

Compte-rendu 3 de Projet

Reconnaissance et reconstruction de visages

Laurent Fainsin
Florian Guilloux
Lisa Larger

Département Sciences du Numérique
Première année
2020 — 2021

Table des matières

Question 1	3
Question 2	6
Question 3	9
Question 4	11
Question 5	12

Question 1

En utilisant le script matlab `eigenfaces_part3.m`, on peut ainsi calculer les eigenfaces d'individus masqués et non masqués (en utilisant dans un premier temps la fonction `eig`) :

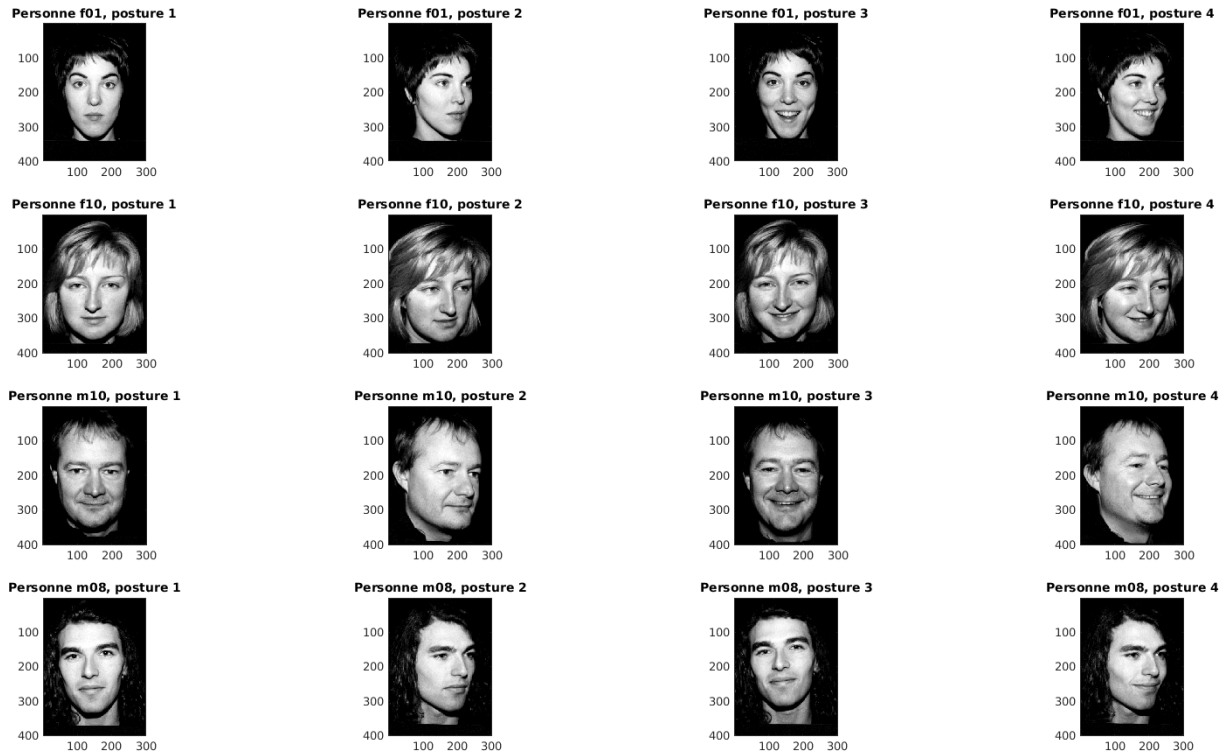


FIGURE 1 – Individus (non masqués) utilisés pour la création des eigenfaces

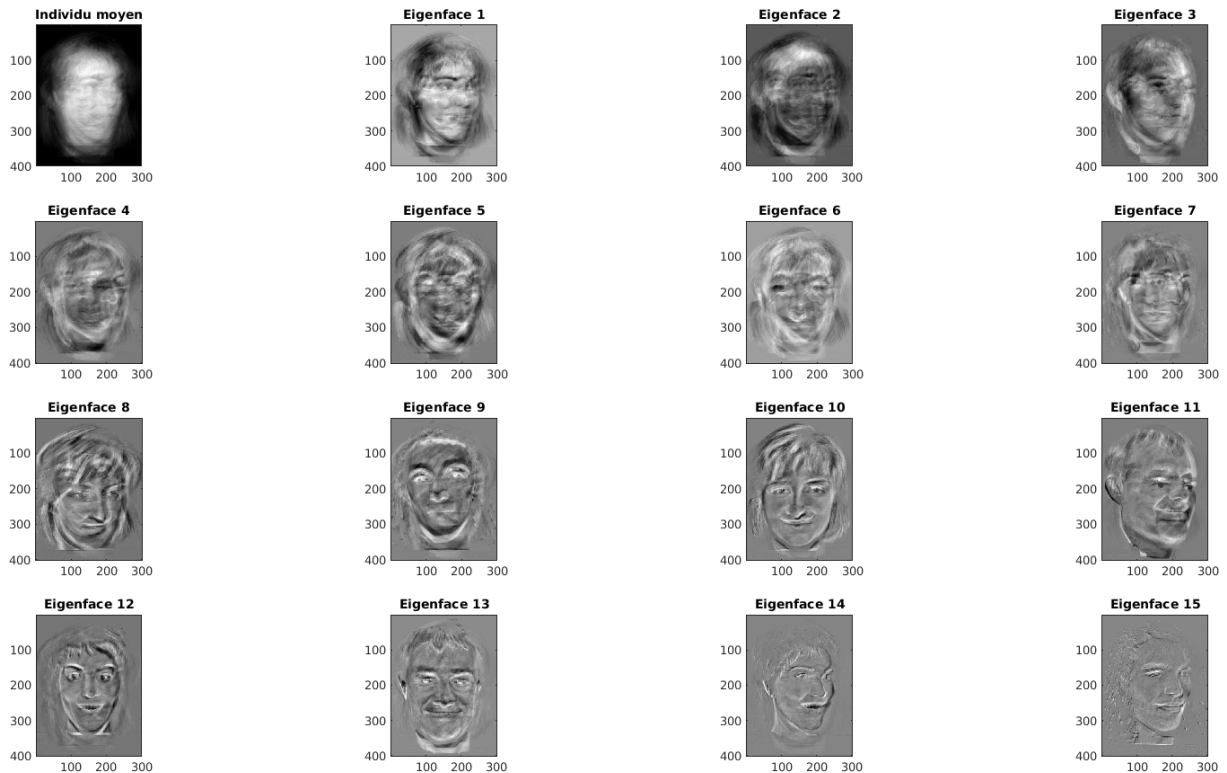


FIGURE 2 – Eigenfaces (non masquées)

