

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

Option : ingénierie des systèmes d'informations (ISI)

Mémoire de Fin d'Etudes
En vue d'obtention de diplôme:
Master

Thème :

**FACTORISATION MATRICIELLE
NON NEGATIVE POUR LA
RECONNAISSANCE FACIALE**

Par :BAHLOUL RACHIDA

Encadré par :MR HENNI FOUAD

MR AMIR ABDSSAMED

Année Universitaire : 2018-2019



REMERCIEMENTS

En premier lieu, je remercie "DIEU" de m'avoir donné tant de patience et de puissance de pouvoir réaliser ce modeste travail.

Je remercie mes encadreur pour leurs soutient.

Je remercie les jurys pour avoir accepté d'examiner mon travail.

Je dédie ce modeste travail:

À mon marie pour leurs soutient et leurs patience

A mes très chers parents ,

A toute ma famille , à mes enfants :

Nesrine, Nasro, Islam et ma petite Rihab

A mes collègue,

*A toute personne ayant contribué à ce travail
de près ou de loin.*

BAHLOUL RACHIDA

Résumé :

Dans ce projet on a étudié la reconnaissance des visages Qui est considéré l'un des problèmes les plus difficiles à résoudre. Jusqu'à présent, plusieurs méthodes et approches sophistiquées ont été développées pour obtenir les meilleurs résultats de reconnaissance en utilisant des bases de données de visage spécifiques. Et on a étudié aussi le traitement d'image à des applications dans pratiquement tous les domaines. L'explosion numérique a exigé une réduction de la taille des images afin de limiter le stockage mais aussi de permettre une transmission plus rapide dans les réseaux. La compression d'images a pour but de réduire la taille de l'image tout en minimisant la détérioration de celle-ci. Beaucoup d'algorithmes et de formats de compression ont été proposés. Ce rapport étudie méthode de compression basée sur la factorisation matricielle non négative (NMF) et composition en valeurs singulières (SVD) et ses applications. et on a aussi étudié la reconnaissance faciale basé aussi sur les deux méthodes (SVD) et (NMF) . finalement faire les tests et afficher les résultats et faire la comparaison entre eux. Des tests ont été effectués sur plusieurs images, et les résultats sont satisfaisants.

Abstract :

In this project we studied the recognition of faces which is considered one of the most difficult problems to solve. So far, several sophisticated methods and approaches have been developed to obtain the best recognition results using specific face databases. And we also studied image processing applications in virtually every field. The digital explosion required a reduction in the size of the images in order to limit the storage but also to allow a faster transmission in the networks. The purpose of image compression is to reduce the size of the image while minimizing the deterioration of the image. Many algorithms and compression formats have been proposed. This report investigates compression methods based on non-negative matrix factorization (NMF) and singular value composition (SVD) and its applications, and facial recognition also based on both (SVD) and (NMF) methods. finally do the tests and display the results and make the comparison between them. tests were performed on several images, and the results are satisfactory.

Keywords: Image processing, Image compression, Singular value decomposition, non-negative matrix factorization, facial recognition.

SOMMAIRE

Introduction générale	3
Chapitre I : La compression d'image	
I.1. Introduction.....	5
I.2. Définition d'une image.....	5
I.3. Image numérique.....	5
I-4. Caractéristiques d'une image numérique.....	5
I-4-1. Pixel.....	6
I-4-2. Dimension (définition).....	6
I-4-3. Bruit.....	6
I-4-4. Résolution.....	6
I-4-5. Histogramme de l'image.....	6
I-4-6. Contours et textures.....	6
I-4-7. Luminance.....	7
I-4-8. Contraste.....	8
I-4-9. Le poids de l'image.....	7
I-5. Les différents types d'images.....	8
I-5-1. Mode monochrome.....	8
I-5-2. Images aux Niveaux de Gris.....	8
I-5-3. L'image couleur.....	9
I-5-3-1. La représentation en couleurs réelles.....	9
I-5-3-2. La représentation en couleurs indexées.....	9
I-5-3-3. Autres modèles de représentation.....	9
I-6. Les formats d'image.....	9
I-6-1. Principaux formats de fichiers non compressés.....	9
I-6-2. Principaux formats de fichier compressés.....	10
I-7. La compression.....	11
I-7-1. La compression sans perte.....	11
I-7-2. La compression avec perte.....	11
I-7-3. Le but de la compression d'image.....	11
I-7-4. Caractéristiques des méthodes de compression.....	11
I-7-4-1. Rapport et taux de compression.....	10
I-7-4-2. Entropie.....	12
I-7-4-3. Mesure de distorsion.....	12
I-8. Conclusion.....	13
Chapitre II : La Décomposition En Valeur Singulières (SVD)	
II-1. Introduction.....	15
II-2. La décomposition en valeurs singulières SVD.....	15
II-2-1. Principe.....	16
II-2-2. La décomposition.....	17
II-3. Théorie de La décomposition en valeurs singulières SVD.....	17
II-3-1. Processus de décomposition en valeurs singulières.....	17
II-3-2. Propriétés du SVD.....	18
II-3-3. Exemple de SVD.....	19
II-4. Méthodologie de svd appliquée en traitement d'images.....	20
II-4-1. Approche SVD pour la compression d'image.....	20
II-4-2. Mesures de compression d'image.....	21
II-4-3. Approche SVD pour la reconnaissance d'images de visage.....	21
II-4-3-1. Étapes à suivre pour effectuer la reconnaissance faciale avec (SVD).....	23
II-5. conclusion.....	25

Chapitre III : Factorisation matricielle non négative (NMF)

III-1.Introduction..... 27
 III-2.NMF : méthode et implémentations..... 28
 III-2-1. Principes..... 28
 III-2-2. Méthodes de résolution..... 29
 III-3. Algorithme de Lee et Seung..... 31
 III-4. Méthode des moindres carrés en alternance (ALS) 31
 II-5. Approche NMF pour la compression d'image..... 33
 III-6.Reconnaissance faciale avec factorisation matricielle non négative (NMF) 33
 III-6-1. Représentation et l'apprentissage 33
 III-6-2. Le test..... 34
 III-6-3. Étapes à suivre pour effectuer la reconnaissance faciale avec NMF..... 34
 III-7.Conclusion..... 37

Chapitre IV : CONCEPTION ET IMPLEMENTATION

IV-1.Introduction..... 39
 IV-2 .Bases de données..... 39
 IV-2-1. La Base de donnée « ORL » 39
 IV-2-2. La base de données "YALE"..... 41
 IV-3. Aspect matériel 42
 IV-4. Outils de développement..... 42
 IV-5.La Compression d'image 43
 IV-5-1.La compression d'image par SVD..... 43
 IV-5-2.La compression d'image par NMF..... 46
 IV-6.La reconnaissance faciale 48
 IV-6-1.La reconnaissance faciale par SVD..... 48
 IV-6-2.La reconnaissance faciale par NMF..... 52
 IV-7-Conclusion..... 55

LISTES DES FIGURES

Figure I-1 : Image avec et sans bruit..... 6
 Figure I-2 : Image et histogramme associé..... 7
 Figure I-3 :Image avec texture..... 7
 Figure II-1 : Illustration de factorisation de A à USV^T 18
 Figure II-2 : Organigramme de la reconnaissance faciale par SVD..... 24
 Figure III.1 : Mise à jour multiplicative non croissant 30
 Figure III.2 : Alternance des moindres carrés (ALS)vs Inexact Alternance des moindres carrés (IALS)..... 33
 Figure III.3:L'image la plus à gauche est l'image original du visage et l'image reconstruire à l'aide des images de base(W) 35
 Figure III.4 : Organigramme de la reconnaissance faciale par NMF..... 36
 Figure IV-1Bases de donnée des images « ORL » 40
 Figure IV-2-La 1 ère et la 35 ème personne d'ORL..... 40
 Figure IV-3- : base de données des images « YALE »..... 41
 Figure IV-4 :La 2^{ème} et La 14^{ème} personne de la base « YALE » 42
 Figure IV-5 Exemples d'images utilisées pour tester la Compression d'image par SVD..... 43
 Figure IV.6 : erreur entre l'image originale et l'image compressée..... 44
 Figure IV-7Exemples d'images utilisées pour tester la Compression d'image par NMF..... 46

SOMMAIRE

Figure IV.8 : erreur entre l'image originale et l'image compressée.....	47
Figure IV-9 Ensemble des images d'apprentissage S.....	49
Figure IV-10. Image du visage moyen calculé d'après d'ensemble des images d'apprentissage..	49
Figure IV-11 Image test et résultat de reconnaissance faciale par SVD.....	50
Figure IV-12 Image test et résultat de reconnaissance faciale par SVD.....	51
Figure IV-13. Image test et résultat de reconnaissance faciale par SVD.....	51
Figure-IV-14. Ensemble des images d'apprentissage f_i	52
Figure IV-15 L'image la plus à gauche est l'image originale du visage et l'image reconstruite à partir (W).....	53
Figure IV-16. Image test et résultat de reconnaissance faciale par NMF.....	53
Figure IV-17. Image test et résultat de reconnaissance faciale par NMF.....	54
Figure IV-18 Image test et résultat de reconnaissance faciale par NMF.....	54

