

# Rapport BE : Extension d'un système de reconnaissance faciale aux visages portant un masque chirurgical

N'DIAYE BAPTISTE  
21902711 – L3 INFORMATIQUE – TDA2

mai 2022



Entreprise : IRIT/REVA  
Encadrant : Jean-Denis DUROU

## Résumé

Ce bureau d'étude a pour but de développer un système de reconnaissance faciale utilisant la méthode des eigenfaces pour ensuite y implémenter la prise en compte des masques chirurgicaux.

# Table des matières

<b>I</b>	<b>Présentation de l'organisme d'accueil</b>	<b>3</b>
<b>II</b>	<b>Présentation du sujet</b>	<b>4</b>
<b>III</b>	<b>Déroulement du BE</b>	<b>5</b>
	III.1 Diagramme de Gantt . . . . .	5
	III.2 Travail effectivement réalisé . . . . .	6
<b>IV</b>	<b>Bilan</b>	<b>10</b>
<b>A</b>	<b>Statistiques</b>	<b>12</b>
<b>B</b>	<b>Résultats</b>	<b>18</b>
<b>C</b>	<b>Bibliographie</b>	<b>20</b>

# I – Présentation de l'organisme d'accueil



Ce bureau d'étude a été proposé par l'IRIT, ou plus précisément, l'équipe REVA. L'Institut de Recherche en Informatique de Toulouse est une des plus imposante unité de recherche au niveau national, ainsi que l'un des piliers de la recherche en Occitanie. L'IRIT est le 2ème laboratoire de France en terme d'effectifs [1] et est rattaché à l'école doctorale Mathématiques, informatique, télécommunications de Toulouse.

Cette institut agit au niveau européen ainsi qu'international avec plusieurs coopérations comme le "French spanish laboratory for advanced studies in information, representation and processing" (Arménie) ou encore leur participation au " Japanese French Laboratory for informatics".

Leurs recherches se structurent autour de 5 grands sujets scientifiques : [5]

- Conception et construction de systèmes
- Modélisation numérique du monde réel
- Concepts pour la cognition et l'interaction
- Etude des systèmes autonomes adaptatifs à leur environnement
- Passage de la donnée brute à l'information intelligible

L'équipe REVA traite des problématiques liées à l'utilisation de méthodes mathématiques et algorithmiques pour l'analyse de données multimodales, visuelles, biologiques. [6]

Ses thèmes de recherche portent sur des techniques de génération, d'optimisation et d'analyse de données multimodales multidimensionnelles pour produire, calibrer, contrôler des environnements numérisés ou simulés. Ces recherches proposent plusieurs méthodes (méthodes variationnelles, paradigmes de vie artificielle, apprentissage et optimisation ...).

## II – Présentation du sujet

Le but de ce BE est de créer un logiciel de reconnaissance faciale permettant de reconnaître un individu masqué et autoriser ou refuser son accès à un bâtiment. Pour cela, on peut s'imaginer que notre logiciel traitera les images d'une caméra placée à l'entrée du bâtiment image par image. Le logiciel doit surtout limiter le nombre de "faux positifs", c'est à dire qu'il ne doit pas laisser passer des personnes non-autorisées.

La difficulté de la mise en place de ce projet réside principalement dans le fait que l'on n'aura dans très peu de cas une "image parfaite", c'est à dire une image net, prise d'un bon angle, avec un fond unis ainsi qu'un éclairage parfait. Le rajout de la prise en compte du masque augmente encore plus la difficulté.

Dans notre cas, nous avons travaillé avec une banque d'"image parfaite", ce qui nous a facilité la tâche pour développer notre logiciel. La méthode que nous avons du utilisé est celle de la reconnaissance faciale par eigenfaces ou visages propres en français.

Cette méthode a été développée par Sirovich et Kirby en 1987 et utilisée par Matthew Turk et Alex Pentland pour la classification de visages. Nous nous sommes donc basés sur leurs travaux pour mener à bien notre objectif.



*Alex Pentland*

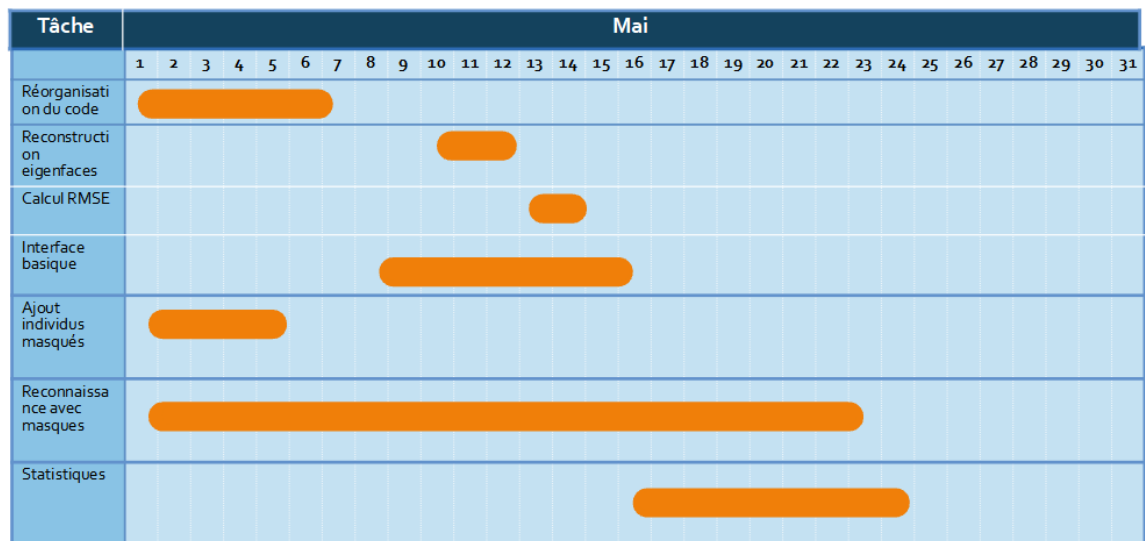
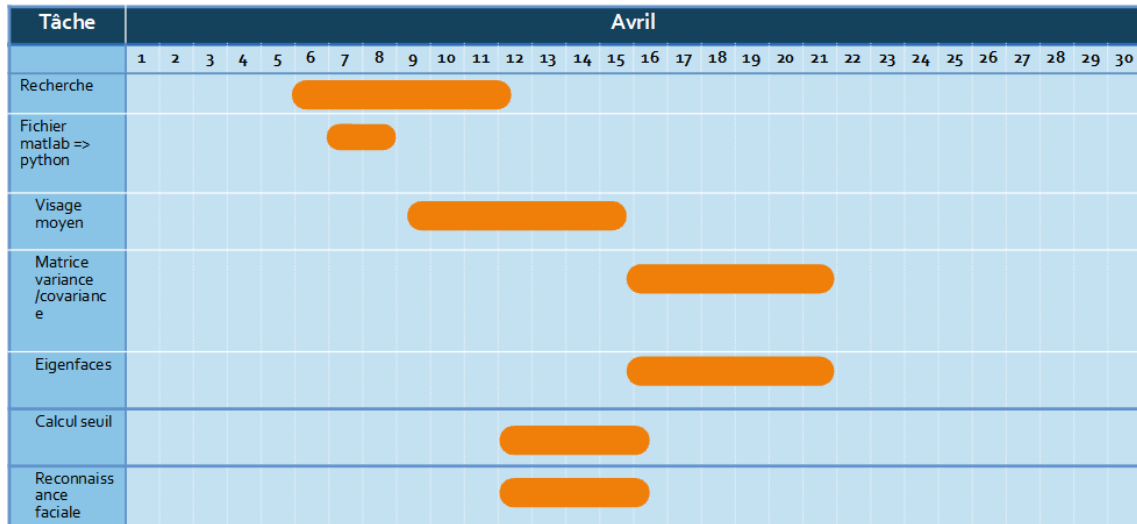


*Matthew Turk*

J'ai travaillé sur ce projet avec Nicholas Makhoul, nous avons tous les deux travaillé à distance en utilisant discord pour échanger autour du projet. Nous communiquions tout au long du projet notre avancé mutuelle et nous avons une réunion environ toute les deux semaines avec M. Jean-Denis DUROU, pour faire le point sur notre avancée ou éventuel blocage.