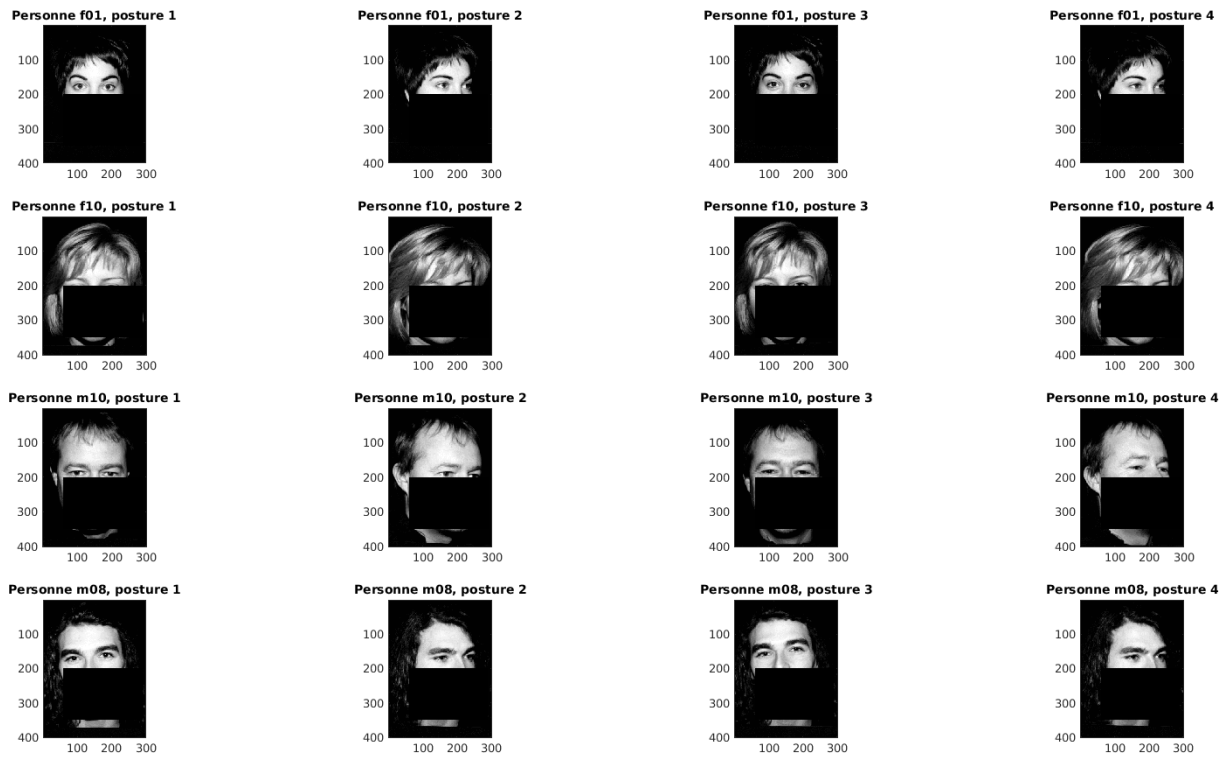


FIGURE 3 – Individus (masqués) utilisés pour la création des eigenfaces

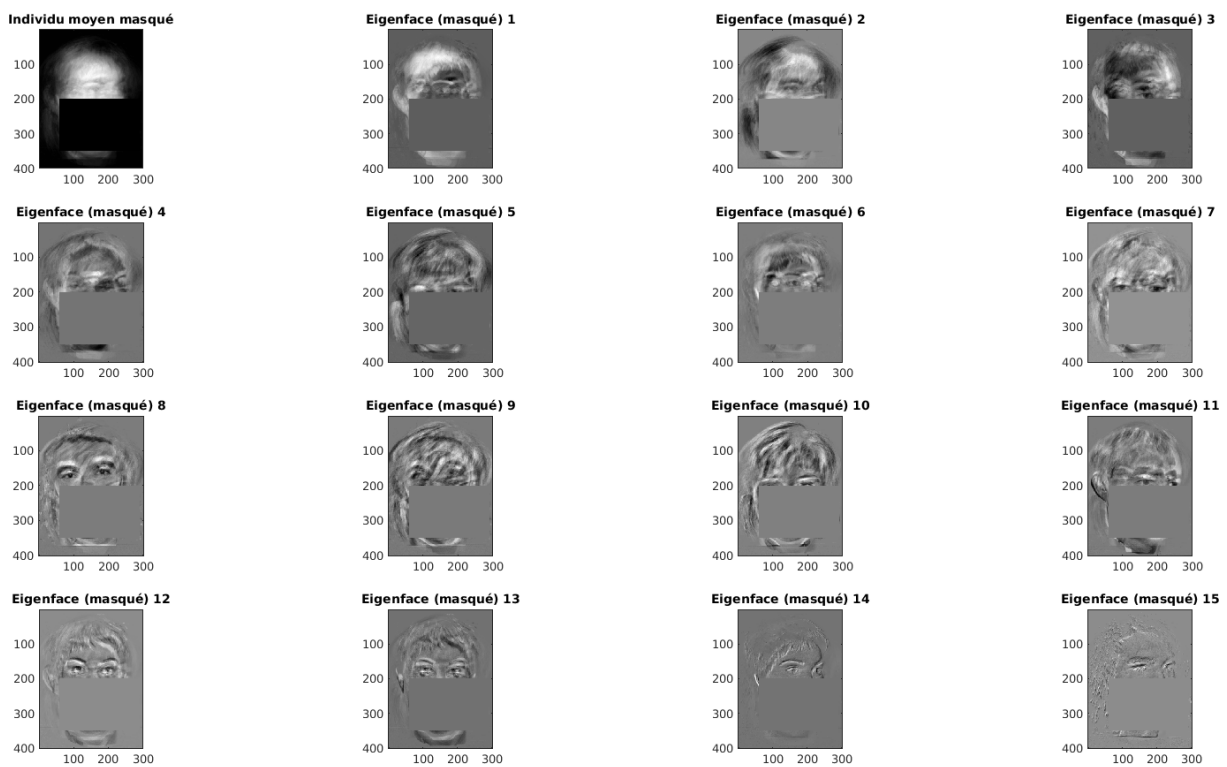


FIGURE 4 – Eigenfaces (masquées)

On peut de même si l'on le souhaite utiliser les fonctions que nous avons créées lors de la partie 2 du projet à la place de eig. On retrouve ainsi les mêmes eigenfaces, à l'exception de certaines dont les niveaux de gris sont inversés.