LISTE DES TABLEAUX

Tableau IV-1. Résumé du résultat pour la Compression d'image par SVD.....	44
Tableau IV-1. Résumé du résultat pour la Compression d'image par NMF.....	47
Conclusion générale	56
Bibliographie	57
Annexe	60

Liste des abréviations :

- SVD** : Singular Value Decomposition
‘ En français Décomposition en valeur singulière
- DIP** : Digital Image Processing
‘ En français Traitement d’image numérique
- PCA** : Principal Component Analysis
‘ En français Analyse en composante principale ACP.
- MSE** : Mean Square Error
‘ En français Erreur quadratique moyenne.
- C_R** : Compression ratio
‘ En français Ratio de compression.
- NMF** : Nonnegative Matrix Factorization
‘ En français Factorisation dematrice non négative

Index des notations

\mathbb{R}	: L'ensemble des nombres réels.
\mathbb{R}_+	: Ensemble des nombres réels positifs.
\mathbf{I}_n	: Matrice d'Identité ou Matrice unité est une matrice carrée avec des 1 sur la diagonale et des 0 partout ailleurs.
X	: vecteur.
$A = (A_{ij})$: La matrice dont A_{ij} est le terme de la $i^{\text{ème}}$ ligne et de la $j^{\text{ème}}$ colonne.
$A_{:j}$: La $j^{\text{ème}}$ colonne de la matrice A .
$A_{i:}$: La $i^{\text{ème}}$ ligne de la matrice A .
$A^T = A_{ji}$: La matrice transposée de A dont A_{ij} est le terme de la $j^{\text{ème}}$ ligne et de la $i^{\text{ème}}$ colonne.
$\text{rg}(A)$: Le rang de la matrice A définit la dimension du sous-espace vectoriel engendré par les colonnes $A_{:j}$.
$\text{Tr}(A)$: La trace d'une matrice carrée A est définie comme la somme étant de ses coefficients diagonaux.
$\text{Im}(A)$: L'image de la matrice A .
$\ A\ _2$: La norme 2 de la matrice A .
$\ A\ _F$: La norme de Frobenius de la matrice A .
\circ	: Le produit matriciel de Hadamard.
$\langle A, B \rangle$: Le produit scalaire de les matrices A et B .

INTRODUCTION GENERALE

Introduction générale :

L'image est un moyen le plus efficace pour communiquer, chacun peut analyser l'image à sa manière, pour en dégager une impression et en extraire des informations précises. Le traitement d'image est l'ensemble des méthodes et techniques opérant sur l'image, dont le but est d'améliorer son aspect visuel.

La compression d'image est une discipline importante dans le domaine du traitement numérique de l'image. Elle traite des techniques de réduction de stockage requises pour enregistrer une image ou la bande passante requise pour la transmission. L'objectif de la compression d'image vise à réduire le manque de pertinence et la redondance des données d'image, pour optimiser ainsi l'espace de stockage et augmenter le taux de transmission sur les pages Web.

La reconnaissance de formes (ou parfois reconnaissance de motifs) est un ensemble de techniques et méthodes visant à identifier des motifs à partir de données brutes afin de prendre une décision dépendant de la catégorie attribuée à ce motif. On considère que c'est une branche de l'intelligence artificielle qui fait largement appel aux techniques d'apprentissage automatique et aux statistiques.

La reconnaissance faciale est un domaine de la vision par ordinateur consistant à reconnaître automatiquement une personne à partir d'une image de son visage. C'est un sujet particulièrement étudié en vision par ordinateur, avec de très nombreuses publications et brevets, et des conférences spécialisées.

Plusieurs méthodes et approches sophistiquées ont été développées pour obtenir les meilleurs résultats de reconnaissance en utilisant des bases de données de visage spécifiques. En raison de ce grand nombre de méthodes et de bases de données, il n'existe pas de moyen uniforme d'établir la meilleure méthode car presque toutes ont été conçues pour fonctionner avec certaines situations de visage spécifiques. Même si certaines de ces méthodologies ont conduit à la mise au point d'un grand nombre de systèmes commerciaux de reconnaissance faciale.

L'objectif de ce projet est d'étudier la compression d'image et la reconnaissance faciale en appliquant des méthodes basées sur des concepts mathématiques solides. Il s'agit de la SVP (la décomposition en valeurs singulières), et de la NMF (factorisation matricielle non-négative). Il s'agit d'implémenter ces méthodes et de conduire des expérimentations en utilisant des bases de données d'images existantes.

Ce rapport est organisé en quatre chapitres. Le premier chapitre présente les notions les plus pertinentes sur le traitement d'image, ainsi que les formats classiques de compression d'image. Le second chapitre est dédié à la présentation du principe de la méthode de décomposition en valeurs singulières (SVD) et leurs approches de compression d'image et de reconnaissance faciale. Le troisième chapitre présente la factorisation matricielle non négative (NMF) : principe et approche compression d'image et aussi que la reconnaissance faciale. Le quatrième chapitre est consacré à la conception et implémentation des techniques présentées ainsi qu'aux expérimentations. Nous terminons avec une conclusion générale et des perspectives.

CHAPITRE I:

La Compression D'Image

I-1.Introduction :

L'image constitue l'un des moyens les plus utilisés par l'homme pour communiquer avec autrui. C'est un moyen de communication universel dont la richesse du contenu permet aux êtres humains de tout âge et de toute culture de se comprendre.

C'est aussi le moyen le plus efficace pour communiquer, chacun peut analyser l'image à sa manière, pour en dégager une impression et en extraire des informations précises.

De ce fait, le traitement d'image est l'ensemble des méthodes et techniques opérant sur l'image, dont le but est d'améliorer son aspect visuel. Il se définit comme l'ensemble des tâches destinées à extraire de l'image des informations qualitatives et quantitatives.

I-2. Définition de l'image :

L'image est une représentation d'une personne ou d'un objet par la peinture, la sculpture, le dessin, la photographie, le film...etc. C'est aussi un ensemble structuré d'informations qui, après affichage sur l'écran, ont une signification pour l'œil humain.

Elle peut être décrite sous la forme d'une fonction $I(x,y)$ de brillance analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et I est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation.

I-3. Image Numérique :

Contrairement aux images obtenues à l'aide d'un appareil photo (analogique), ou dessinées sur du papier, les images manipulées par un ordinateur sont numériques (représentées par une série de bits). L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter.

La numérisation d'une image est la conversion de celle-ci de son état analogique (distribution continue d'intensités lumineuses dans un plan xOy) en une image numérique représentée par une matrice bidimensionnelle de valeurs numériques $X(n,m)$ où : n, m sont les coordonnées cartésiennes d'un point de l'image et $X(n,m)$ le niveau de gris ou de couleur en ce point.

I-4. Caractéristiques d'une image numérique :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

I-4-1. Pixel :

Le pixel est la contraction de l'expression anglaise " picture element". Etant le plus petit point de l'image, le pixel est une entité calculable qui peut recevoir une structure et une quantification. Dans une image couleur (R.V.B.), un pixel peut être représenté sur trois octets : un octet pour chacune des couleurs : rouge (R), vert (V) et bleu (B).

I-4-2. Dimension :

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image.

I-4-3. Bruit :

C'est un signal qui lors de l'acquisition ou transmission vient s'ajouter à l'image. Il se matérialise par la présence, dans une région homogène, de valeurs plus ou moins éloignées de l'intensité de la région. Le bruit est le résultat de certains défauts électroniques du capteur et de la qualité de numérisation.

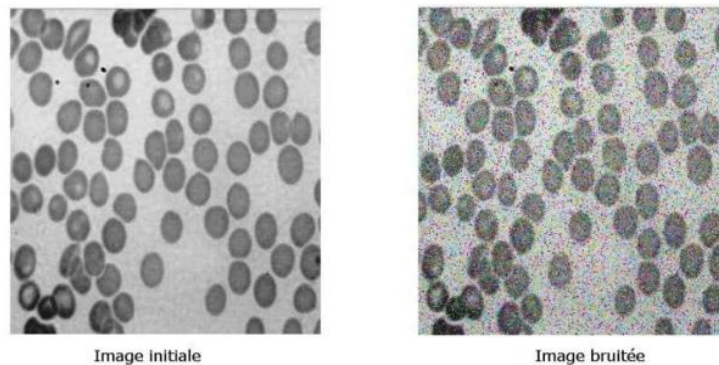


Figure I-1 : image avec et sans bruit

I-4-4. Résolution :

C'est la clarté ou la finesse de détails atteinte par un moniteur dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour désigner le nombre total de pixels affichables horizontalement ou verticalement sur un moniteur ; plus grand est ce nombre, meilleure est la résolution.