## III – Déroulement du BE

### III.1 Diagramme de Gantt



## III.2 Travail effectivement réalisé

Pour commencer, un travail de recherche a été nécessaire, puisque l'on ne connaissait pas grand chose sur les eigenfaces. En parallèle, j'ai commencé à transvaser le code qui nous a été fourni de matlab vers python.

Celui-ci nous permet de stocker les images constituant la base de donnée dans un fichier 12. Une fois cela fait, j'ai pu commencer les calculs permettant de reconnaître un individu de cette base de donnée.

J'ai réalisé une première version du programme, en calculant l'individu moyen, les eigenvectors ainsi que les composantes principales 12. De plus, j'ai aussi calculer la distance minimale entre l'image à tester et les images de la base de donnée pour ainsi déterminer si l'image à tester correspond à un individu de la base de donnée ou non.

J'ai également déterminer un seuil de reconnaissance en calculant la distance minimum des images n'appartenant pas à la base de donnée. Pour cette première version, je me suis inspiré un tutoriel ci-contre [2].

Cette première version ne fonctionnait pas correctement, certains individus n'appartenant pas à la base de donnée on été identifiés comme appartenant à la base de donnée.

Cette version était laborieuse, les calculs étaient trop nombreux, ce qui ralentissait le programme. J'ai alors compris que mes calculs n'étaient pas bon, j'ai donc repris à 0 mes calculs en effectuant des recherches plus approfondis sur les eigenfaces et l'analyse en composantes principales.

Grâce à ces recherches, j'ai compris que mon calcul de l'individu moyen n'était pas bon. De plus, j'ai réalisé que mes calculs pouvaient être énormément simplifier grâce à python, numpy et scipy. J'ai alors corrigé l'individu moyen pour ensuite corrigé le reste de mes calculs, ce qui m'a permis, finalement, d'obtenir de bons résultats.

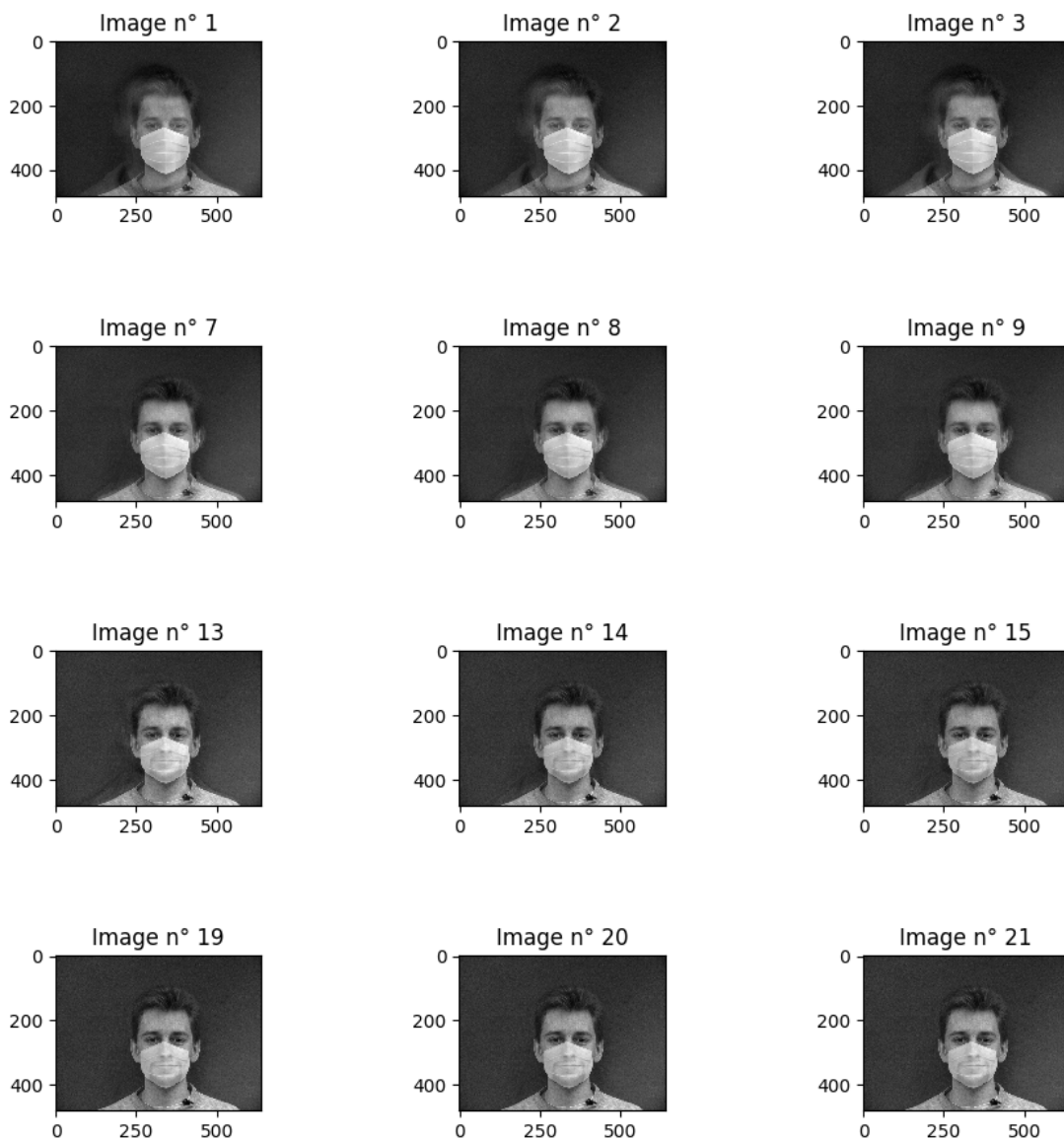
```
20 def eigen_face():
21     av_face = np.mean(X, 0).reshape(480, 640)
22     A = np.reshape(X - av_face, (N, (480 * 640)))
23     eigenfaces, sigma, v = linalg.svd(A.transpose(), full_matrices=False)
24     return eigenfaces, A, av_face
25
26 U, phi, av_face = eigen_face() #U: eigenfaces | phi: X - av_face | av_face: visage moyen
27 weights = np.dot(phi, U)
```

Avec du recul, je constate que j'ai perdu beaucoup de temps en me précipitant sur le code, au lieu de m'assurer de comprendre correctement les calculs. J'ai tout de même pu finir le gros du projet assez rapidement (1 mois avant la date butoir).