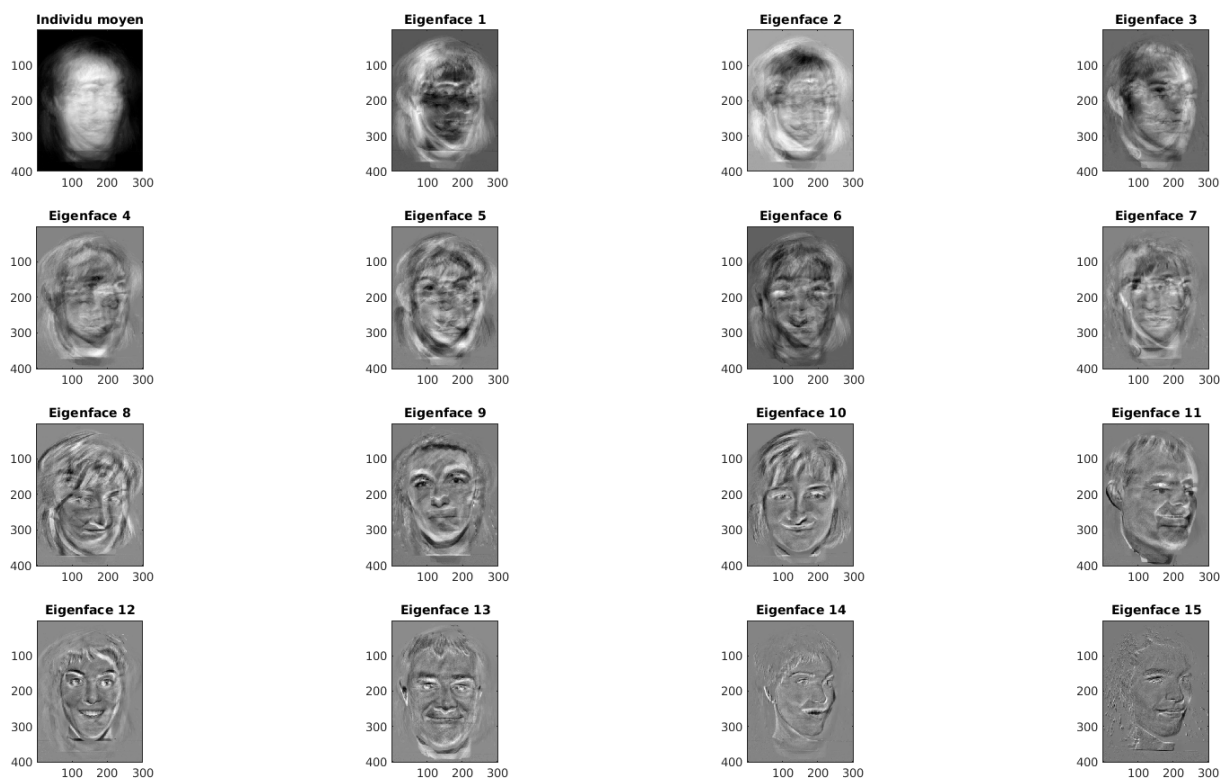


FIGURE 5 – Eigenfaces (non masquées) générées par souspace itération

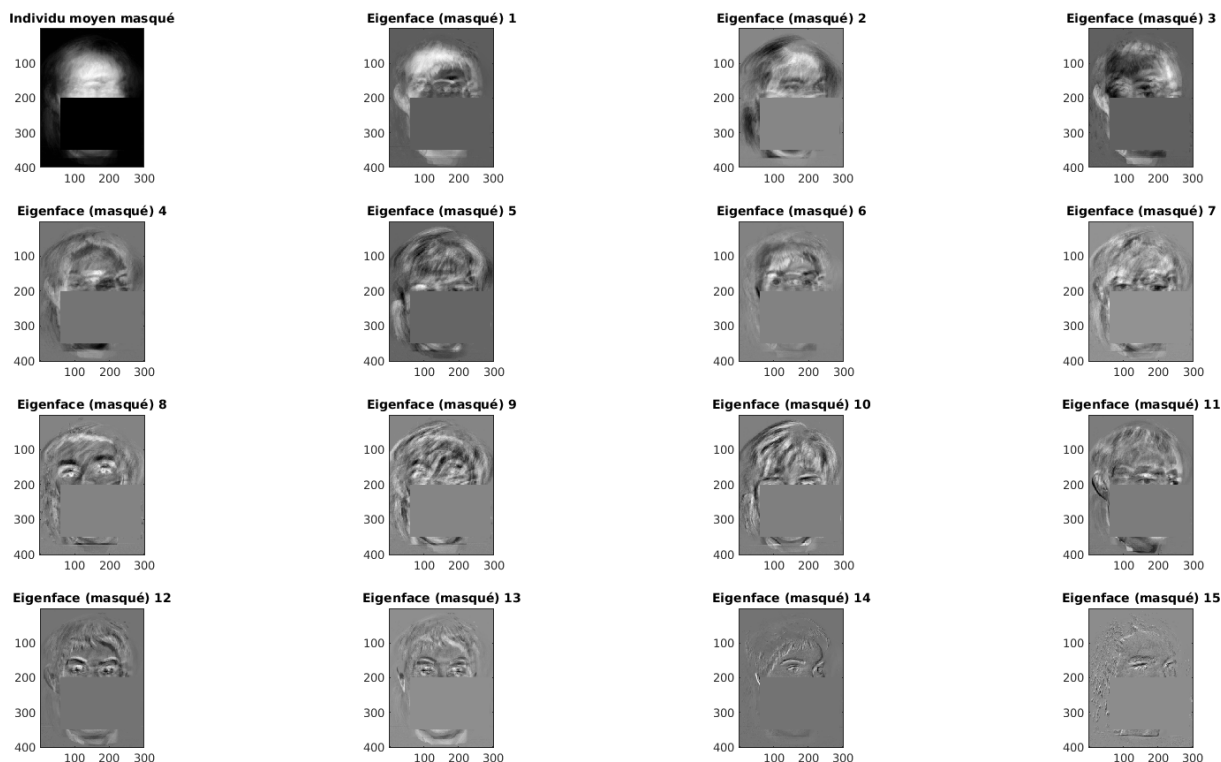


FIGURE 6 – Eigenfaces (masquées) générées par souspace itération

Question 2

À partir de ces eigenfaces on peut tracer la représentation des images dans la base des composantes principales. Ici pour des raisons évidentes de visualisation, seules les deux premières composantes sont utilisées.