I-4-5. Histogramme de l'image :

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image.

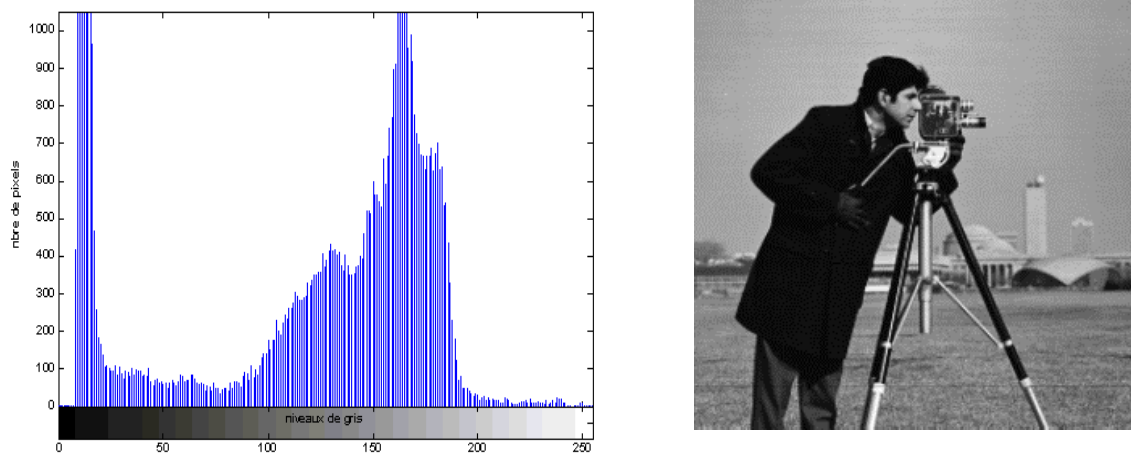


Figure I-2 : Image et histogramme associé

I-4-6. Contours et textures :

Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste à identifier dans l'image les points qui séparent deux textures différentes.

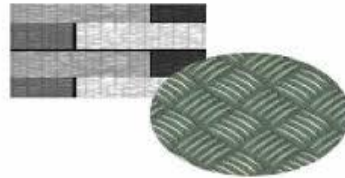


Figure I-3 : Image avec texture

I-4-7. Luminance :

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface. Pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. Une bonne luminance se caractérise par :

- ✓ Des images lumineuses (brillances).
- ✓ Un bon contraste : il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir. Ces images entraînent des pertes de détails dans les zones sombres ou lumineuses.
- ✓ L'absence de parasites.

I-4-8. Contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images. Si $L1$ et $L2$ sont les degrés de luminosité respectivement de deux zones voisines $A1$ et $A2$ d'une image ; le contraste C est défini par le rapport :

$$C = \frac{L1 - L2}{L1 + L2}$$

I-4-9. Poids de l'image :

Le poids d'une image se détermine en fonction de ces 3 paramètres : dimensions, résolution et nombre de couleurs. Il se mesure en octets.

I-5. Les différents types d'images :

Il existe différentes catégories d'images selon le nombre de bits sur lequel est codée la valeur de chaque pixel.

I-5-1. Mode monochrome :

Le mode monochrome est le plus simple ; chaque pixel y est soit allumé [Blanc], soit éteint [Noir]. L'image obtenue n'est pas très nuancée. Alors pour convertir une image couleur en mode monochrome il faut d'abord passer par le mode niveaux de gris.

I-5-2. Images en Niveaux de Gris :

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images en niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise entre 0 et 255. Par convention, la valeur zéro représente le noir (intensité lumineuse nulle) et la valeur 255 le blanc (intensité lumineuse maximale). Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la "couleur" de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux.

I-5-3. L'image couleur :

Il existe plusieurs types de représentations des images couleurs, parmi celle-ci nous trouvons :

I-5-3-1. La représentation en couleurs réelles :

Elle consiste à utiliser 24 bits pour chaque point de l'image. Huit bits sont employés pour décrire la composante rouge (R), huit bits pour la composante verte (V) et huit bits pour la composante bleu (B). Il est ainsi possible de représenter environ 16,7 millions de couleurs différentes simultanément.

I-5-3-2. La représentation en couleurs indexées :

Pour réduire la place occupée par l'information de couleur, on utilise une palette de couleurs attachée à l'image. On parle alors de couleurs indexées : la valeur associée à un pixel ne véhicule plus la couleur effective du pixel, mais renvoie à l'entrée correspondant à cette valeur dans une table (ou palette) de couleurs appelée look-up table (ou LUT en anglais), dans laquelle on dispose de la représentation complète de la couleur considérée.

I-5-3-3. Autres modèles de représentation :

Le modèle RVB représentant toutes les couleurs par l'addition de trois composantes fondamentales, n'est pas le seul possible. Il en existe de nombreux autres. L'un d'eux est particulièrement important. Il consiste à séparer les informations de couleurs (chrominance) et les informations d'intensité lumineuse (luminance). Il s'agit du principe employé pour les enregistrements vidéo. La chrominance est représentée par deux valeurs (selon des modèles divers) et la luminance par une valeur.

I-6. Les formats d'image :

Un format d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler. Voici quelques formats :

I.6.1. Principaux formats de fichiers non compressés :

Ces formats de fichiers utilisent en général beaucoup de mémoire. De par leur poids élevé, ils ne sont pas adaptés pour le web mais doivent être utilisés lorsqu'on a besoin de préserver la totalité des informations d'une image pour retravailler dessus par exemple.

✓ TIFF :

Le TIFF pour (Tagged Image File Format) a été mis au point en 1987. Le format TIFF est un ancien format graphique, permettant de stocker des images bitmap (raster) de taille importante (plus de 4 Go compressées), sans perte de qualité et indépendamment des plateformes ou

des périphériques utilisés (Device-Independent Bitmap, noté DIB). Il supporte différents types de compression autant avec que sans perte de données.

Le format TIFF permet de stocker des images en noir et blanc, en couleurs réelles (True color, jusqu'à 32 bits par pixels) ainsi que des images indexées, faisant usage d'une palette de couleurs.

✓ **BMP :**

Le BMP est un des formats les plus simples développés conjointement par Microsoft et IBM, ce qui explique qu'il soit particulièrement répandu sur les plateformes Windows et OS/2. C'est un format ouvert et non compressé. Sa taille rédhibitoire rend son utilisation en ligne difficile, mais sa grande compatibilité en fait un format de travail efficace. En BMP la couleur est codé en RGB (synthèse additive), le format lui-même supportant la palette 256 couleurs que le «true color».

I-6-2. Principaux formats de fichier compressés :

Ce sont les formats de fichiers qui permettent, selon un algorithme particulier, de gagner plus ou moins de mémoire en supprimant certaines informations peu ou non perceptibles par l'œil humain. Ils sont particulièrement adaptés à Internet, mais ne doivent pas être utilisés lors d'un travail de création d'une image (Par exemple, sous Photoshop) car chaque nouvel enregistrement détériore un peu plus le fichier. Ce format est le plus souvent utilisé pour exporter des images destinées à la visualisation sur Internet ou l'archivage

✓ **JPEG :**

Ce format est l'un des plus complexes, son étude complète nécessite de solides bases mathématiques, cependant malgré une certaine dégradation il offre des taux de compressions plus qu'intéressants.

JPEG est la norme internationale (ISO 10918-1) relative à la compression d'images fixes, notamment aux images photographiques. La méthode de compression est "avec pertes" et s'appuie sur l'algorithme de transformée en cosinus discrète DCT. Un mode "sans perte" a ensuite été développé mais n'a jamais été vraiment utilisé. Cette norme de compression a été développée par le comité JPEG (Joint Photographic Experts Group) et normalisée par l'ISO/JTC1 SC29. Ce type de compression est très utilisé pour les photographies, car il est inspiré des caractéristiques de perception visuelles de l'œil humain.

Le JPEG2000 est la norme internationale (ISO 15444-1). Elle apporte quelques améliorations au JPEG classique et notamment permet un réglage autorisant une compression sans perte ou encore la résistance aux erreurs de transmission. JPEG2000 est relative à la compression d'images qui s'appuie sur un mécanisme de compression par ondelettes.

✓ **GIF :** GIF (Graphic Information Format) est un format léger qui peut également contenir des animations. Une image GIF ne peut contenir que 2, 4, 8, 16, 32, 64, 128 ou 256 couleurs parmi 16.8 millions dans sa palette en mode RGB. Elle supporte également une couleur de transparence.

