRR = 1 - TPR and FPR = FAR. Similarly to what we observed in the ROC curves, PCA and LBP do no offer good trade-offs between FAR and FRR.

Let us now have a look at the Equal Error Rate. The EER point is the fairest choice (in the sense that FRR = FAR) between a low FAR (i.e. high security) and a low FRR (i.e. high convenience). As already mentioned before, FAR and FRR's importance are however application specific. We may want to lower one more than the other. Considering that they have as much importance, the approaches can be classified in crescent order of the EER as follows : LDA, DL, LBP, PCA.

For the precision-recall curves, we observe that the embeddings retrieved from LDA and DL-Euclidean can be really well discriminated with their respective *dist_metric*. The precision-recall curves are especially interesting in scenarios where there is typically a high class imbalance. It however focuses mainly on the genuine class (via PPV and TPR) in minority and cares less about the frequent impostor class (the true negatives are not taken into account as in ROCs). In the end, the precision is not a valid alternative for the FPR of the ROC. Security is indeed not captured by precision as the true negatives are not taken into account. A verification system should however balance security (i.e. low FPR) and convenience of use (i.e. high TPR).

We see that the curves for DL-Euclidean and LDA are quite flat under low recall values. It therefore implies that the recall can be increased effortlessly without affecting the precision. However, from a recall of more or less 0.8 onwards for DL-Euclidean (resp. 0.95 for LDA), we observe a steep drop in precision. It becomes very costly in terms of precision to improve the recall (= sensitivity or 1-FRR, i.e. convenience of use). For decent recall values, the precision with LDA is better than with DL. However, at extreme values of precision or recall, both systems are equivalent. DL-Cosine achieves lower performance especially in terms of precision with a steeper slope.

With regard to PCA and LBP, we hardly find a good balance between precision and recall that would suit a practical verification system. Both performs badly but if precision is preferred over recall, PCA is a better choice than LBP and DL-Cosine (and vice-versa).

We observe also that the lower limit of the precision in the PR-curves is very low (≈ 0). It is determined by the ratio of genuine samples among the total population which is unsurprisingly very low given the important class imbalance.

As calculated for ROC curves, let us know look at the area under the precision-recall curve. It is an aggregate measure of the precision across all possible recall values. The higher it is, the better the system is at correctly detecting genuine samples. It is, as for ROC-AUC, classification threshold independent. The classification threshold is however a crucial parameter that is adjusted differently for each application. In decreasing order of the PR-AUC, we have LDA (0.9853), DL-Euclidean (0.9715), DL-Cosine (0.6604), PCA (0.5151) and LBP (0.4604). Note that DL-Cosine seems to be a way less valuable choice when looking at PR curves and PR-AUCs then when looking at ROCs and ROC-AUCs. Note also that the PR-AUC metric considers PCA better than LBP while it is the contrary with regard to ROC-AUC. We will prefer ROC-AUCs over PR-AUCs for the same reason that we prefer ROCs over PR curves.

The average precisin score is an alternative method to compute the PR-AUC. It is explained in sklearn documentation that

"[*average_precision_score*] implementation is not interpolated and is different from computing the area under the precision-recall curve with the trapezoidal rule, which uses linear interpolation and can be too optimistic." [1]

1. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html#sklearn.metrics.average_precision_score

We will thus preferably look at the results of this second implementation in place of the first one. The results are not much different in practice : LDA (0.9853), DL-Euclidean (0.9715), DL-Cosine (0.6588), PCA (0.5151) and LBP (0.4604).

In general, we can conclude that only DL and PCA could lead to a practical application in verification system. DL-Cosine has however less applications as TPR (the convenience of use) plateaus around 0.95.

6 Q5

To compute the CMC curves, I select the first sample of each class to form the n_{ref} reference samples and I compute their matching scores to the other samples, namely the n_{probe} probe samples. The recognition rate at each rank r is computed and summed cumulatively.

Looking at figure 5, we see that for most ranks the recognition rate is the highest for LDA, then DL-Euclidean, then DL-Cosine, then, LBP and finally PCA. However, there are 3 exceptions to this statement. In the very first ranks, PCA outperforms LBP. In the ranks from 7 to 12, DL-Euclidean performs better than LDA and after that, it performs as well as PCA. From rank 14 onwards, DL-Cosine performs as well as DL-Euclidean and PCA.