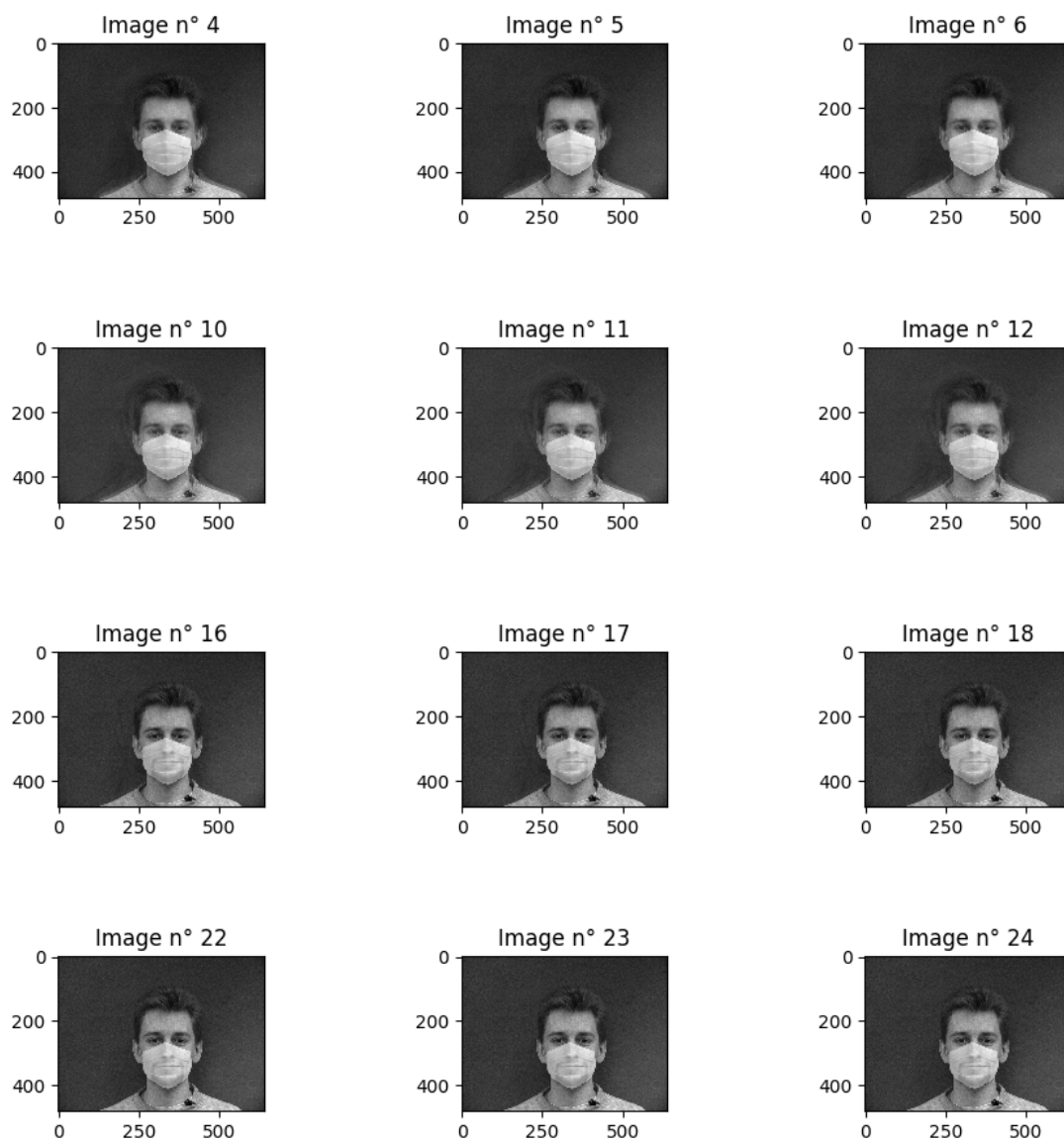
La reconnaissance faciale basique étant faite, j'ai donc essayer d'implémenter la reconnaissance faciale avec masque, pour cela, je me suis de nouveau pencher sur des articles / recherches, mais je n'ai pas trouver grand chose susceptible de m'aider mis à part la notice qui nous a été donné.

Au départ, j'ai considéré les images des individus masqués comme des images "dégradées" puis je les ai reconstruites à l'aide des eigenfaces. J'ai ensuite calculé les composantes principales de cette nouvelle image pour la comparer aux composantes principales des images de ma base de donnée. Cette méthode n'a pas donné de bons résultats, je n'ai pas su identifier mon erreur dans cette implémentation, mais je suspectes que lors de la reconstruction, certaines images très différentes sont devenues plus proche de la bonne image que la bonne image, ce qui fausse complètement les calculs.

Ci dessous voici les images reconstruites pas a pas, l'image n°1 représente l'image reconstruite avec la première eigenface, l'image n°24 représente celle reconstruite avec 24 eigenfaces.

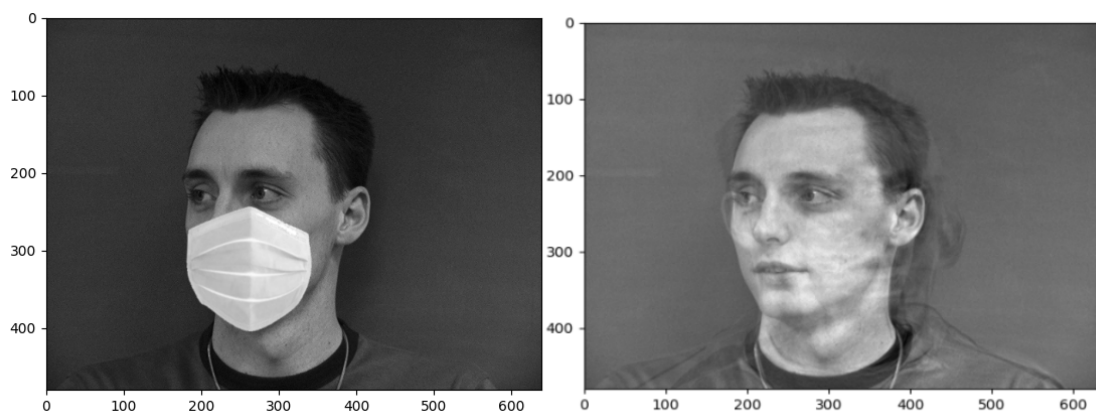


Reconstruction d'une image masquée (1)



Reconstruction d'une image masquée (2)

Je me suis aperçu plus tard que cette méthode n'était pas la bonne. Voici à quoi ressemble une image reconstruite dans mon implémentation finale :



Reconstruction d'une image masquée | implémentation finale



J'ai alors tenter diverses méthodes pour enfin réussir une implémentation correcte de la prise en compte des masques qui est celle-ci :

```
27 def facial_recognition_2(new_Ik, nb_postures, U, weights, seuil, numeros_individus, av_face):
28     u = U.T
29     w2 = np.dot((new_Ik-av_face).reshape(480*640), U) #composantes principales
30     img_test = av_face + np.dot(w2, u).reshape(480, 640) #image reconstruite
31     w = np.dot(u, img_test.reshape(480*640)) #composantes principales de l'image reconstruite
32
33     newW = weights * w + w2
34     dist = np.min(newW**2, axis=1)
35     indiceImg = np.argmin(dist)
36     mindist = np.sqrt(dist[indiceImg])
37
38     if mindist < seuil:
39         print("Identification réussie")
40         print("Correspond à l'individu "+str(numeros_individus[indiceImg//nb_postures]))
41         return True
42
43     else:
44         print("Identification échouée")
45         print("Fin")
46         return False
```

Pour tester mon programme, j'ai modifier les 90 images fournies, avec le logiciel GIMP. J'ai pris une image de masque que j'ai détourer puis ajouter à ces images, j'ai ensuite modifier mon programme pour qu'il les prenne en compte. Cette partie là était très fastidieuse, j'aurai pu m'arrêter à quelques images, mais j'ai préférer en faire davantage pour avoir de meilleurs résultats. Je n'avais jamais fais de montage auparavant, c'était plutôt intéressant mais un peu trop répétitif ici.

Pour inclure la reconnaissance faciale avec masques à notre projet, j'ai fait en sorte qu'un premier essai d'identification soit effectué, si cet essai échoue, le deuxième essai ci-dessus prend le relais, sinon, on renvoie le résultat du premier essai.

Les performances pour 30 identifications d'images sont indiqués ici 10, celles pour 3000 identifications sont indiqués ici 12

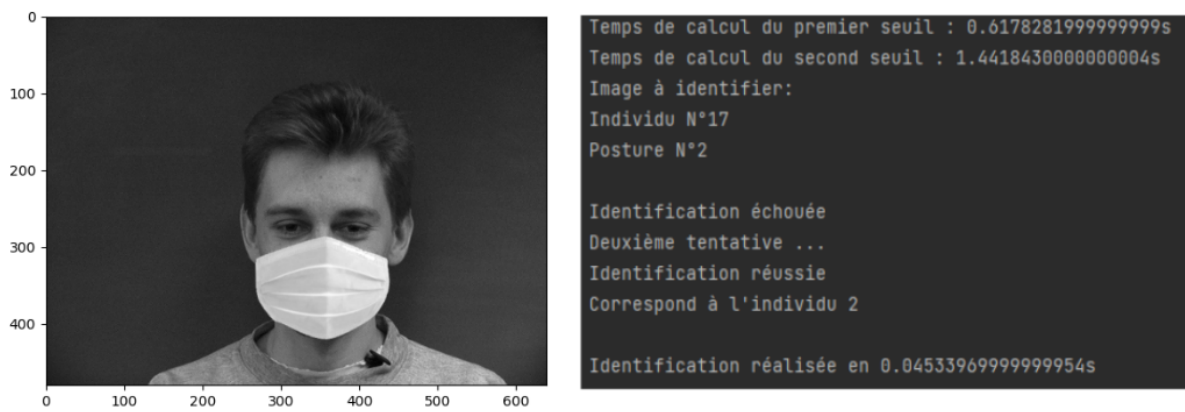
Pour finir, j'ai construits plusieurs graphes et affichages 12, ainsi que relier les statistiques faites par mon collègue à mon code. 11

