



A quoi tu ressembles sans ton masque ?

Séance-Projet 3 : Reconnaissance et Reconstruction de Visages

1 Rappel du contexte

Avec la pandémie de la Covid-19, nous avons appris à porter des masques quotidiennement. Désormais la moitié de notre visage est cachée nous laissant le regard pour seul moyen d'expression visible. Cette opération de cacher le visage s'apparente à un domaine en traitement d'images et de vidéos appelé "*inpainting*". En effet, les pixels correspondant à la zone du masque (modélisé par un rectangle) sont mis à 0. Et les *eigenfaces* permettent, entre autres, de restaurer la zone dégradée.

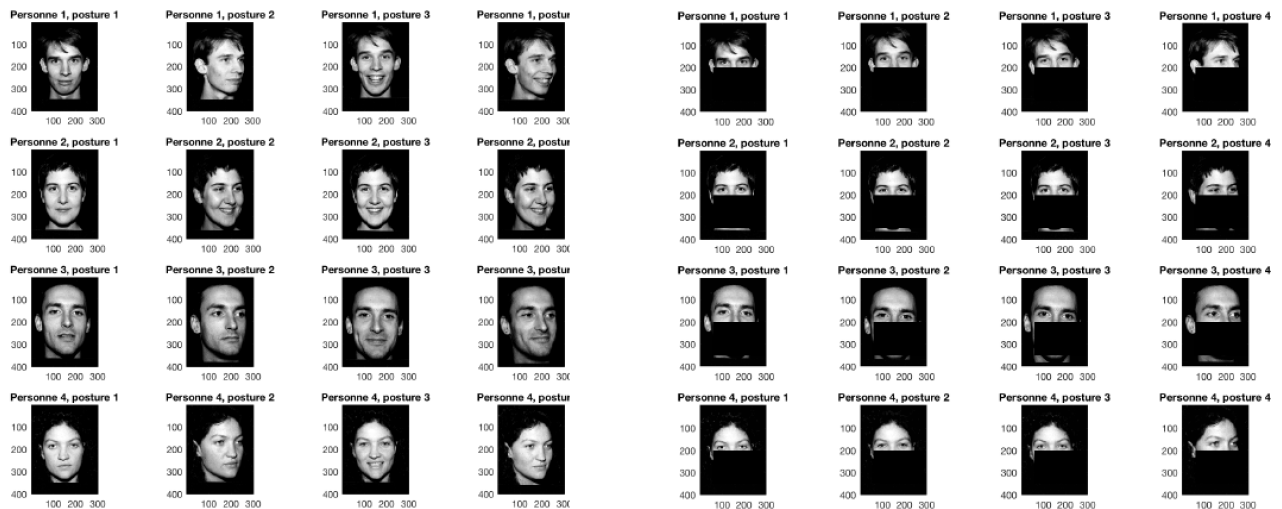


FIGURE 1 – Une base de visages et de visages masqués

Données manipulées

La base de données sur laquelle ce projet est appliqué est une collection d'images utiles pour mener des expériences en psychologie (*Psychological Image Collection at Stirling (PICS)*¹). Elle comporte 32 personnes (16 hommes et 16 femmes) avec 6 postures/expressions faciales (face, trois quart face et trois émotions différentes par posture). La FIGURE 1 montre une base de $n = 16$ visages présentant 4 personnes masquées et non masquées dans 4 positions/émotions différentes relativement à la caméra.

1. <http://pics.stir.ac.uk>

2 Retour sur les eigenfaces

Pour pouvoir remplir la zone du masque, nous nous appuyons sur le travail de M. Turk et A. Pentland intitulé «Eigenfaces for Recognition»² [1].

Il constitue une des applications les plus intéressantes et populaires de l'ACP au domaine de la reconnaissance de forme. Il s'agit tout simplement d'appliquer l'ACP à partir de données de très grandes tailles : des images de visages. Ici, l'ACP va nous permettre de réduire la dimension des images et de créer un système de reconnaissance grâce aux composantes principales pour retrouver le visage entier dans la base d'apprentissage le plus similaire au visage masqué et de permettre une reconstruction de la zone du masque (en faisant le lien avec le visage sans masque correspondant).

Pour réaliser cette reconnaissance, il est inutile et inopportun d'utiliser les $p = 120000$ niveaux de gris pour comparer l'image masquée avec chacune des images de la base d'apprentissage. L'ACP est donc un prétraitement qui consiste à réduire la dimension des données selon le schéma suivant où le prédicteur h prend en entrée la sortie de l'ACP :

$$\mathbf{x} \in \mathbb{R}^{120000} \rightarrow \boxed{\text{ACP}} \rightarrow \mathbf{c}(\mathbf{x}) \in \mathbb{R}^{q \ll 120000} \rightarrow \boxed{h} \rightarrow h(\mathbf{x})$$

Le vecteur caractéristique $\mathbf{c}(\mathbf{x})$ est tout simplement formé des q premières composantes principales résultant de la projection de l'image requête \mathbf{x} sur les q premiers vecteurs propres unitaires de la matrice Σ correspondant aux images masquées

Question 1 :

Le fichier `eigenfaces_part3.m` vous est fourni et réalise une partie du travail de la séance 1 (création de la base d'apprentissage, calcul des eigenfaces sans masque, ...).