✓ **PNG et MNG** :Le PNG pour Portable Network Graphic(ISO 15948) a été développé par le W3C pour remplacer le GIF. Il surpasse ce dernier en ce qu'il n'est notamment pas limité à 256couleurs. De même, le format est ouvert et permet une bonne compression sans perte. Son utilisation est recommandée à l'instar du GIF pour les petits logos. Coté photo, s'il permet une compression sans perte, le poids de la photo n'est pas compétitif avec les formats JPEG. Précisons que le PNG ne gère pas l'animation mais un format dérivé, le MNG, y est destiné.

I-7. La compression d'image:

La compression de données consiste à obtenir des fichiers plus légers afin d'améliorer la vitesse de transfert sur internet ou limiter l'espace de stockage utilisé sur un disque dur. Il existe deux principaux types de compression.

I-7-1. La compression sans perte :

Appelée aussi « compactage »,cette solution consiste simplement à coder les données binaires de manière plus concise dans un fichier. Elle permet ainsi de retrouver la totalité des informations après une procédure de décompactage.

I-7-2. La compression avec perte :

Elle concerne essentiellement les fichiers de média (image, son, vidéo), et consiste en une « réduction » de l'information basée sur les limites humaines à percevoir ces médias. Puisque l'œil ne perçoit pas nécessairement tous les détails d'une image, il est possible de réduire la quantité de données de telle sorte que le résultat soit très ressemblant à l'original, voire identique, pour l'œil humain.[8].

I-7-3 Le but de la compression d'image :

Les images numériques sont des fichiers volumineux qui occupent beaucoup d'espace sur disque ou allongent considérablement les temps de transmission sur réseau. La compression d'images permet de réduire énormément la taille des images.

I-7-4 Caractéristiques des méthodes de compression :

I-7-4-1 Rapport et taux de compression :

Le rapport de compression est l'une des caractéristiques les plus importantes de toutes les méthodes de compression.Il représente le rapport entre le nombre de bits de la forme canonique au nombre de bits après codage :

$$C_R = \text{rapport} = \frac{\text{nombre bits avant compression}}{\text{nombre bits après compression}}$$

Le taux de compression est un pourcentage de l'espace obtenu après la compression par rapport à l'espace total requis par les données avant la compression.

$$Tc = \text{tauxc} = \left(1 - \frac{1}{\text{rapport de compression}}\right) * 100$$

Cela indique : Qu'un fichier compressé indique à sa taille originale aura un taux de compression de 0 %. Un fichier réduit à 0 octet, aura un taux de compression de 100%.

I-7-4-2. Entropie :

L'entropie $H(S)$ d'une source simple $[S]N$ associée à une loi de probabilité $[P]N$ est définie selon la formule suivante :

$$H(S) = - \sum_{i=1}^N (P(S_i) \log^2(P(S_i))) \text{ bits}$$

Dans une image, l'entropie est une grandeur qui caractérise la qualité de l'information que contient cette dernière. Par exemple, une image dont tous les pixels ont la même valeur contient très peu d'information car elle est extrêmement redondante, son entropie est faible.

En revanche, une image dont tous les pixels ont une valeur aléatoire contient beaucoup d'information, son entropie est forte.

En pratique, l'entropie d'une image numérique est inversement liée à la probabilité d'apparition des niveaux de gris dans l'image. Par définition, l'entropie d'ordre zéro H_0 est donnée par :

$$H_0 = - \sum_{k=1}^{2^R-1} (P(k) \log^2(P(k))) \text{ bpp}$$

Avec : $P(k)$ est la probabilité d'apparition des niveaux de gris dans l'image, k est la valeur de gris et R est le nombre de bits par pixels.

L'entropie H_0 d'une image originale fournit le débit minimal qu'il est possible d'atteindre par compression, pixel par pixel sans dégrader l'image, et par la même, un taux de compression sans perte maximal.

I-7-4-3. Mesure de distorsion :

Pour mesurer la distorsion entre l'image reconstruite et l'image originale (Mesure de la qualité visuelle de l'image reconstruite) on utilise l'Erreur Quadratique Moyenne MSE (Mean Square Error) ou le rapport signal à bruit PSNR (Peak Signal to Noise Ratio).

Etant donnée une image originale composée de pixels $a_i (i=1, \dots, N)$ et l'image décodée composée de pixels $a'_i (i=1, \dots, N)$.

Alors l'erreur quadratique moyenne est donnée par :

$$MSE = 1/N \sum_{i=1}^N (a_i - a'_i)^2$$

Le PSNR est donnée par :

$$PSNR = 10 \log_{10} \frac{(2^R - 1)^2}{MSE}$$

En compression d'images, le PSNR d'une image de taille 512 par 512 pixels, chaque pixel est codé sur 8 bits est défini par:

$$PSNR = 10 \log_{10} \frac{(255)^2}{MSE}$$

I-8.Conclusion :

Dans ce chapitre, nous avons présenté les généralités et les notions de base des images numériques d'une manière générale. Nous nous sommes intéressés aux terminologies et aux notions pertinentes dans le domaine des images numériques, Nous avons également présenté la nécessité de compresser des images ainsi que quelques formats classiques de compressions. Le but du chapitre suivant est d'aller plus loin dans la compression en utilisant la méthode SVD et NMF.

CHAPITRE II:

La Décomposition En Valeurs Singulières

II-1.Introduction :

La décomposition en valeurs singulières généralise la notion de valeurs propres aux matrices rectangulaires. C'est un outil de factorisation de telles matrices, et peut être vu comme le procédé de diagonalisation pour les matrices carrées. Nous utiliserons dans toute la suite de ce document le sigle SVD pour parler de la décomposition en valeurs singulières (Singular Values Decomposition pour les anglophones, acronyme largement répandu). Bien que la SVD s'applique aussi bien aux matrices réelles que complexes, nous ne traiterons qu'avec des matrices à coefficients réels qui sont celles rencontrées dans les divers champs d'application de ce calcul de SVD. Néanmoins, tout théorème ou définition générale liée à la SVD sera énoncé au sens large, et donc pour des matrices à coefficients complexes.

La réduction de matrices rectangulaires par le calcul de la SVD est assez classique dans la littérature d'aujourd'hui, puisque traitée depuis le milieu des années 1960 par l'informaticien Gene Howard Golub (auteur du très bon livre Matrix Computations[19]) et le mathématicien William Morton Kahan, qui proposent en 1965 le premier algorithme pour calculer la SVD. Cependant, le procédé d'approximation de rang faible consistant au calcul d'une SVD tronquée, bien connu également au 20^{ème} siècle, permet plus récemment de fournir des résultats en analyse de données avec par exemple la complétion de matrices aux données manquantes. Ce travail qui revient à résoudre un problème d'optimisation convexe est plutôt à la mode depuis le début des années 2000.

La compression d'images numériques a connu une évolution incessante, parallèlement à celle des télécommunications et du multimédia, depuis les années 60. Elle permet de réduire la taille d'une image dans le but d'augmenter la capacité des supports de stockages (limités en capacité) et d'optimiser l'utilisation de la bande passante d'un réseau. Depuis la normalisation de l'algorithme JPEG basé sur la transformée en cosinus discrète, le volume des données multimédias (son, image, vidéo, etc.) n'a cessé d'augmenter. La norme JPEG2000 basée sur la transformée par ondelettes a permis d'augmenter le taux de compression des images avec une qualité supérieure à celle de JPEG.

La SV consisté décomposer une matrice en un produit de 3 matrices U, S et V (S est appelée matrice des valeurs singulières). Chen ainsi qu'Abrahamsen [1] ont déjà proposé une méthode simple de compression d'images à niveaux de gris ne retenant que les k premières valeurs singulières. Des améliorations ont été proposées en utilisant l'algorithme SVD standard. D'autres applications de la décomposition en valeurs singulières comme la compression et la reconnaissance faciale ont montré que la SVD est utilisée dans plusieurs domaines de l'imagerie. En ce qui concerne les images en couleurs, Adams et Cooper [17] ont proposé une méthode qui applique la compression SVD à chaque composante R, V et B.

II.2.La décomposition en valeurs singulières SVD :

Une matrice est un tableau de nombres dont il est parfois difficile d'extraire les caractéristiques intéressantes pour résoudre un problème donné. Une stratégie efficace pour mettre en évidence les propriétés d'une matrice est de la décomposer (ou factoriser) en un

produit de matrices plus simples et dont les caractéristiques sont clairement identifiables et interprétables. La factorisation la plus générale, et peut-être la plus utile, est la SVD.

La théorie de la décomposition en valeurs singulières a été établie pour les matrices réelles carrées dans les années 1870 par Beltrami et Jordan et pour les matrices complexes par Autonne en 1902. Récemment, la décomposition en valeurs singulières a été utilisée dans différentes applications du traitement d'image telle que la compression, la dissimulation de l'information et la réduction du bruit.