## IV – Bilan

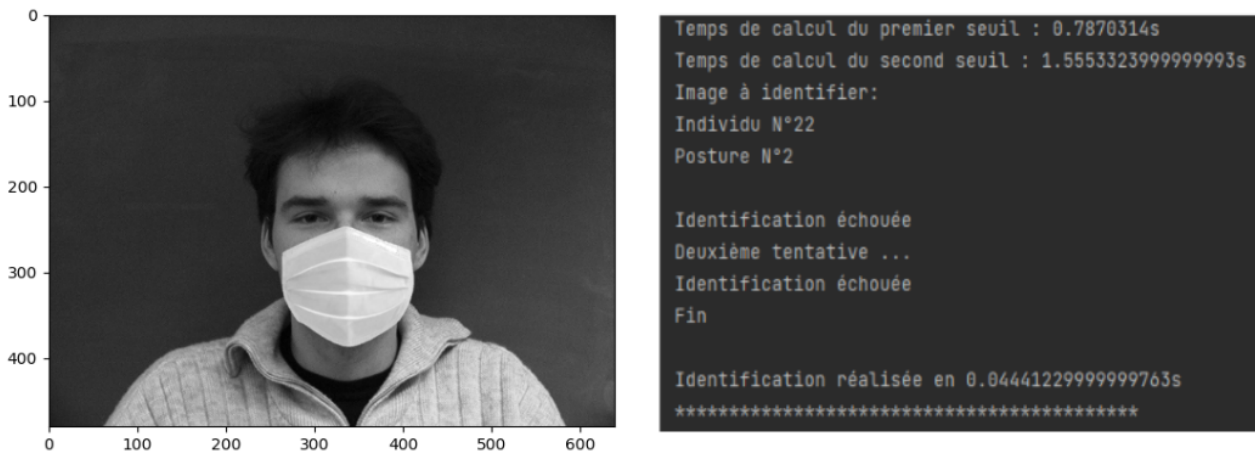
J'ai réussi à implémenter la reconnaissance faciale sans masque en ayant de bons résultats. La reconnaissance faciale avec masque donne de moins bons résultats, mais est acceptable tout de même. Pour continuer ce BE, j'aurai pu tenter de combiner du machine learning à l'analyse en composantes principales pour améliorer les résultats de la reconnaissance faciale avec masque.

Ce BE a été pour moi l'occasion de retrouver un langage que je n'avais pas utilisé depuis un moment (python) et qui me sera sûrement très utile dans ma poursuite d'étude, puisque j'envisage un master en intelligence artificielle pour l'année prochaine. De plus, ce BE m'a également permis d'en apprendre davantage sur les statistiques en python.

Ci-dessous voici deux utilisations de notre logiciel, un échec et une réussite, deux autres utilisations sans masques et une seule identification sont disponibles ici 18.

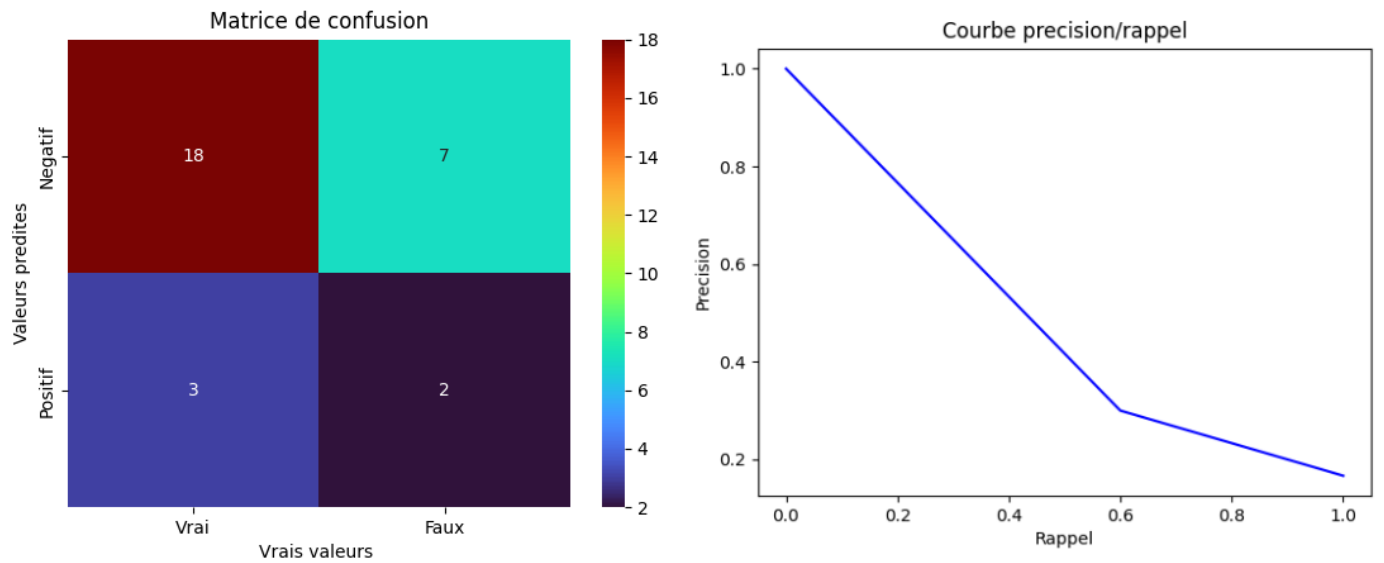


Identification réussie

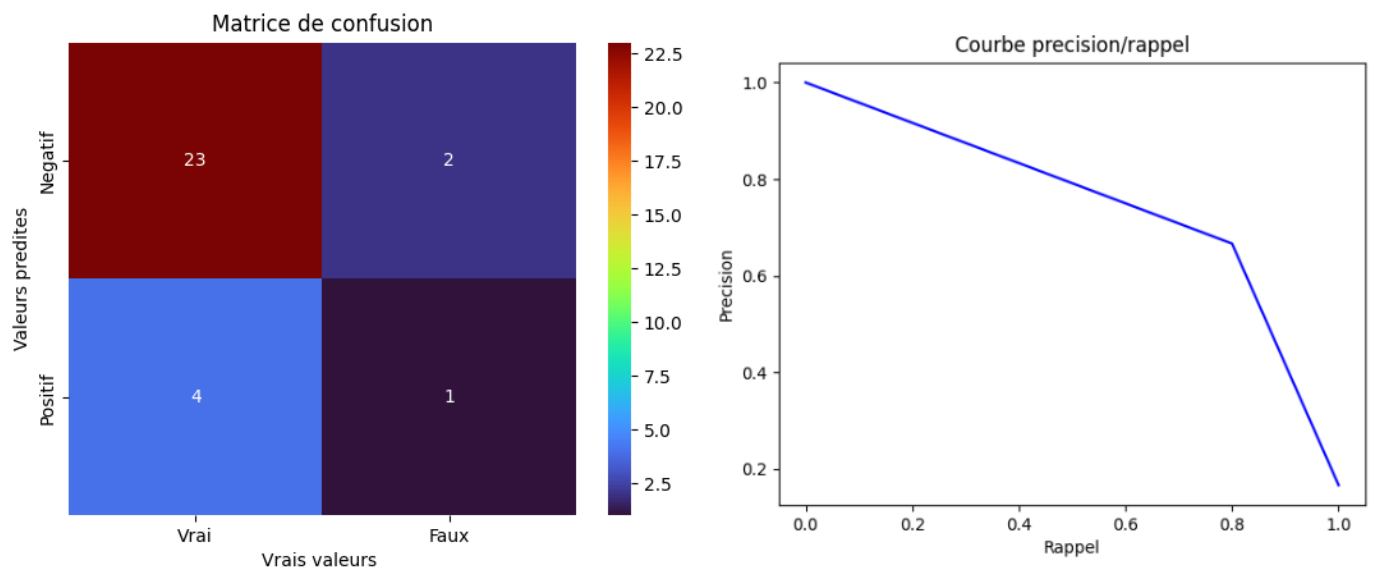


Identification échouée

Sur 30 images tirés aléatoirement, nous avons obtenus les statistiques ci-dessous. On peut observer que la diminution du nombre de positions, donc de données, entraîne une augmentation de mauvais résultats (Faux Négatifs ou Faux Positifs).