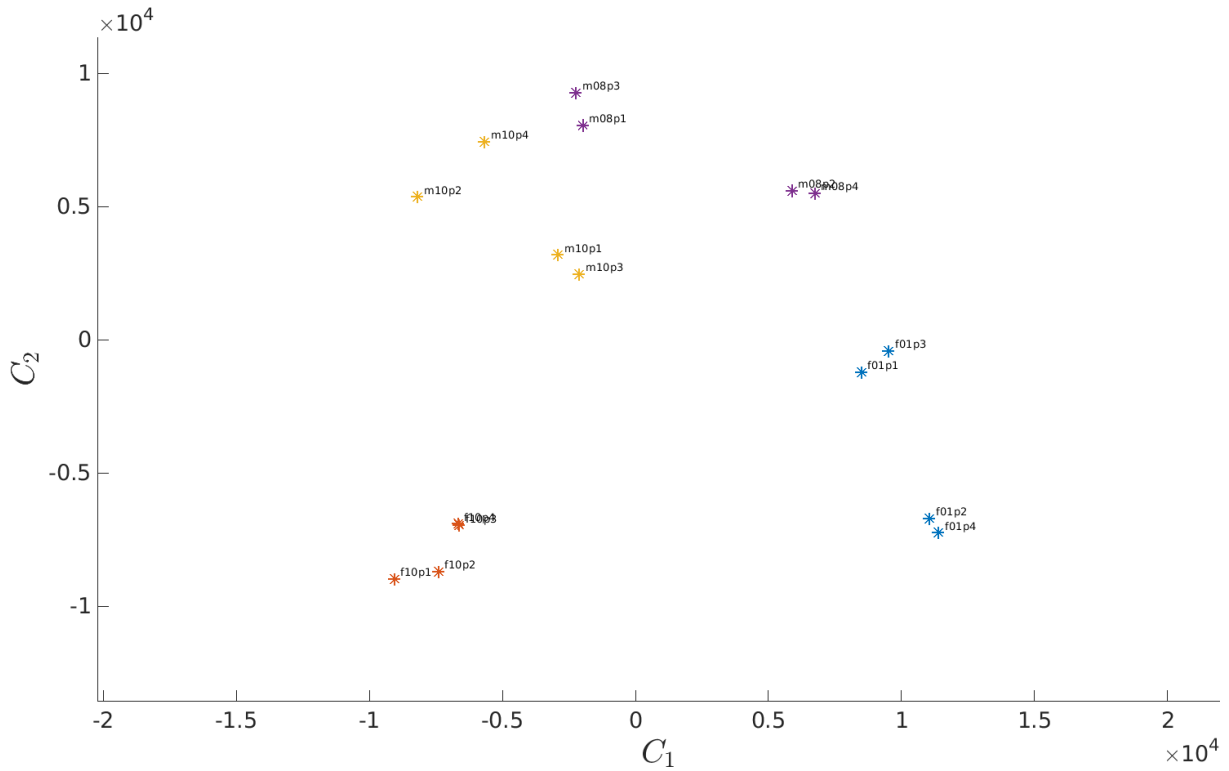


FIGURE 7 – ACP 2D des individus (non masqués)

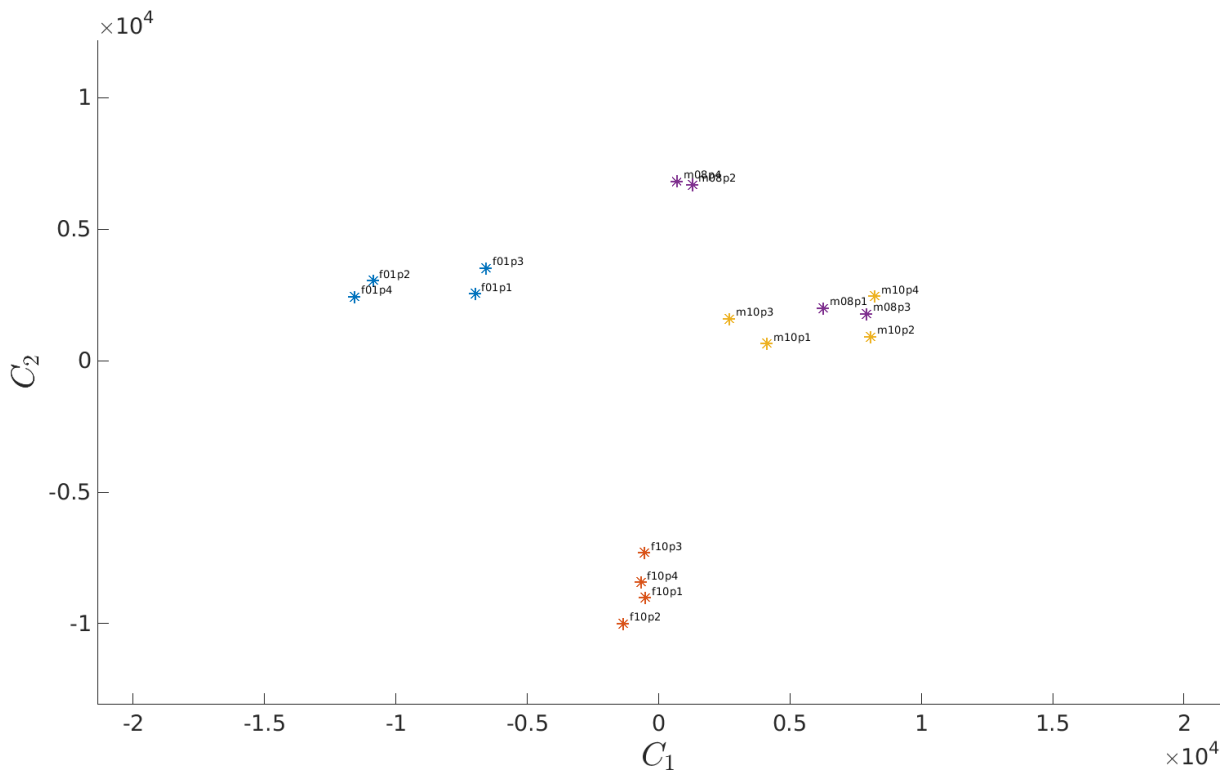


FIGURE 8 – ACP 2D des individus (masqués)

Cependant, on observe en traçant les contrastes de nos valeurs propres, que les deux premières dimensions regroupent 43% de l'information dans nos images. Ainsi si l'on souhaite par exemple 95% de l'information, il faudrait utiliser 12 composantes. Or un tracé en 12D est plutôt complexe à réaliser.

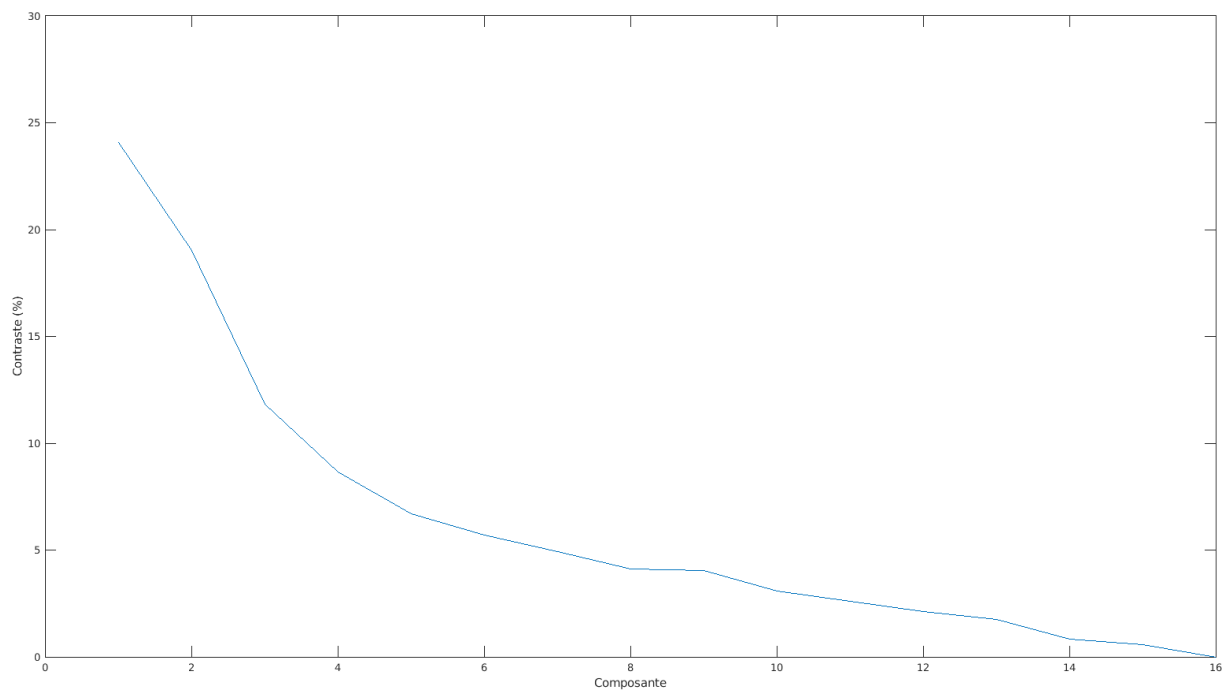


FIGURE 9 – Contraste des eigenvalues (non masquées)