Le traitement d'image est une forme de traitement d'information, dans lequel l'entrée est une image. Le traitement des images étudie comment transformer, stocker, récupérer l'image. Image digitale, le traitement est l'utilisation d'algorithmes informatiques pour effectuer un traitement d'image sur des images numériques. Beaucoup de techniques de traitement d'image ont été développées avec des applications comme le traitement d'image satellitaire à satellite imagerie, l'imagerie médicale, la reconnaissance d'objets, et l'amélioration de la photo. Avec la disponibilité d'ordinateurs et de processeurs rapides pour le traitement de signal dans les années 2000, le traitement numérique des images est devenue la forme la plus courante de traitement d'image, et est généralement utilisé parce que ce n'est pas seulement la méthode la plus polyvalente, mais aussi la moins chère.

II-2-1. Principe :

L'idée essentielle de la SVD est de décomposer la matrice de données en trois matrices simples : deux orthogonales et une diagonale. Du fait qu'elle produise une estimation aux moindres carrés de la matrice de données de même dimension et d'un rang inférieur.

L'un des avantages de la SVD est son pouvoir de réduction des données après leur blanchissement. En effet, cette technique fournit une description plus compacte des données contenues dans une matrice, exprimée par les premiers modes statistiques. Elle peut être considérée comme une méthode permettant de construire une partition de la variance d'une base de données, c'est à dire qu'elle fournit la base orthogonale qui maximise la variance au sens des moindres carrés.

La décomposition en valeurs singulières utilise la décomposition en valeurs propres d'une matrice semi définie positive obtenue par la multiplication d'une matrice par sa transposée, pour dériver une décomposition similaire applicable à toutes les matrices rectangulaires composées de nombres réels.

Toute matrice A de taille $m \times n$ de rang r peut être décomposée en une somme, pondérée de matrices unitaires $m \times m$ par Décomposition en Valeurs Singulières.

Les matrices U et V sont unitaires et A peut donc s'écrire :

$$A = USV^T = \sum_{i=1}^n (\sigma_i u_i v_i^T)$$

Où S est une matrice diagonale dont les r -premiers termes diagonaux sont positifs, tous les autres étant nuls. Les r -termes σ_i non nuls sont appelés valeurs singulières (SV) de A .

Avec : $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ et $\sigma_{r+1} \geq \sigma_{r+2} \geq \dots \geq \sigma_n = 0$

II-2-2. La décomposition :

Formellement, si A est une matrice rectangulaire, son SVD la décompose comme suit :

$$A = USV^T$$

U est la matrice des vecteurs propres normalisés de la matrice AA^T , c'est-à-dire $U^T U = I$. Les colonnes de S sont appelées les vecteurs singuliers gauches de A .

V est la matrice des vecteurs propres normalisés de la matrice $A^T A$, c'est-à-dire $V^T V = I$.

Les colonnes de V sont appelées les vecteurs singuliers droits de A .

S est la matrice diagonale des valeurs singulières.

La SVD a l'importante propriété de donner la meilleure approximation d'une matrice rectangulaire par une autre matrice de même dimension mais de rang inférieur, au sens des moindres carrés. Précisément, si « A » est de dimension $(m \times n)$ et de rang « r », donc « A » a « r » valeurs singulières non nulles. La décomposition en valeurs singulières (SVD) offre un nouveau moyen d'extraire des caractéristiques d'une image.

Les principales propriétés théoriques de la SVD relatives à la compression d'image sont :

- ✓ La SVD d'une image présente une bonne stabilité. Quand une petite perturbation est ajoutée à une image, une grande variance de ses (SV) ne se produit pas.
- ✓ Les valeurs singulières représentent les caractéristiques dominantes d'une image.

II-3. Théorie de La décomposition en valeurs singulières SVD :

II-3-1. Processus de décomposition en valeurs singulières :

La décomposition en valeurs singulières (SVD) est considérée comme étant un sujet important en algèbre linéaire par beaucoup de mathématiciens. SVD a beaucoup de valeurs pratiques et théoriques. La particularité de SVD est qu'il peut être effectué sur toute matrice (m, n) réelle. Disons que nous avons une matrice A avec m lignes et n colonnes, avec rang r et $r \leq n \leq m$. Alors la matrice A peut être factorisée en trois matrices :

$$A = USV^T \quad (1.1)$$

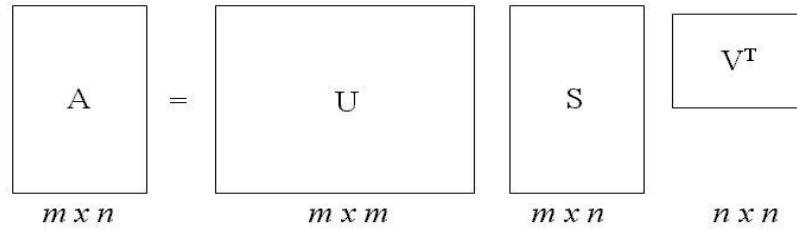


Figure II-1. Illustration de factorisation de A à USV^T

Où la matrice $U_{(m \times m)}$ est une matrice orthogonale

$$U = [u_1, u_2, \dots, u_r, u_{r+1}, \dots, u_m] \tag{1.2}$$

Les vecteurs de colonne u_i , pour $i = 1, 2, \dots, m$, forment un ensemble orthonormé :

$$u_i^T u_j = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases} \tag{1.3}$$

Et la matrice $V_{(n \times n)}$ est une matrice orthogonale

$$V = [v_1, v_2, \dots, v_r, v_{r+1}, \dots, v_n] \tag{1.4}$$

Les vecteurs de colonne v_i , pour $i = 1, 2, \dots, n$, forment un ensemble orthonormé :

$$v_i^T v_j = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases} \tag{1.5}$$

Ici, $S_{(m \times n)}$ est une matrice diagonale avec les valeurs singulières (SV) sur la diagonale.

La matrice S peut être montrée dans la suite

$$S = \begin{bmatrix} \sigma_1 & 0 \dots & 0 \\ \vdots & \sigma_2 \ddots & \vdots \\ 0 & 0 \dots & \sigma_n \end{bmatrix} \tag{1.6}$$

Pour $i = 1, 2, \dots, n$, les σ_i sont appelées valeurs singulières (SVs) de la matrice A .

On peut prouver que :

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0 \quad \text{et} \quad \sigma_{r+1} = \sigma_{r+2} = \dots = \sigma_n = 0 \tag{1.7}$$

Pour $i = 1, 2, \dots, n$, les σ_i sont appelées valeurs singulières (SVs) de la matrice A .

Les v_i et u_i sont appelés vecteurs singuliers droits et gauches de la matrice A . [18]

II-3-2 Propriétés du SVD :

- Les valeurs singulières $\sigma_1, \sigma_2, \dots, \sigma_n$ sont uniques, cependant, les matrices U et V ne sont pas uniques.

- Puisque $A^T A = V S^T S V^T$, donc V diagonalise $A^T A$, il s'ensuit que les v_j sont les vecteurs propres de $A^T A$.
- Puisque $AA^T = U S S^T U^T$, il en résulte que U diagonalise AA^T et que les u_j sont les vecteurs propres de AA^T .
- Le rang de la matrice A est égal au nombre de ses valeurs singulières non nulles.
- La norme L2 et la norme de Frobenius d'une matrice $A \in R^{m \times n}$ de rang r sont données respectivement par :

$$\|A\|_2 = \sigma_1, \quad (1.8)$$

et

$$\|A\|_F = (\sum_{i=1}^r \sigma_i^2)^{\frac{1}{2}}, \quad (1.9)$$

- Si A est de rang r , alors $V_{1,1}, V_{2,1}, \dots, V_{r,1}$ forment une base orthonormale pour l'espace $\text{Im}(A^T)$ et $U_{1,1}, U_{2,1}, \dots, U_{r,1}$ forment une base orthonormale pour l'espace $\text{Im}(A)$.
- Le rang de la matrice A est égal au nombre de ses valeurs singulières non nulles [27].

$$A = U S V^T = \sum_{i=1}^r (\sigma_i u_i v_i^T) \quad (1.10)$$

II-3-3 Exemple de SVD :

Soit la matrice positive A ,

$$A = \begin{bmatrix} 3 & 10 & 7 \\ 4 & 2 & 1 \end{bmatrix}$$

Nous voulons trouver la décomposition SVD de A ,

On a

$$A^T A = \begin{bmatrix} 25 & 38 & 25 \\ 38 & 104 & 72 \\ 25 & 72 & 50 \end{bmatrix},$$

D'après (1.1), on obtient :

$$A^T A = V \begin{bmatrix} 168.3242 & 0 & 0 \\ 0 & 10.6758 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