Statistiques pour 2 posture sur 6 des individus de la BD

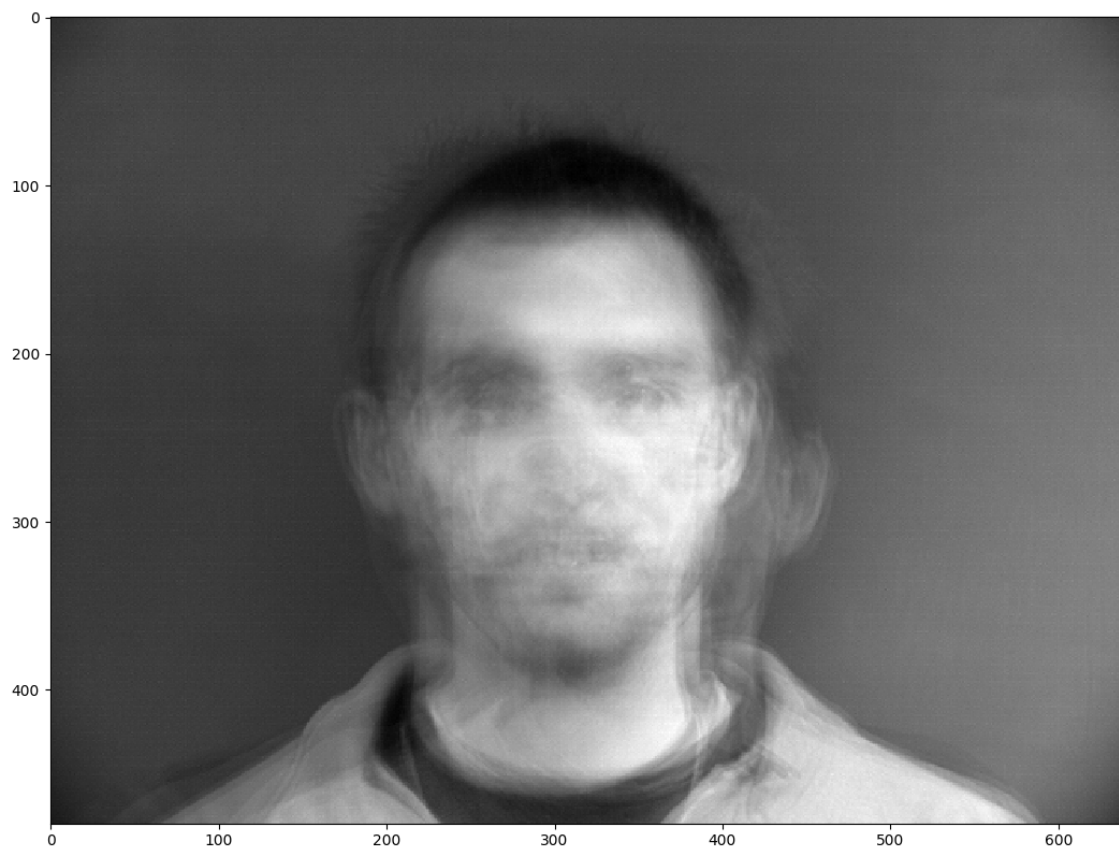


Statistiques pour 6 posture sur 6 des individus de la BD

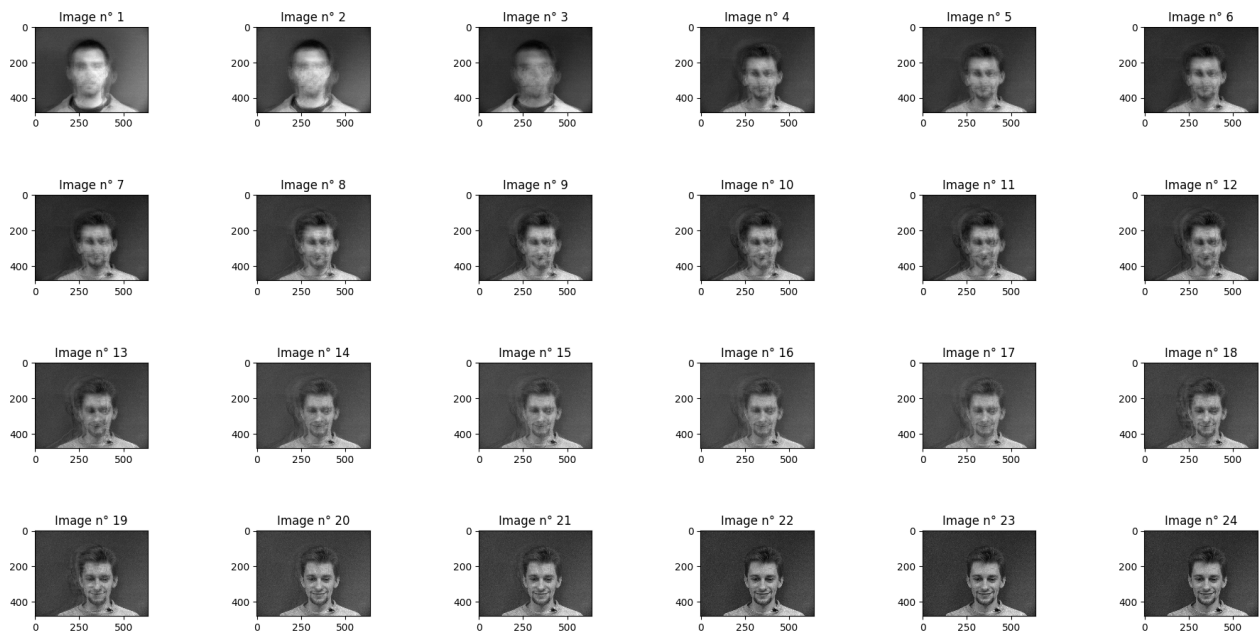
## A – Statistiques



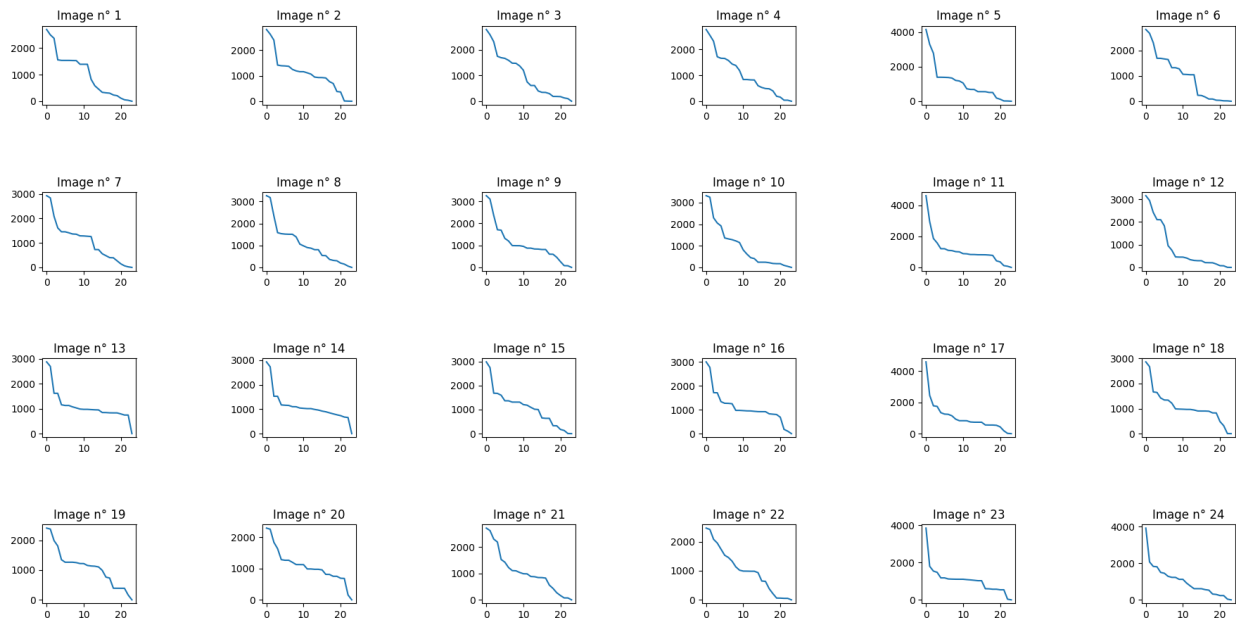
Base de donnée