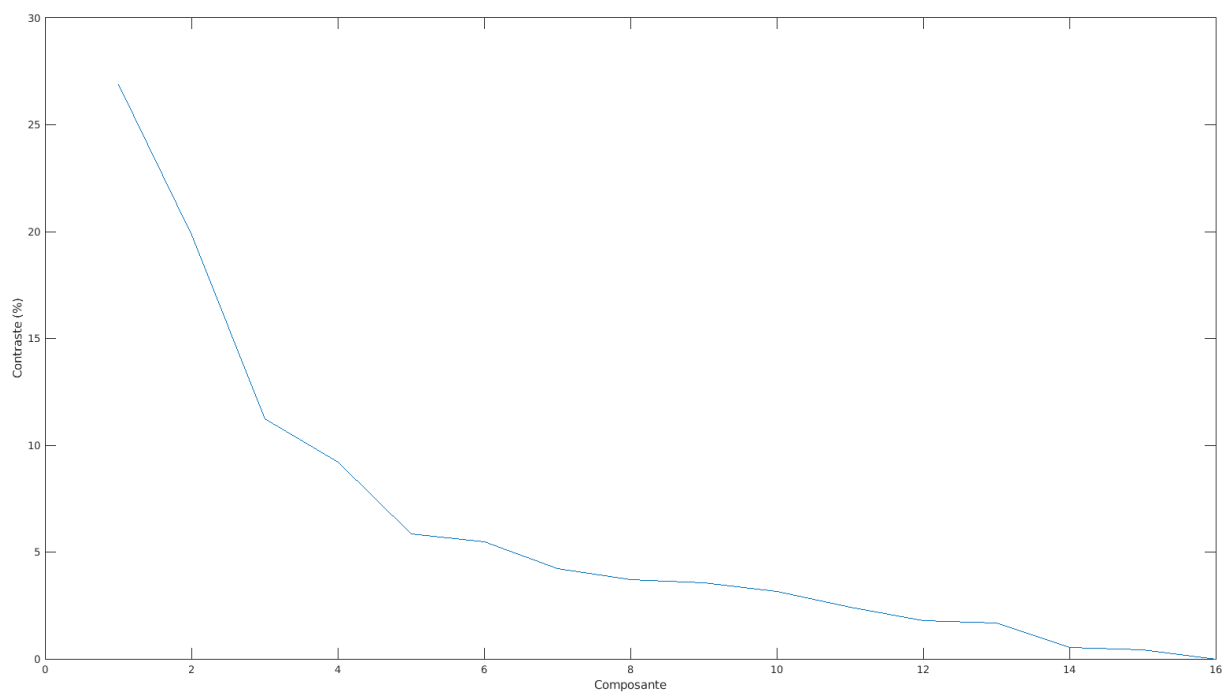


FIGURE 10 – Contraste des eigenvalues (masquées)

Voici tout de même deux figures représentant nos données dans les trois premières composantes principales. Nous invitons le lecteur à utiliser les scripts fournis pour tracer et explorer lui-même ces figures 3D.

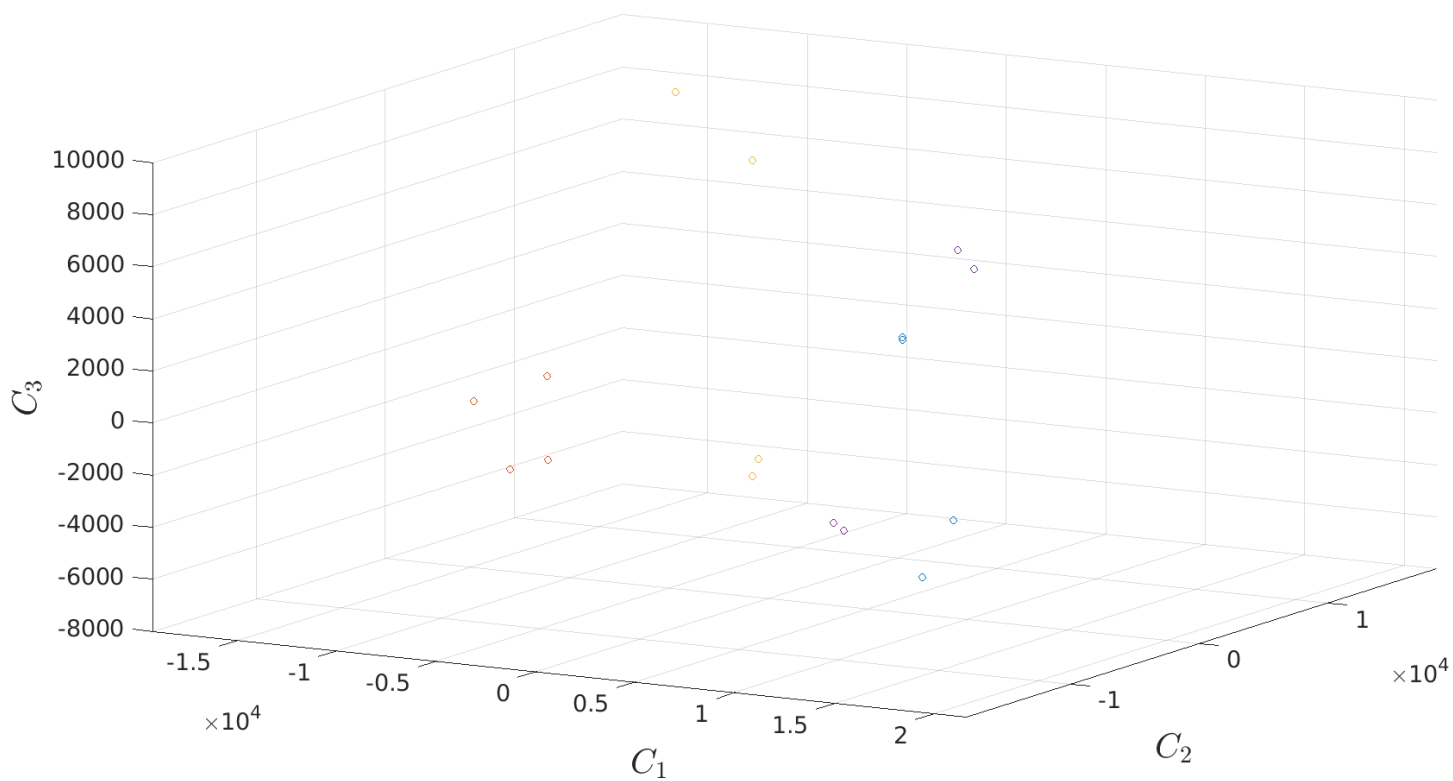


FIGURE 11 – ACP 3D des individus (non masqués)