Où

$$V = \begin{bmatrix} -0.3024 & 0.9485 & 0.0944 \\ -0.7846 & -0.1915 & -0.5897 \\ -0.5412 & -0.2524 & 0.8021 \end{bmatrix} = [v_1 \ v_2 \ v_3]$$

D'où, les valeurs singulières non nulles de A sont : $\sigma_1 = \sqrt{168.3242} = 12.9740$

Et $\sigma_2 = \sqrt{10.6758} = 3.2674$

Maintenant, d'après (1.1), nous avons $U \Sigma = AV$, où

$$U\Sigma = [\sigma_1 u_1 \sigma_2 u_2 \ 0] = [12.9740 u_1 \ 3.2674 u_2 \ 0],$$

Et

$$A = \begin{bmatrix} -12.5420 & -0.8361 & 0 \\ -3.3201 & 3.1586 & 0 \end{bmatrix}$$

Donc

$$u_1 = \frac{1}{12.9740} \begin{bmatrix} -12.5420 \\ -3.3201 \end{bmatrix} = \begin{bmatrix} -0.9667 \\ -0.2559 \end{bmatrix}, \quad u_2 = \frac{1}{3.2674} \begin{bmatrix} -0.8361 \\ 3.1586 \end{bmatrix} = \begin{bmatrix} -0.2559 \\ -0.9667 \end{bmatrix}$$

Et

$$U = [u_1 u_2] = \begin{bmatrix} -0.9667 & -0.2559 \\ -0.2559 & -0.9667 \end{bmatrix}$$

On a aussi, d'après (1.8) et (1.9)

$$\|A\|_2 = \sigma_1 = 12.9740 \quad \text{et} \quad \|A\|_F = (\sigma_1^2 + \sigma_2^2)^{\frac{1}{2}} = \sqrt{179} = 13.3791$$

II-4.Méthodologie de SVD appliquée en traitement d'images :

II-4-1. Approche SVD pour la compression d'image :

La compression d'image traite le problème de réduction de la quantité de données nécessaires pour représenter une image numérique. La compression est atteinte par la suppression de trois données de base redondances:

- La redondance du codage, qui est présente quand elle n'est pas optimale ;
- La redondance interpixel qui résulte de la corrélation entre les pixels ;
- La redondance psychovisuelle, due à des données ignorées par la vision humaine [24].

La propriété de SVD dit « Lerang de la matrice A est égal au nombre de ses valeurs singulières non nulles ». Dans de nombreuses applications, les valeurs singulières d'une matrice diminuent rapidement avec un rang croissant. Cette propriété nous permet de réduire le bruit ou compresser les données de la matrice en éliminant les petites valeurs singulières ou les rangs plus élevés.

Quand une image est transformée en SVD, ce n'est pas compressé, mais les données prennent une forme dans laquelle la première valeur singulière a une grande quantité d'informations sur l'image. Avec cela, nous ne pouvons utiliser que quelques valeurs singulières pour représenter l'image avec de petites différences par rapport à l'originale.

Pour compresser une image par SVD, nous montrons les procédures de détail :

$$A = USV^T = \sum_{i=1}^n (\sigma_i u_i v_i^T)$$

C'est A peut être représentée par :

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \dots + \sigma_r u_r v_r^T \quad (1.10)$$

Lors de la compression d'image, la somme n'est pas effectuée jusqu'aux derniers (SV). Les (SV) avec des valeurs assez petites sont supprimées. (les (SV) sont ordonnés sur la diagonale.). La matrice de rang k est obtenue par troncation de ces sommes après les k premiers termes

$$A_k = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \dots + \sigma_k u_k v_k^T \quad (1.11)$$

Le stockage total de A_k sera : $k(m + n + 1)$

Le nombre entier k peut être choisi moins que n et l'image numérique correspondante reste très proche de l'image originale.

II-4-2. Mesures de compression d'image :

Pour mesurer les performances de la méthode de compression de l'image par SVD, on va calculer le facteur de compression de l'image en utilisant le taux de compression[6] :

$$C_R = m * n / (k(m + n + 1))$$

Mesure la qualité entre l'image d'origine A et l'image compressée A_k , la mesure de l'erreur quadratique moyenne (MSE) peut être calculée comme suit[15] :

$$MSE = \frac{1}{mn} \sum_{y=1}^m \sum_{x=1}^n (f_A(x, y) - f_{A_k}(x, y))^2$$

II-4-3. Approche SVD pour la reconnaissance d'images de visage :

Au cours des dernières décennies, la reconnaissance faciale a largement attiré l'attention des chercheurs dans les domaines de la vision par ordinateur, réseaux de neurones, reconnaissance de formes, apprentissage automatique, etc. L'application de la reconnaissance faciale comprend : le contrôle d'accès basé sur la reconnaissance faciale, l'interaction homme-machine, la sécurité de l'information, etc [28].

Plusieurs approches de reconnaissance faciale ont été proposées pour la reconnaissance faciale bidimensionnelle. Une grande partie du travail a été consacrée à la détection de caractéristiques individuelles telles que les yeux, le nez, la bouche et le contour de la tête, et à la définition d'un modèle de visage par la position, la taille et les relations entre ces caractéristiques[2],[30].

L'approche SVD traite un ensemble de visages connus comme des vecteurs dans un sous-espace, appelé «espace de visage», engendré par un ensemble de vecteur visage appelé «visage de base» [7]. Comme l'analyse en composantes principales (PCA)([14]), la reconnaissance est effectuée en projetant une nouvelle image sur l'espace, puis une comparaison est faite entre ses coordonnées (position) dans cet espace et les

cordonnées des visages connus. Cependant, l'approche SVD a de meilleures propriétés numériques que la PCA.

Dans ce cas, nous avons défini la matrice A en tant qu'ensemble des visages connus. Supposons que chaque image de visage a $m \times n = M$ pixels et qu'elle est représentée par un $M \times 1$ vecteur colonne f_i . Un ensemble S de N images de visages d'individus connus forme une matrice $M \times N$:

$$S = [f_1, f_2, f_3, \dots, f_N]$$

L'image moyenne \bar{f} de l'ensemble S , est donnée par :

$$\bar{f} = \frac{1}{N} \sum_{i=1}^N f_i \quad (1.12)$$

La soustraction de f des visages connus donne :

$$a_i = f_i - \bar{f}, \quad i = 1, 2, 3, \dots, N$$

Cela donne une autre matrice de taille $M \times N$:

$$A = [a_1, a_2, a_3, \dots, a_N] \quad (1.13)$$

Nous appliquons ensuite le concept de décomposition en valeurs singulières pour décomposer A en USV^T .

Puisque $\{u_1, u_2, u_3, \dots, u_N\}$ forme une base orthonormale pour $\text{Im}(A)$, et puisque les colonnes de A sont des images de visages, $\text{Im}(A)$ est appelé "sous espace de visages" dans l'espace vectoriel des images de $m \times n$ pixels, et chaque U_i , $i = 1, 2, \dots, r$ peut être appelé un vecteur de base visage.

Soit $([x_1, x_2, \dots, x_r]^T)$ les coordonnées d'une image visage quelconque f de taille $m \times n$. Alors, d'après le théorème de PCA [29], x est la projection scalaire de $f - \bar{f}$ sur la base U_1, U_2, \dots, U_r tel que :

$$X = [u_1, u_2, \dots, u_r]^T [f - \bar{f}] \quad (1.14)$$

Ce vecteur de coordonnées X est utilisé pour trouver laquelle des images visages connus décrit mieux le visage f . C'est-à-dire trouver un visage f_i , $i = 1, 2, 3, \dots, N$, qui minimise la distance:

$$\varepsilon_i = \|X - X_i\|_2 = [(X - X_i)^T (X - X_i)]^{\frac{1}{2}}$$

où X_i est le vecteur de coordonnées de f_i qui est la projection scalaire de $f_i - \bar{f}$ sous-espace vectoriel engendré par les U_i (Visage de base) :

$$X_i = [u_1, u_2, \dots, u_r]^T [f_i - \bar{f}] \quad (1.15)$$

Un visage f est classé comme face f_i lorsque le minimum ε_i est inférieur à un seuil prédéfini ε_0 . Sinon, le visage f est classé comme "visage inconnu".

Si f n'est pas un visage, sa distance au sous-espace de visage sera supérieure à 0. La projection vectorielle de $f - \bar{f}$ sur l'espace de visage est donnée par :

$$f_p = [u_1, u_2, \dots, u_r]^T X$$

Où X est donnée dans (1.14)

La distance de f à la surface faciale est la distance entre $f - \bar{f}$ sur le sous-espace engendré par le visage de base U_i , est donnée par :

$$\varepsilon_f = \|(f - \bar{f}) - f_p\|_2 = \left[(f - \bar{f} - f_p)^T (f - \bar{f} - f_p) \right]^{\frac{1}{2}} \quad (1.16)$$

Si ε_f est supérieur à un seuil prédéfini ε_1 , alors f n'est pas une image de visage.