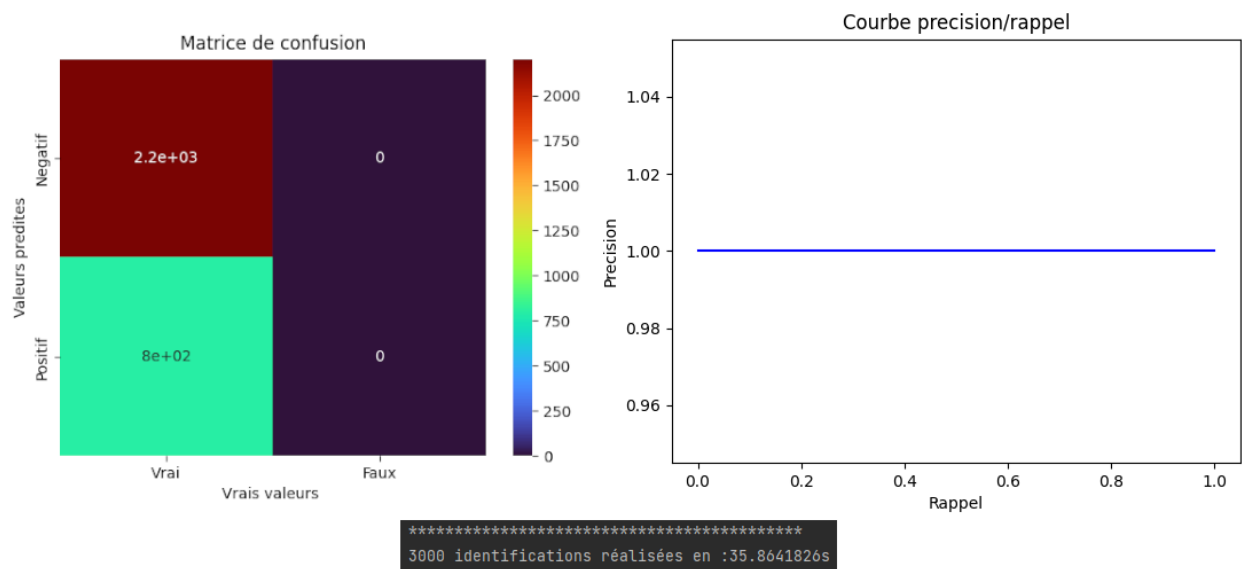
Visage Moyen



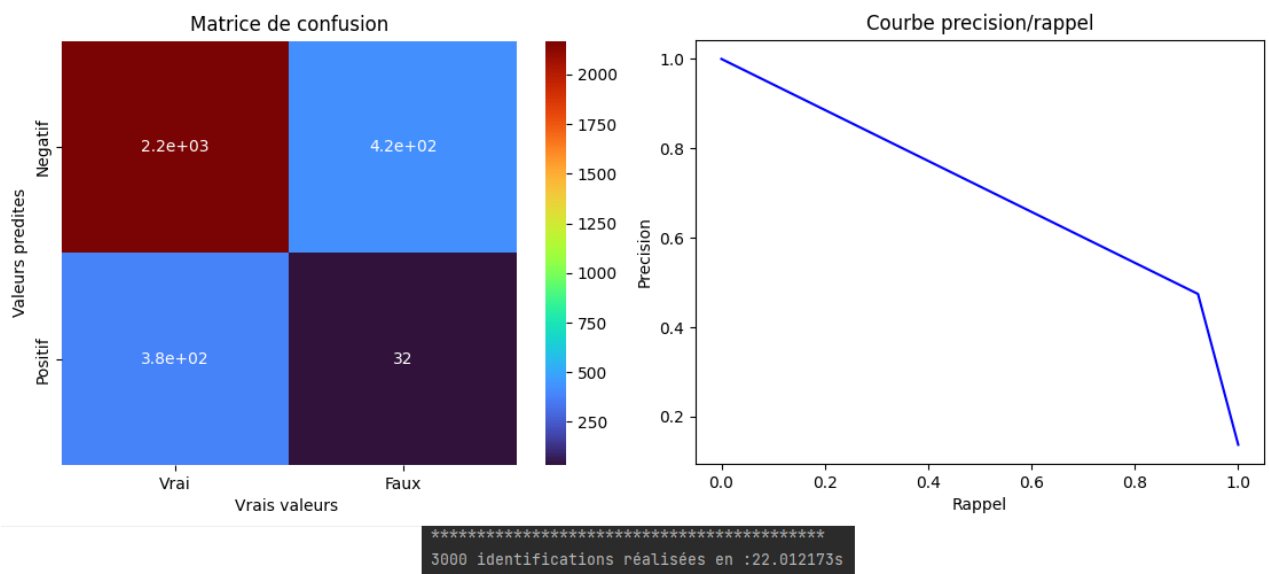
## Reconstruction d'une eigenfaces



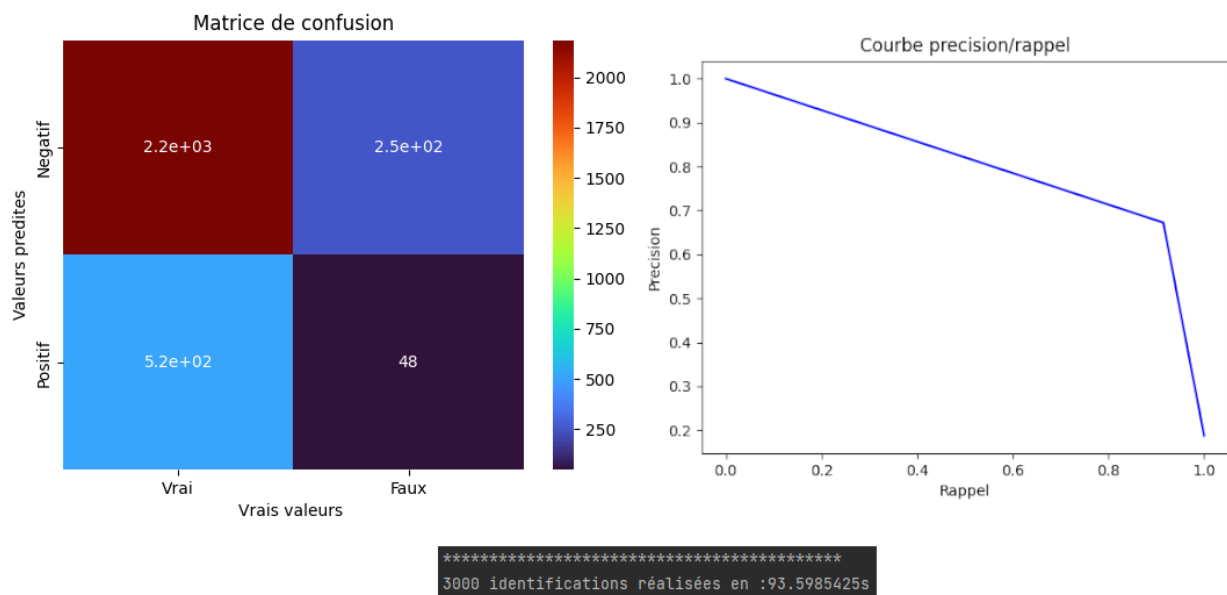
RMSE



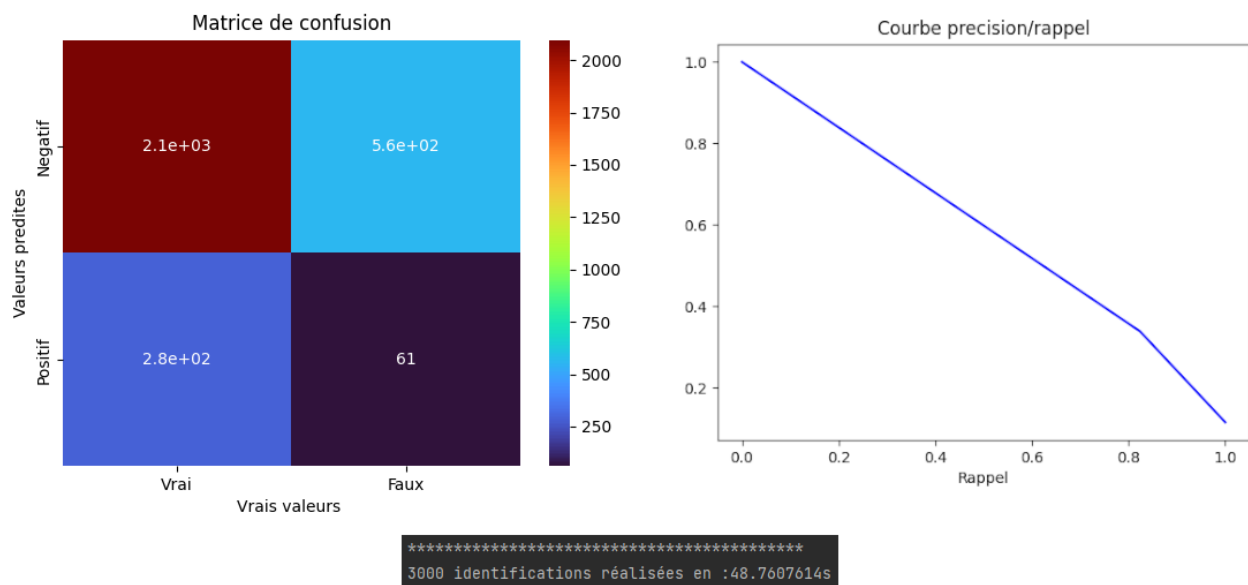
3000 identifications, 6 postures | Sans masques



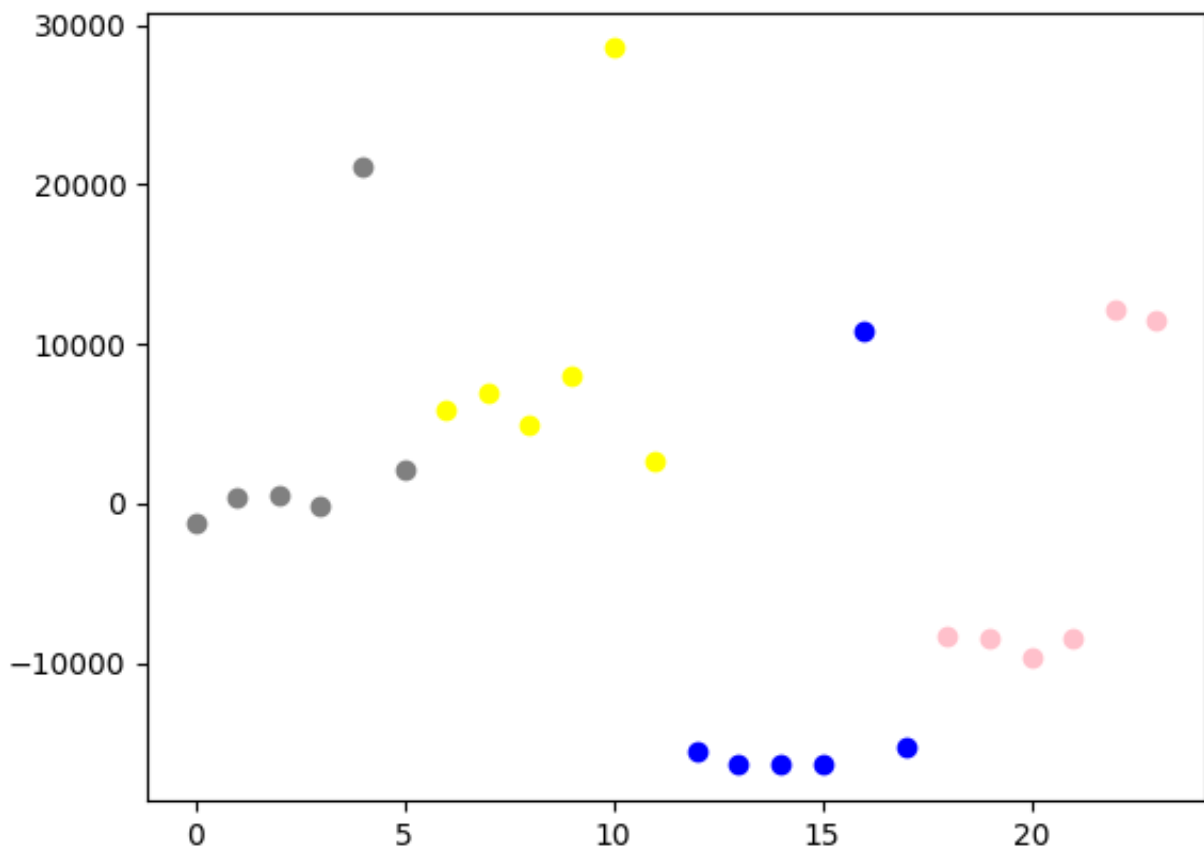