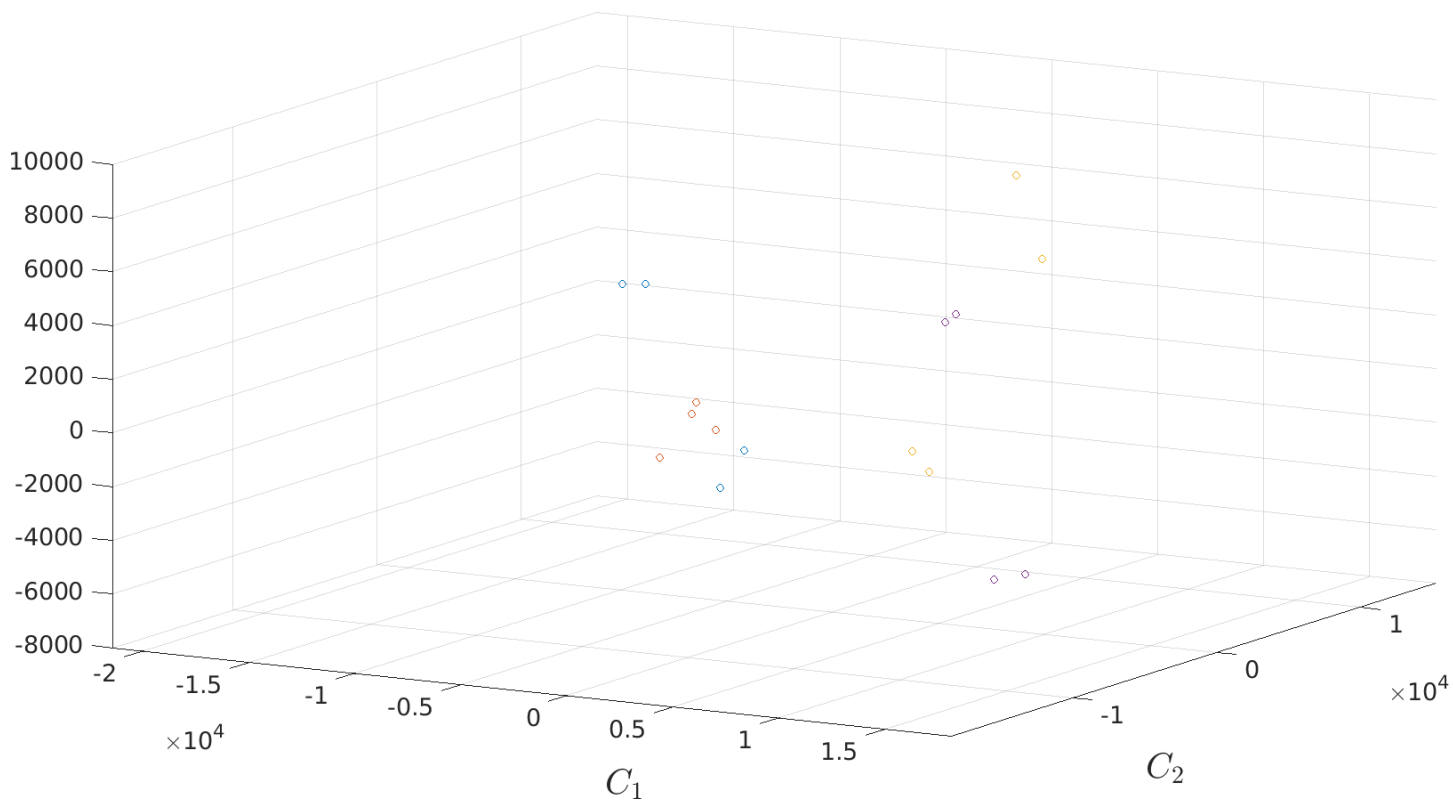


FIGURE 12 – ACP 3D des individus (masqués)

Question 3

En utilisant la méthode du 1 plus proche voisin, il est ainsi facile de classifier une nouvelle image, il suffit de la centrer, de la projeter sur les axes principaux et de trouver le point (associé à une image projetée) le plus proche de la projection. L'image de test est représentée sur la figure 14 par une croix verte et reliée à son plus proche voisin par une ligne bleue.



FIGURE 13 – Plus proche voisin (non masqué)

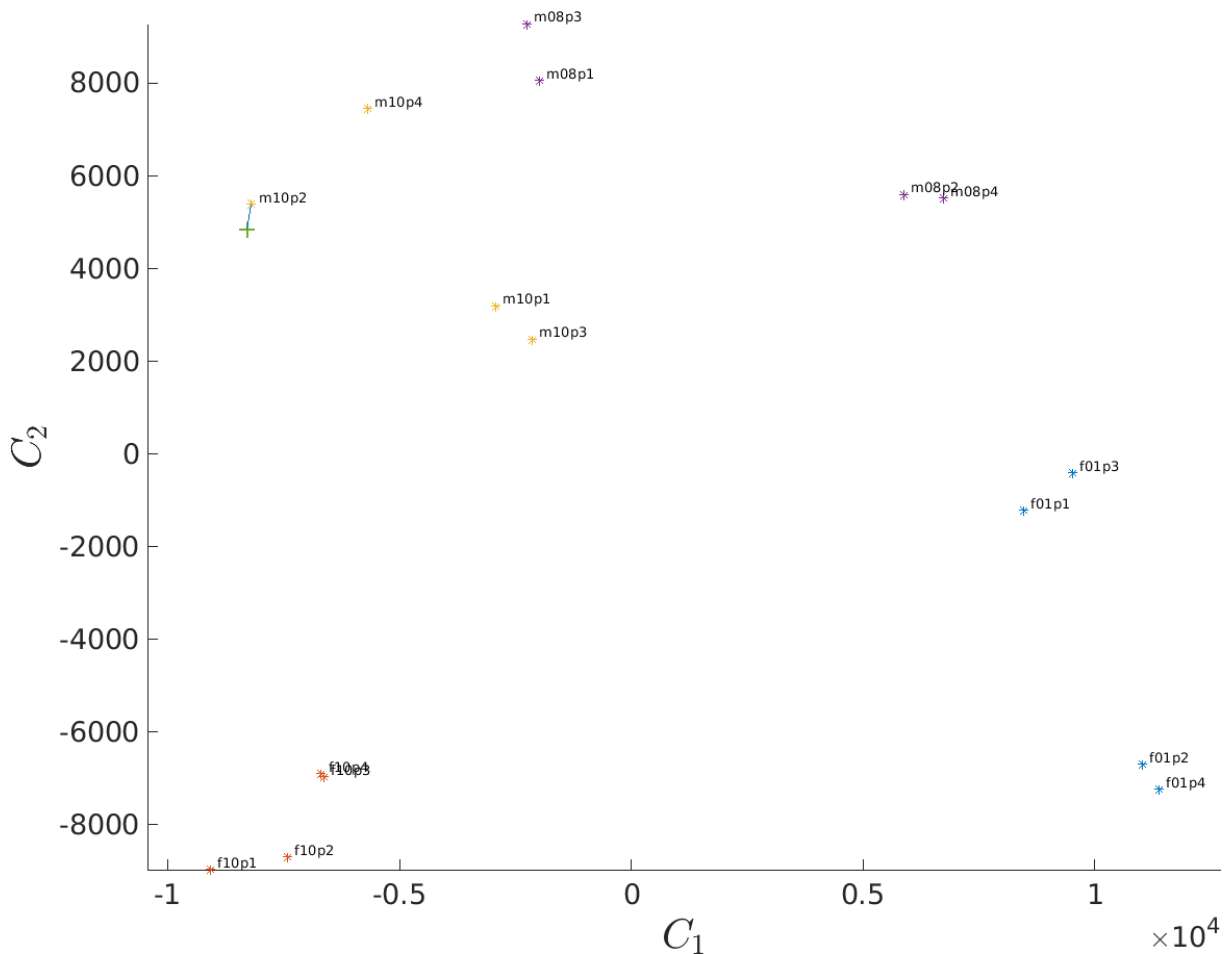


FIGURE 14 – ACP de l'individu de test (non masqué)