II-4-3-1. Étapes à suivre pour effectuer la reconnaissance faciale avec SVD :

L'organigramme de la reconnaissance faciale avec SVD est présenté à la figure (II.2)
Les explications de chaque étape sont comme suit :

1. Obtenir un ensemble des visages S avec N image de visage de personnes connues.
2. Calcule le visage moyen \bar{f} de S par (1.12).
3. Calcul de la matrice A donnée par (1.13).
4. Calculer la SVD de A par (1.10)
5. Pour chaque visage connu, calculez les coordonnées de vecteur X_i à partir de (1.15).
Choisissez un seuil ε_1 qui définit la distance maximale autorisée à partir de l'espace.
Déterminez un seuil ε_0 définissant la distance maximale autorisée par rapport à tout visage connu de l'ensemble S .
6. Pour identifier une nouvelle image d'entrée f , calculez les coordonnées de son vecteur X à partir de (1.14), la projection vectorielle f_p et la distance ε_f par (1.1). Si $\varepsilon_f > \varepsilon_1$, l'image d'entrée n'est pas un visage.
7. Si $\varepsilon_f < \varepsilon_1$, calculez la distance ε_i à chaque visage connu. Si tout $\varepsilon_i > \varepsilon_0$, l'image d'entrée peut être classée comme visage inconnu.
Si $\varepsilon_f < \varepsilon_1$ et un certain $\varepsilon_i < \varepsilon_0$, classifiez l'image d'entrée en tant que visage connu associé au minimum ε_i , et cette image peut éventuellement être ajoutée à l'ensemble d'apprentissage original [7].

Les 5 étapes peuvent être répétées. Cela peut mettre à jour le système avec plus d'instances de visages connus.

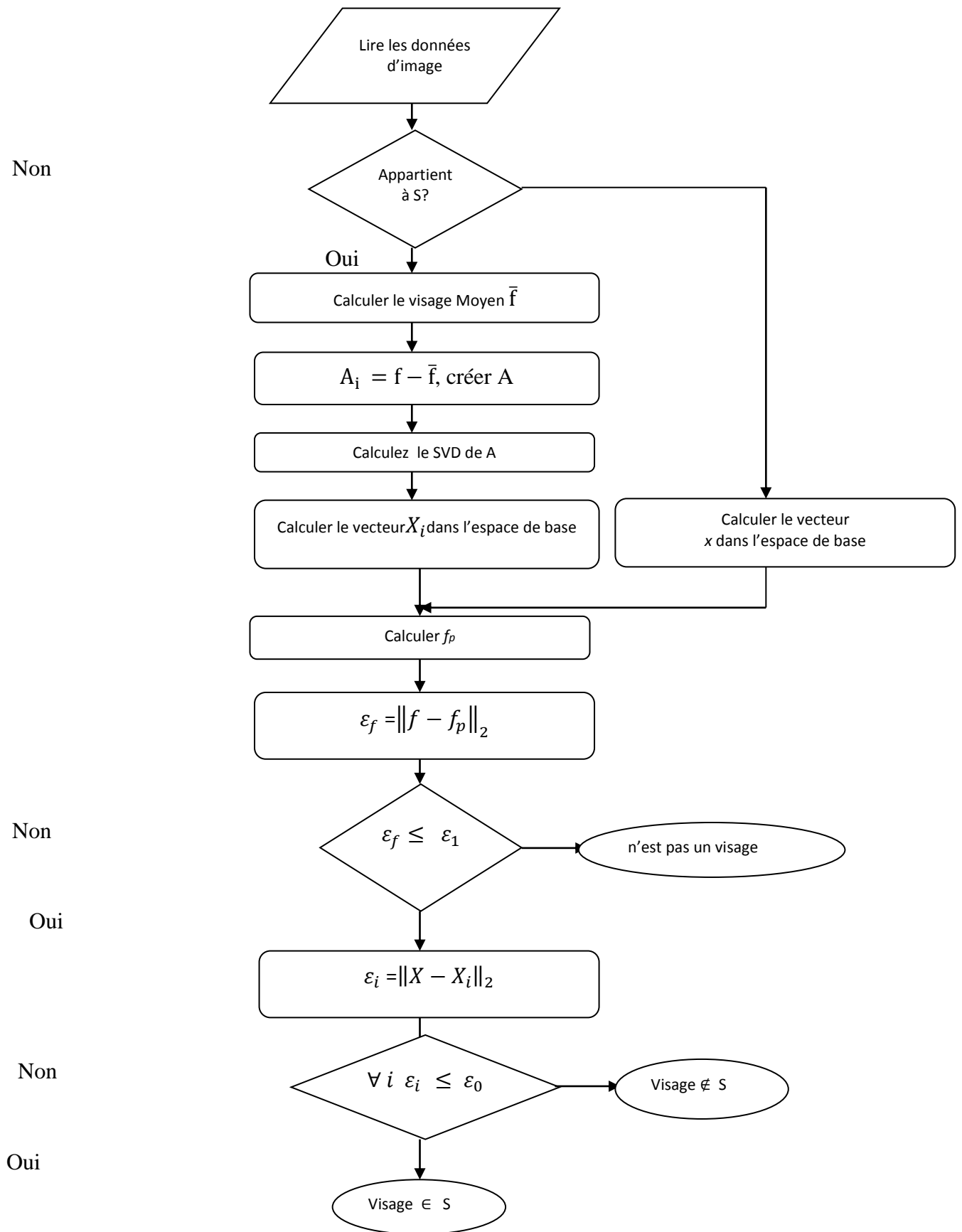


FIGURE-II-2: Organigramme de la reconnaissance faciale avec SVD

II-5.Conclusion :

Dans ce chapitre nous avons parlé de la décomposition en valeurs singulières et leurs principe et dans cette méthode il est possible d'obtenir des valeurs négatives. Et dans le domaine du traitement d'image sur l'image témoin permet de vérifier la validité de notre modèle. En effet, nous avons mis en évidence que si l'ensemble des valeurs singulières était conservé, l'écart avec l'image témoin était quasi-nul et que la qualité de l'image restituée se détériore au fur et à mesure que l'on prend en considération un nombre plus restreint de valeurs singulières. Il faut alors envisager l'utilisation d'une autre méthode.

CHAPITRE III:

Factorisation Matricielle Non Négative (NMF)

III-1.Introduction :

Le principe de la factorisation en SVD d'une matrice est largement utilisé en analyse en composantes principales (ACP) qui utilise la décomposition en valeurs singulières de la matrice X (SVD) pour construire des facteurs orthogonaux deux à deux. Paatero et Tapper (1994)[21] puis Lee et Seung (1999)[8] ont proposé une autre décomposition sans contrainte d'orthogonalité mais avec celle de non-négativité des matrices des facteurs afin d'en simplifier l'interprétation et sur la base d'une motivation "neuronale" : les neurones ne fonctionnent que de façon additive, pas soustractive.

Cette technique a depuis été largement utilisée dans de très nombreux domaines : imagerie, reconnaissance de formes, fouille de textes, systèmes de recommandations, génomique, avec pour objectif d'étudier la structure des très grandes matrices creuses.

La bibliographie s'est donc largement développée autour de ce thème en proposant différentes versions de l'algorithme avec différentes initialisations et contraintes, par exemple de parcimonie, dont certaines parallélisables, et tout un ensemble d'applications.

La NMF est donc une technique de réduction de dimension adaptée aux matrices creuses contenant des données positives, par exemple des occurrences ou dénombrements de mots, de pannes... La méthode est donc plus adaptée à certaines situations que la SVD mais cela a un prix.

La complexité algorithmique de la SVD est polynomiale de l'ordre du produit $n \times p$ des dimensions de la matrice. La complexité de la NMF est un problème non NP difficile; l'existence d'un algorithme de complexité polynomiale est inconnue. En revanche, il existe des approches itératives efficaces mais convergeant vers une solution locale sauf dans des cas très spécifiques (Donoho et Stodden, 2003)[9] ; contrairement à la SVD qui conduit à une solution unique (vecteurs propres et valeurs propres d'une matrice).

Lee et Seung (1999) illustre cette méthode sur la classification d'un corpus de 30991 articles de l'encyclopédie Grolier. Plutôt que de classer ces articles par thèmes choisis a priori, ils sont classés sur la base d'un vocabulaire de 15276 mots. Chaque article se décompose (coefficients positifs), en principe parcimonieusement, sur des "facteurs" ou thèmes, eux-mêmes définis chacun par un sous-ensemble petit, jugé "pertinent", de ces mots. En traitement d'images, un corpus se classifie à partir de facteurs ou motifs élémentaires d'images, en génomique par rapport à des "métagènes". L'approche non supervisée est ainsi susceptible de révéler des structures cachées ou des tendances sans a priori.