3000 identifications, 2 postures | Sans masques



3000 identifications, 6 postures | Avec masques

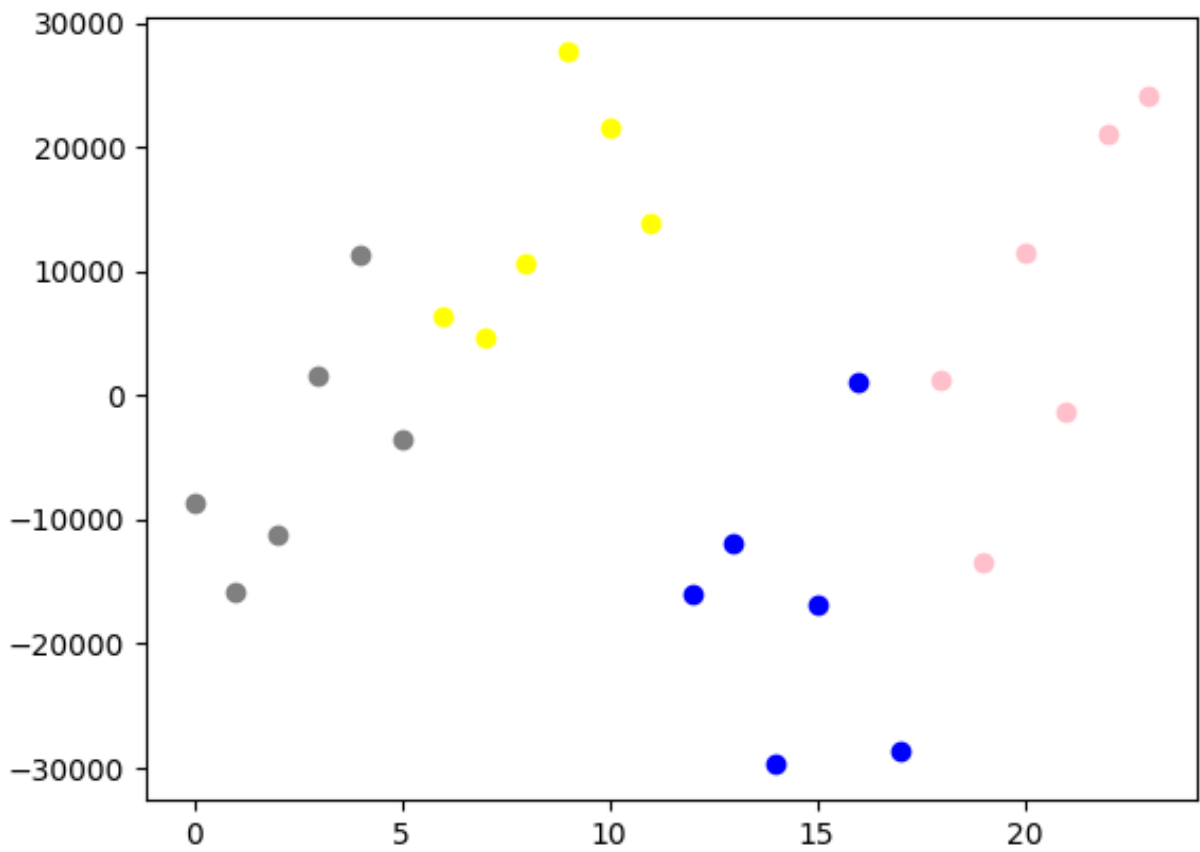


3000 identifications, 2 postures | Avec masques



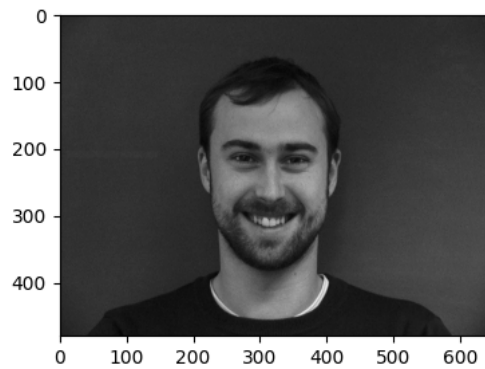
Clusters : 2 premières composantes principales