Pour reconnaître un individu masqué, seuls les images et notre modèle (nos eigenfaces) changent dans la méthode précédente, on obtient donc :

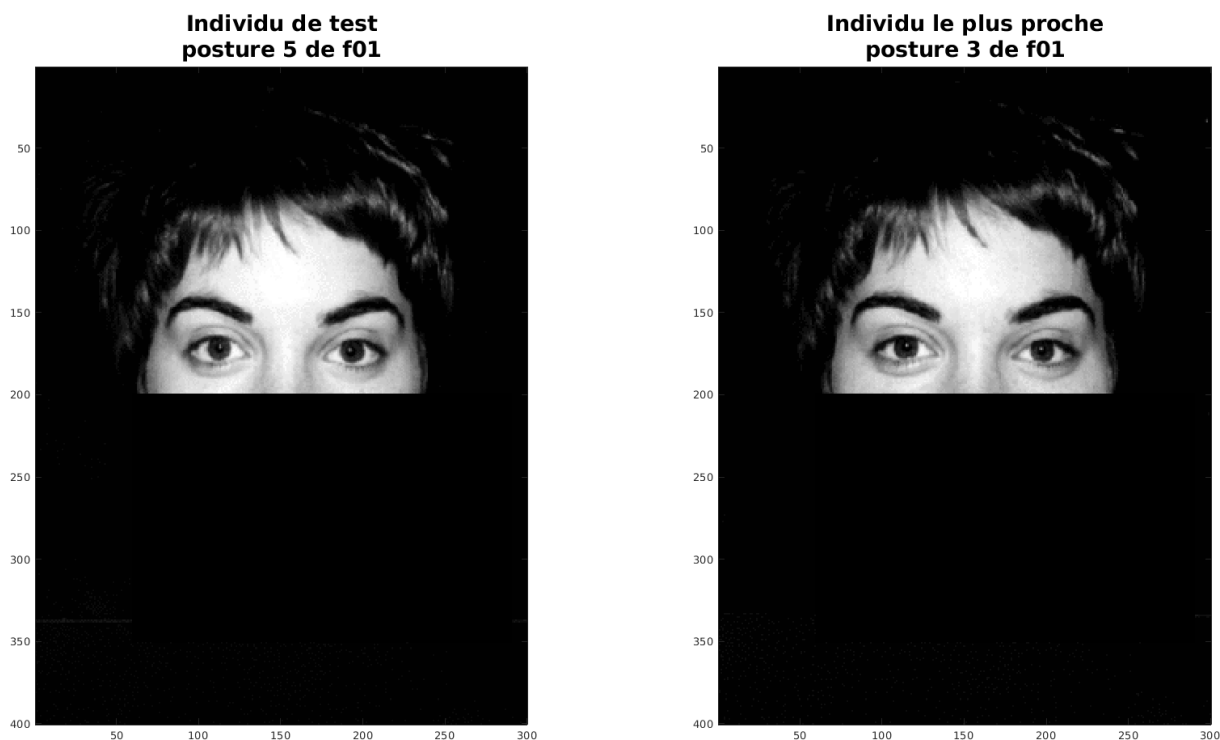


FIGURE 15 – Plus proche voisin (masqué)

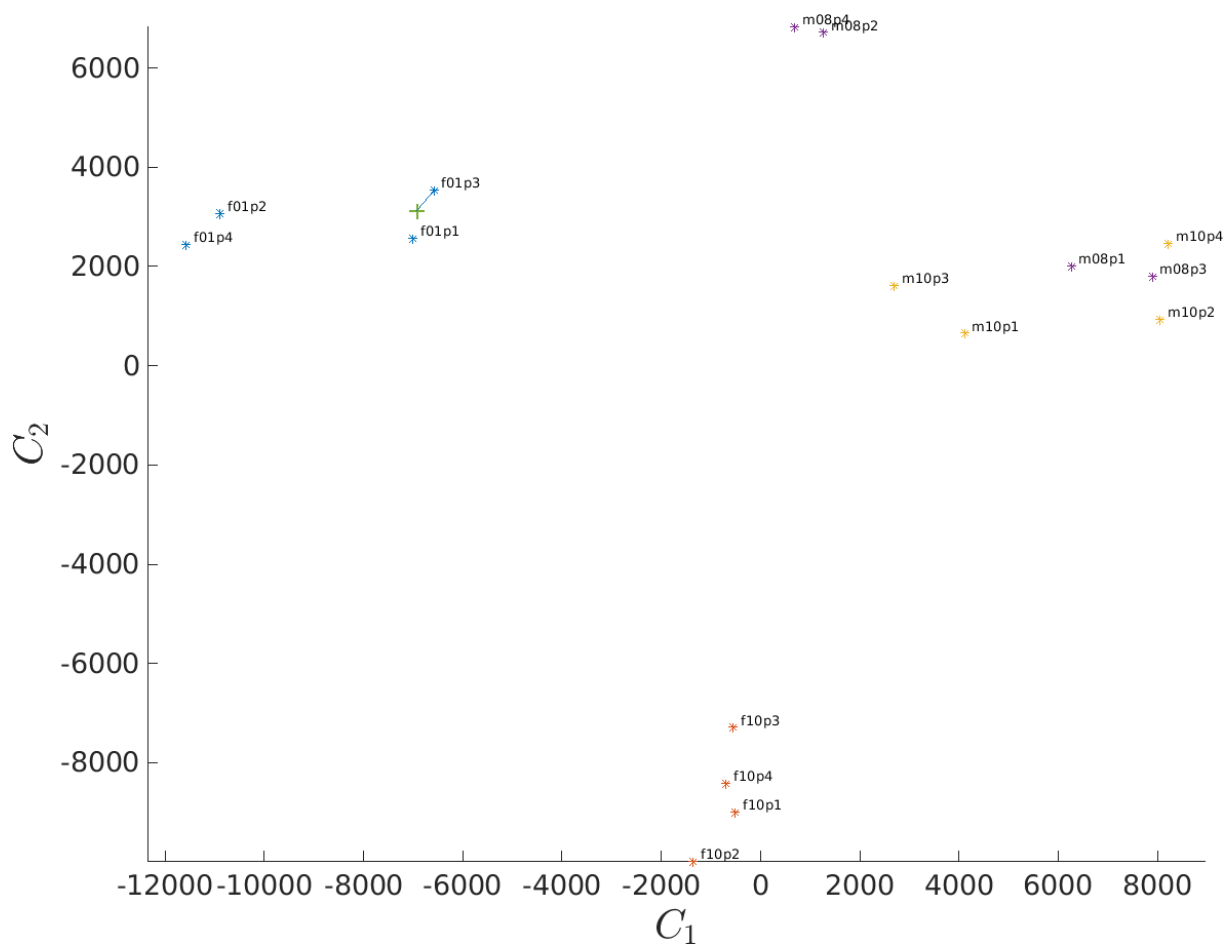


FIGURE 16 – ACP de l'individu de test (masqué)

Question 4

Finalement en utilisant l'image du plus proche voisin on peut ainsi essayer de reconstituer le visage de notre individu de test. On obtient généralement des résultats satisfaisant (et parfois très cused).