1. lisez bien le code pour comprendre le rôle des variables définies,
2. complétez le code pour calculer les «eigenfaces» des visages avec masque en reprenant votre travail de la partie 1,
3. modifiez le code pour utiliser les fonctions de calcul de couples propres utilisant la méthode «Subspace Iteration» que vous avez développées dans la partie 2 du Projet.

3 Reconnaissance et reconstruction

3.1 Reconnaissance

A partir des composantes principales de la base des visages avec masque, on a défini ainsi $c(\mathbf{x})$, une représentation de \mathbf{x} , image requête (ou de test), dans cette base :

2. <https://www.mitpressjournals.org/doi/pdfplus/10.1162/jocn.1991.3.1.71>

$$\mathbf{c}(\mathbf{x}) \stackrel{\text{déf}}{=} \begin{bmatrix} \mathbf{w}_1^\top \mathbf{x} \\ \mathbf{w}_2^\top \mathbf{x} \\ \vdots \\ \mathbf{w}_q^\top \mathbf{x} \end{bmatrix}$$

On dit que $\mathbf{c}(\mathbf{x})$ est une représentation compacte préservant au mieux l'information contenue dans \mathbf{x} avec seulement q composantes principales. La prédiction h peut alors être définie de diverses manières. Cette fonction $h(\mathbf{x})$ est aussi appelée **classifieur**.

Parmi les possibilités les plus simples, on peut chercher l'image $\bar{\mathbf{x}}$ de la base dont la représentation compacte est :

1. la plus proche de celle associée à l'image requête \mathbf{x} , ce qui revient à minimiser la fonction $h_1(\mathbf{x})$ définie par :

$$h_1(\mathbf{x}) = \arg \min_{i \in 1 \dots n} d(\mathbf{c}(\mathbf{x}_i), \mathbf{c}(\mathbf{x}))$$

où $d(\mathbf{c}(\mathbf{x}_i), \mathbf{c}(\mathbf{x})) = \|\mathbf{c}(\mathbf{x}_i) - \mathbf{c}(\mathbf{x})\|_2$.

Le classifieur est ici la méthode des 1-plus proche voisin vue lors du TP4 d'Analyse de Données.

2. la plus vraisemblable de celle associée à l'image requête \mathbf{x} , ce qui revient à maximiser la fonction $h_2(\mathbf{x})$ définie par :

$$h_2(\mathbf{x}) = \arg \max_{i \in 1 \dots k} p(\mathbf{c}(\mathbf{x}) | \omega_i)$$

avec $p(\mathbf{c}(\mathbf{x}) | \omega_i)$ défini, en dimension d , par :

$$p(\mathbf{c}(\mathbf{x}) | \omega_i) = (2\pi)^{-d/2} (\det \Sigma_i)^{-1/2} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)\right).$$

où ω_i est une classe représentant un individu de la base avec μ_i et Σ_i respectivement la moyenne et la matrice de variance-covariance.

Le classifieur est dans ce cas la classification bayésienne vue lors du TP3 d'Analyse de Données.

Question 2 : définition des classifieurs

Le script `classification_avec_masque.m` va vous permettre d'appliquer les deux méthodes de classifications.

1. À partir du TP4, créer un fichier `kppv.m` correspondant à une fonction **kppv** permettant d'identifier la personne et la posture qui est la plus proche de la représentation compacte de l'image requête en utilisant la méthode des k-plus-proches voisins,
2. En utilisant `estimation_mu_Sigma.m` du TP3 et le fichier `gaussienne.m` fourni, créer un fichier `bayesien.m` correspondant à une fonction **bayesien** permettant d'identifier la personne la plus vraisemblable de la représentation compacte de l'image requête en utilisant la classification bayésienne.

Il faudra bien entendu s'adapter à la situation présente avec les données représentées sous forme compacte avec un nombre de variables différentes des TP ainsi qu'un nombre de classes, pour la classification bayésienne, qui n'est pas limité à 3 (les 3 types de fleurs) mais peut varier (le nombre de personnes que vous utilisez dans votre base d'apprentissage).

Nous vous conseillons de bien réfléchir sur la façon dont vous allez représenter les données pour être le plus adaptatif possible (chose pas vraiment possible avec le code que l'on vous a fourni pour le TP3 de classification bayésienne où nous avons des structures de données différentes pour les 3 types de fleurs).