Clusters : 24 premières composantes principales

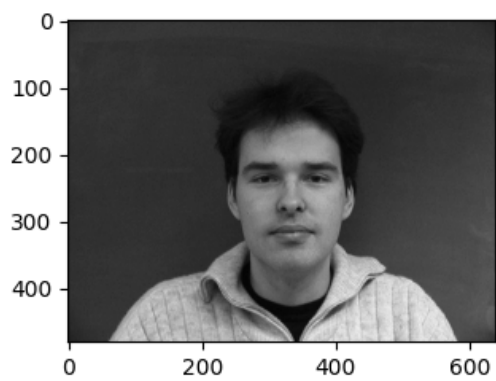
## B – Résultats



```
Temps de calcul du premier seuil : 0.9495258999999998s
Image à identifier:
Individu N°6
Posture N°2

Identification réussie
Correspond à l'individu 6
Image n° : 13
Identification réalisée en 0.01823699999999917s
*****
```

Identification réussie

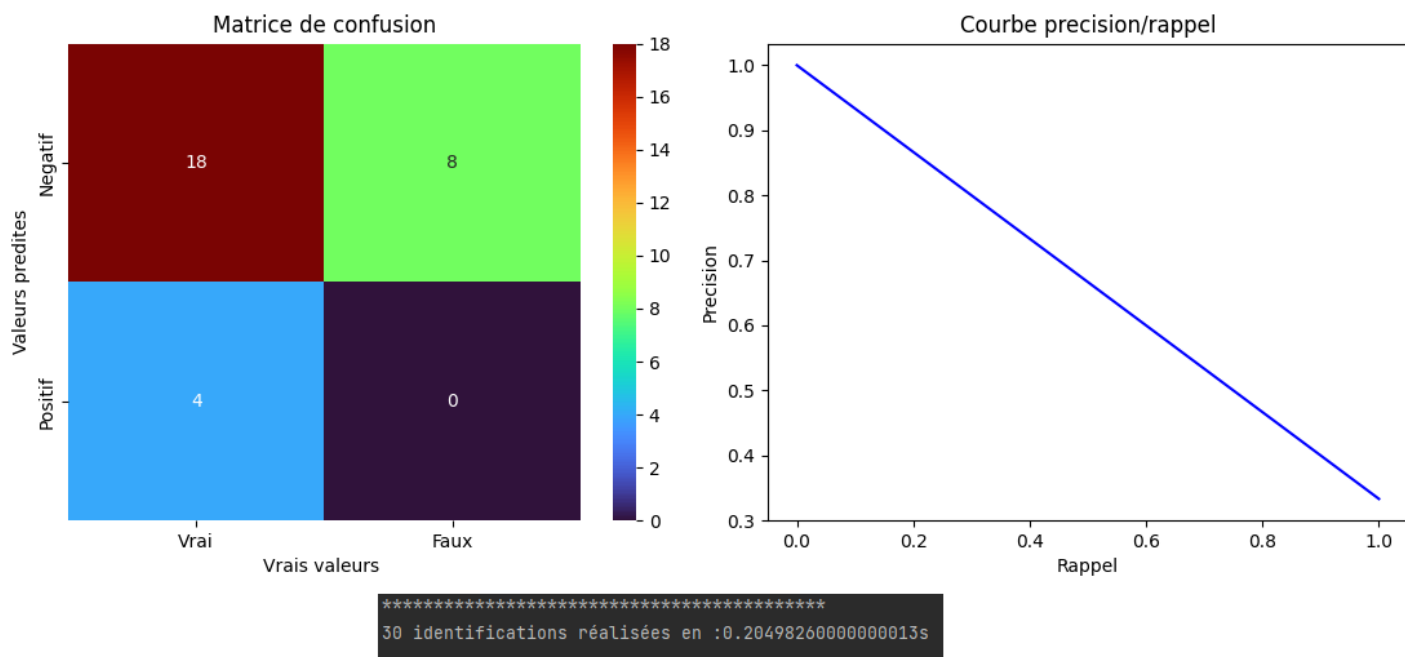


```
Image à identifier:
Individu N°7
Posture N°2

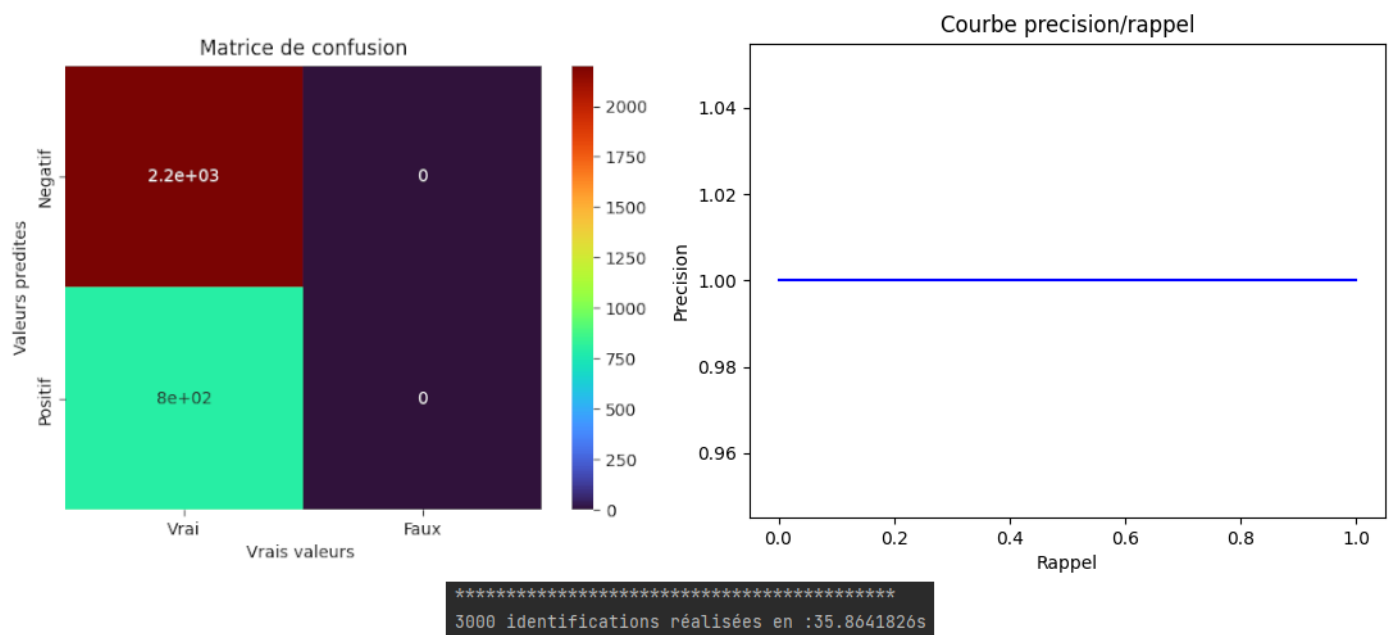
Identification échouée

Identification réalisée en 0.02540900000001045s
*****
```

Identification échouée



Statistiques pour 1 posture sur 6 des individus de la BD | Sans masques



Statistiques pour 6 posture sur 6 des individus de la BD | Sans masques

## C – Bibliographie

- [1] Effectif Irit  
<https://www.journaldunet.com/solutions/reseau-social-d-entreprise/1192757-carte-de-france-des-laboratoires-d-intelligence-artificielle/>
- [2] Tuto eigenfaces 1  
<http://blog.manfredas.com/eigenfaces-tutorial/>
- [3] Tuto eigenfaces 2  
<https://silanus.fr/sin/?p=785>
- [4] eigenfaces articles  
<https://sites.cs.ucsb.edu/~mturk/Papers/mturk-CVPR91.pdf>
- [5] Irit  
<https://www.irit.fr/le-laboratoire/presentation-du-laboratoire/>
- [6] Reva  
<https://www.irit.fr/departement/calcul-intensif-simulation-optimisation/reva/>