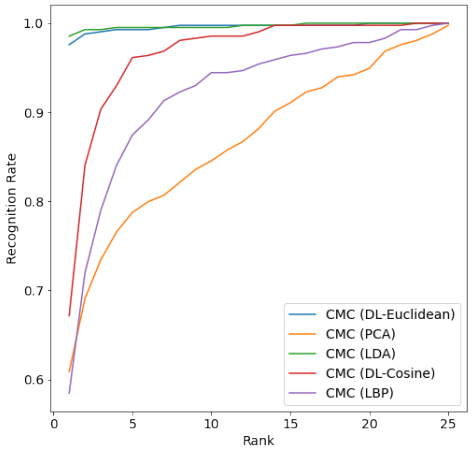


FIGURE 5 – CMC

If we look more specifically at the rank-1 recognition rate, the highest value is achieved by LDA (0.9855), DL-Euclidean (0.9758), DL-Cosine (0.6715), PCA (0.6087) and then LBP (0.5845). We notice a particularly important gap between LDA, DL-Euclidean and the other systems : DL-Cosine, PCA and LBP. Note that PCA performs slightly better on the rank-1 recognition rate than LBP whereas otherwise, there is generally a substantial gap between the CMC curve of LBP and PCA.

We can conclude that DL-Euclidean and LDA could be used in a practical identification scenario.

7 Classification-based scoring method

In this section, I will describe my solution to task 3.

At first, I leave one image of each of the n classes out for the test set. I experienced a perfect rank-1 recognition rate when using the rest of the dataset as train set. I decided thus to try something a bit more challenging : one-shot learning. It consists in learning from only one or a few samples per class. In my case, I built a train set and a validation set, each with two images of each class.

I chose to build a classifier based on a combination of a CNN and SVM. The architecture of the deep-learning siamese network with euclidean distance from the previous questions is reused. I however train it this time on our new train set and validate it on our new validation set, Then, from the retrieved embeddings, I train an SVM classifier with an RBF kernel.

To obtain the classification probabilities for the test images, I retrieve their DL embeddings and pass them through the trained SVM. It yields a (nxn) similarity matrix in which each row corresponds to a test image and each column corresponds to an individual in the dataset.

Hereunder, I provide the figures for the evaluation of the verification and identification scenario.

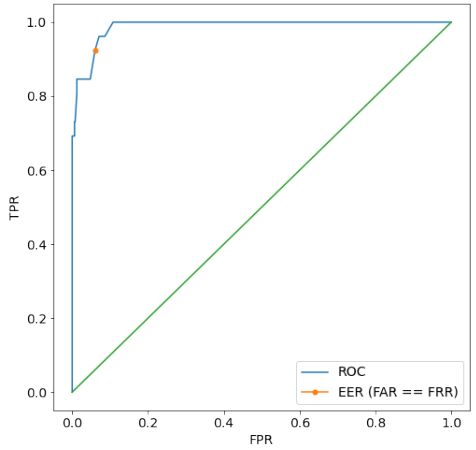


FIGURE 6 – Classifier ROC

The ROC-AUC is of 0.9878 which announces very good performance. And indeed, the ROC shows that a very good balance between TPR and FPR can be achieved. The curve has a relatively flat slope for most FPR values. It ensures that security can be easily increased without damaging the convenience of use. For instance, a TPR of 1.0 can be kept at a FPR of 0.15. Using this system in a practical verification scenario seems thus possible. The EER shows also a decent trade-off between TNR and TPR around 0.925.

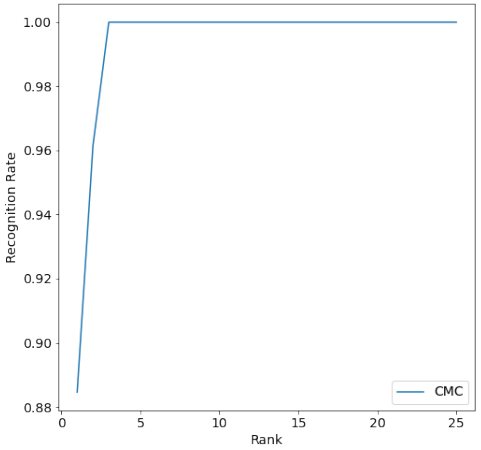


FIGURE 7 – Classifier CMC

Concerning the identification system, a high rank-1 recognition rate of 0.8846 is obtained. For a practical application, we would however benefit from the rank-2 or even better rank-3 recognition rate of 1.0.