Remarque : pour les nouvelles fonctions créées *kppv* et *bayesien*, vous donnerez une en-tête complète, décrivant les paramètres et les résultats.

Question 3 : reconnaissance avec masque

Le script `reconnaissance_sans_masque.m` tire aléatoirement une image de test, parmi les 32 personnes et les 6 postures faciales disponibles dans la base de données complète.

1. complétez le script pour trouver l'individu sans masque (personne+posture) dans la base d'apprentissage qui est le plus proche de l'image de test en utilisant les deux classifieurs,
2. calculez la matrice de confusion pour les deux classifieurs avec la fonction matlab *confusion-mat*,
3. identifiez les paramètres de vos classifieurs et essayez de trouver des valeurs optimales à ces paramètres dans la situation "sans masque".

Question 4 : reconnaissance sans masque

Refaites les différentes étapes de la question 3 dans la situation "avec masque"

(script `reconnaissance_sans_masque.m` recopié en `reconnaissance_avec_masque.m` et adapté).

3.2 Reconstruction

La reconstruction peut être réalisée en remplaçant la zone correspondant au masque par la zone de l'image de la base d'apprentissage de visages entiers la plus proche dans l'espace défini par les eigenfaces masqués.

Voici deux exemples de reconstruction d'images après reconnaissance :

- personne présente dans la base d'apprentissage mais avec une posture/émotion différente (Figure 2)
- personne ne se trouvant pas dans la base d'apprentissage (Figure 3)

Question 5 : reconstruction à partir des visages avec masque

En partant du script `reconnaissance_avec_masque.m` (que vous recopierez dans un fichier `reconnaissance_reconstruction.m`), développez le code permettant de reconstruire un visage de test masqué.

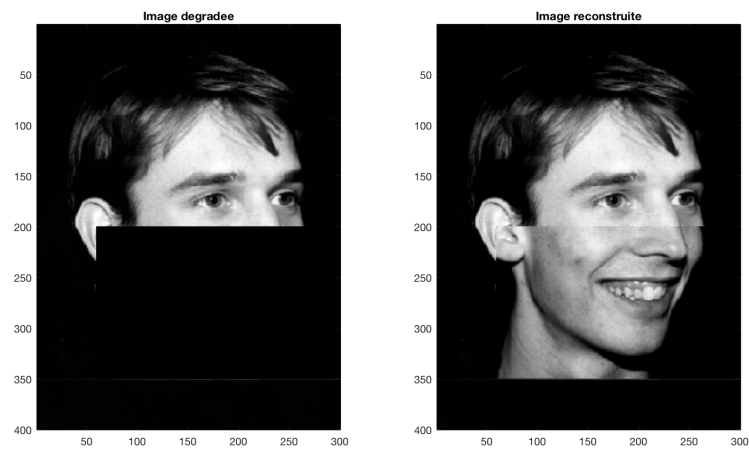


FIGURE 2 – Personne présente dans la base d'apprentissage mais avec une posture/émotion différente

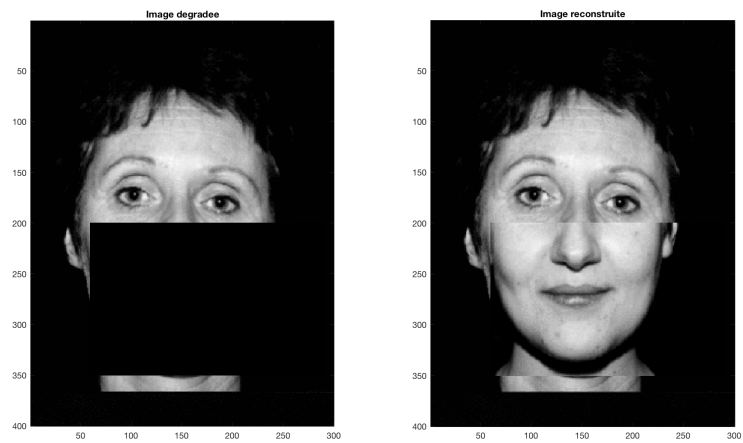


FIGURE 3 – Personne ne se trouvant pas dans la base d'apprentissage

4 Évaluation des classifieurs

Question 6 : évaluation

Réaliser une étude pour évaluer la qualité des deux classifieurs. Pour cela reprenez le travail réalisé aux question 3 et 4 mais en considérant cette fois-ci différentes bases d'apprentissage ; vous ferez varier les personnes, les postures, leurs nombres respectifs.

5 Documents à rendre

Voir Moodle

6 Bibliographie

- [1] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1) :71–86, 1991.

**N'oubliez jamais que vous avez à votre disposition
32 personnes et 6 postures**