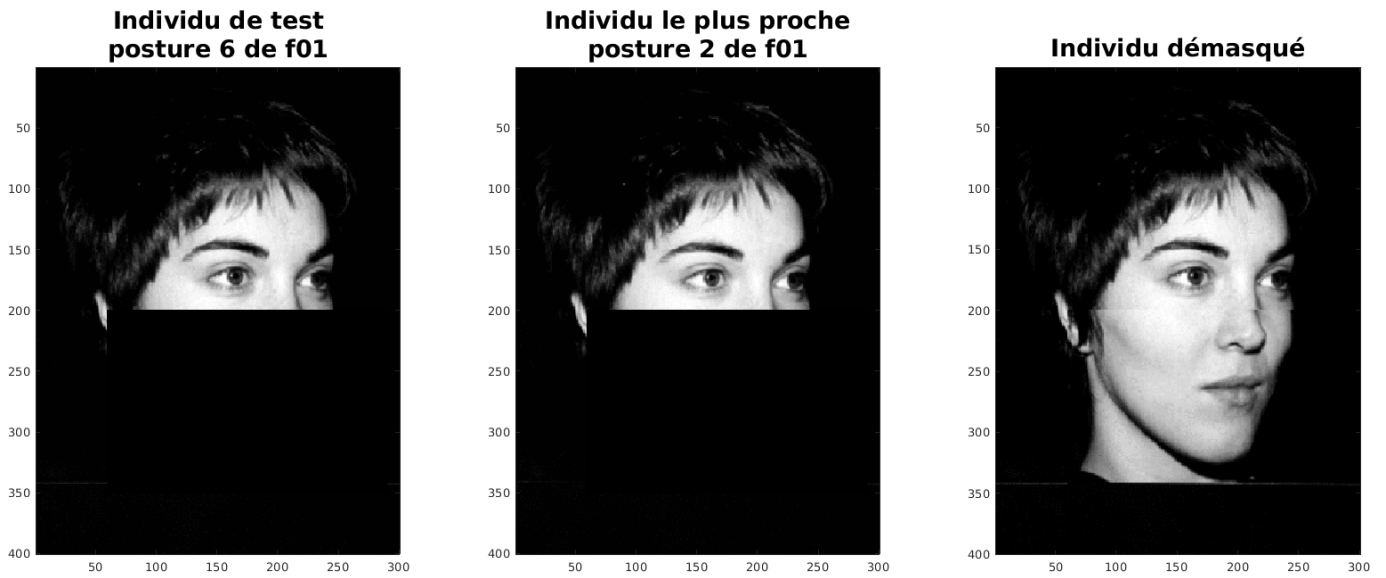


FIGURE 17 – Reconstitution de l'individu de test

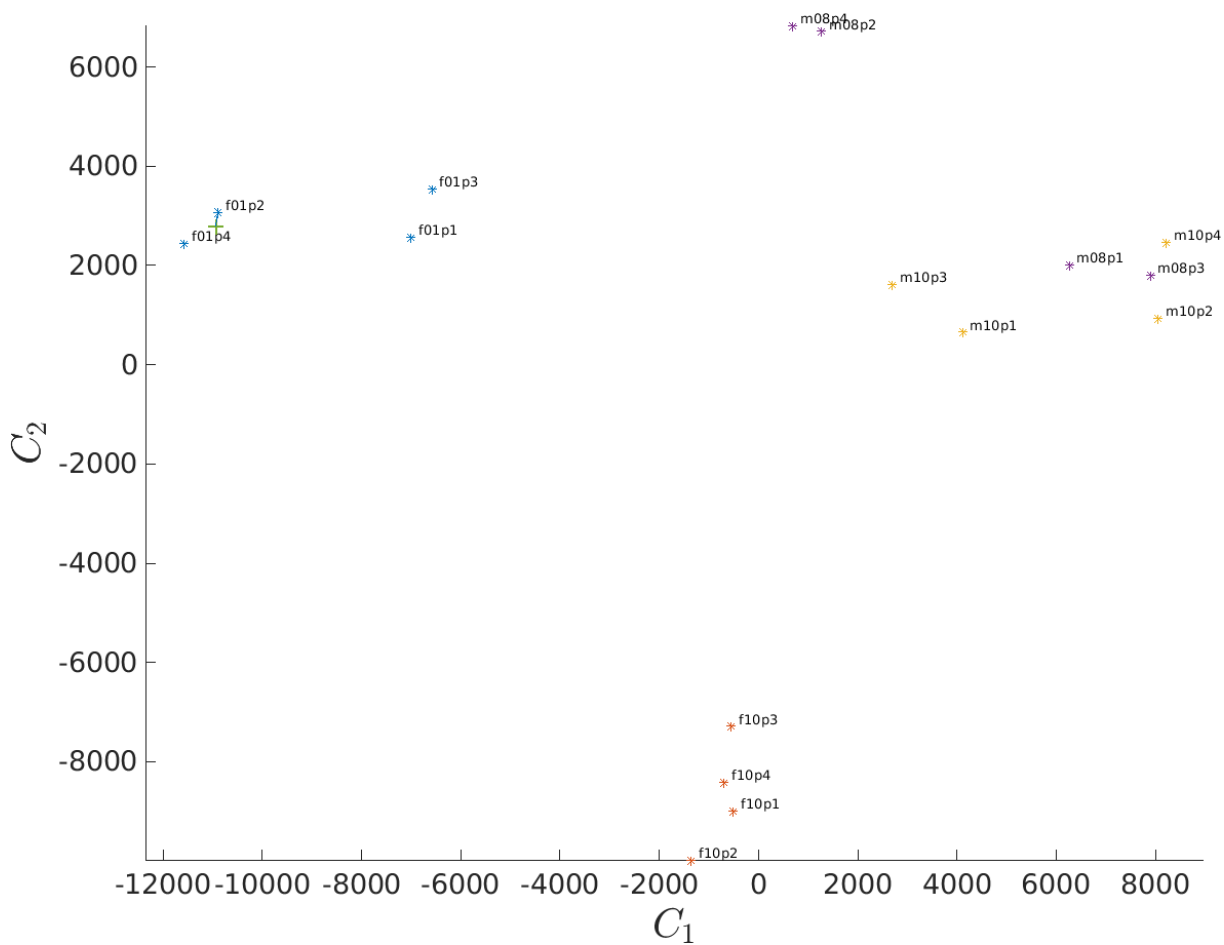


FIGURE 18 – ACP de l'individu de test (masqué)