Par ailleurs, les facteurs de décomposition n'étant pas orthogonaux, des superpositions apparaissent : des mêmes mots participants à plusieurs thèmes, des gènes à plusieurs fonctions.

III-2.NMF : méthode et implémentations :

La description présente de la méthode de NMF ne se veut pas exhaustive ; elle est axée sur l'implémentation réalisée dans le package éponyme par Gaujoux et Seoighe (2010)[3] afin d'en préciser les options et critères mis en œuvre.

III-2-1. Principes :

Soit X une matrice ($n \times p$) ne contenant que des valeurs non négatives et sans ligne ou colonne ne comportant que des 0 ; r un entier choisi relativement petit devant n et p .

La factorisation non-négative de la matrice X est la recherche de deux matrices $W_{n \times r}$ et $H_{r \times p}$ ne contenant que des valeurs positives ou nulles et dont le produit approche X .

$$X \approx WH$$

Le choix du rang de factorisation $r \ll \min(n, p)$ assure une réduction drastique de dimension et donc des représentations parcimonieuses. Évidemment, la qualité d'approximation dépend de la parcimonie de la matrice initiale.

La factorisation est résolue par la recherche d'un optimum local du problème d'optimisation :

$$\min_{W, H \geq 0} [L(X, WH) + P(W, H)]$$

L est une fonction perte mesurant la qualité d'approximation et P une fonction de pénalisation optionnelle ; L est généralement soit un critère de moindres carrés (LS ou norme de Frobenius des matrices ou "norme trace"), soit la divergence de Kullback-Leibler (KL) ; P est une pénalisation optionnelle de régularisation utilisée pour forcer les propriétés recherchées des matrices W et H , par exemple, la parcimonie des matrices ou la régularité des solutions dans le cas de données spectrales.

$$LS: L(A, B) = \text{tr}((A - B)(A - B)') = \sum_{i,j} (a_{i,j} - b_{i,j})^2$$

$$KL: L(A, B) = KL(A||B) = \sum_{i,j} a_{i,j} \log\left(\frac{a_{i,j}}{b_{i,j}}\right) - a_{i,j} + b_{i,j}$$

N.B. Non seulement la solution est locale car la fonction objectif n'est pas convexe en W et H mais en plus la solution n'est pas unique. Toute matrice $D_{r \times r}$ non négative et inversible fournit des solutions équivalentes en termes d'ajustement :

$$X \approx W D D^{-1} H$$

Une fois la factorisation construite, il est ensuite facile d'utiliser ces matrices W et H pour construire des classifications (CAH, k-means), représentations (ACP, MDS), et prévisions à l'aide d'une des nombreuses méthodes d'apprentissage.

III-2-2. Méthodes de résolution :

Dans cette section, nous décrivons brièvement un certain nombre d'algorithmes existants pour le problème de factorisation matricielle non négative et les questions connexes telles que : l'initialisation des algorithmes, les conditions d'arrêt et la convergence.

Nous avons choisi des algorithmes typiques dans deux catégories principales : les mises à jour multiplicatives et les méthodes de calcul des moindres carrés alternatifs. Ceci a été établi en fonction de la popularité des algorithmes dans la pratique. Le deuxième algorithme est la méthode des moindres carrés alternatifs proposés par Paatero [21] pour la factorisation de la matrice positive. Moins d'attention a été attribuée à cette technique après l'introduction des mises à jour multiplicatives de Lee et Seung [11]. Le problème a ensuite été rebaptisé en factorisation matricielle non négative. La simplicité des mises à jour multiplicatives et l'interprétabilité des résultats ont permis d'étendre l'influence du NMF à presque tous les domaines de recherche : traitement d'image [20] [5] [26], traitement de texte [32] [16], musique, la transcription [22], l'analyse vidéo [12], la bio-informatique [33], la chimie [9]. La méthode standard du gradient projeté a été appliquée au NMF [3], qui présente certains avantages en termes de problèmes à grande taille. Récemment, une version révisée des moindres carrés alternés a été proposée dans [13], offrant une implémentation plus rapide en sacrifiant la propriété de convergence. D'autres tentatives tentent de changer la variable pour éliminer les contraintes de non-négativité.

III-3. Algorithme de Lee et Seung :

L'algorithme le plus populaire pour le problème NMF sont les règles multiplicatives (Algorithme 1) suggérées par Lee et Seung.

Pour formuler ces règles, nous choisissons de fixer l'un des facteurs (c'est-à-dire U ou V) et essayons de minimiser la fonction de coût par rapport à l'autre facteur. Le développement suivant montre comment formuler la règle de mise à jour pour V . Nous supposons d'abord que U et V sont positifs et nous y reviendrons plus tard.

Puisque la fonction de coût peut être découplée comme suit :

$$\frac{1}{2} \|A - UV^T\|_F^2 = \frac{1}{2} \sum_{i=1}^n \|A_{:,i} - U(V_{:,i})^T\|_2^2$$

On peut le minimiser par rapport à chacune lignes de V séparément. Cela aboutit à la résolution d'une séquence de problèmes quadratiques comme suit :

$$\min_{v \geq 0} F(v) \text{ où } F(v) = \frac{1}{2} \|a - Uv\|_2^2 \quad (3.1)$$

Considérons une approximation actuelle $\bar{v} > 0$ de la solution et formulons le problème suivant :

$$\min_{v \geq 0} \bar{F}(v) = \min_{v \geq 0} \frac{1}{2} [\|a - Uv\|_2^2 + (v - \bar{v})^T H_{\bar{v}} (v - \bar{v})]$$

Où $H_{\bar{v}} = D_x - U^T U$ avec $x = \frac{U^T U v}{[v]}$. Parce que nous pouvons prouver la semi-définitivité positive de $H_{\bar{v}}$, nous avons $\bar{F}(v) \leq F(v)$ pour tout v et surtout $\bar{F}(\bar{v}) = F(\bar{v})$. De plus, la fonction est également convexe. Nous fixons la dérivée de $\bar{F}(v)$ à zéro, c'est-à-dire :

$$\nabla_v \bar{F} = U^T U v - U^T a + H_{\bar{v}} (v - \bar{v}) = 0$$

pour obtenir le minimum v^* :

$$(U^T U + H_{\bar{v}})v^* = U^T a - H_{\bar{v}} \bar{v}$$

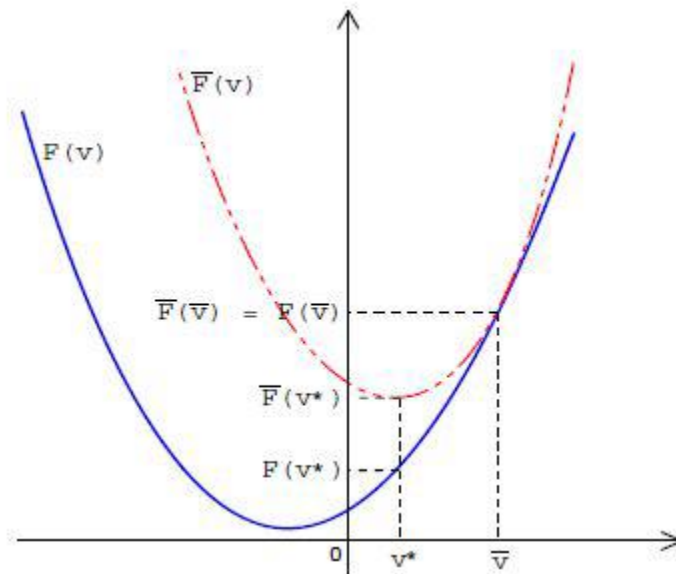


FIGURE III.1 – Mise à jour multiplicative non croissante

Puisque $U^T U + H_{\bar{v}} = D_{U^T U \bar{v}} D_{\bar{v}}^{-1}$ et $H_{\bar{v}} \bar{v} = 0$, nous concluons :

$$v^* = \bar{v} \circ \frac{[U^T a]}{[U^T U \bar{v}]} \quad (3.2)$$

Puisque v^* est le minimum global de $\bar{F}(v)$, nous avons $\bar{F}(v^*) \geq \bar{F}(\bar{v})$. De plus, $\bar{F}(v)$ est construit pour satisfaire $\bar{F}(v) \leq F(v)$ pour tout v . Cela implique que :

$$F(v^*) \geq \bar{F}(v^*) \leq \bar{F}(\bar{v}) = F(\bar{v}),$$

Ou nous avons une descente sur la fonction de coût. Ceci peut être résumé à la (FIGURE III-1) La fonction \bar{F} est généralement appelée fonction auxiliaire.

La résolution pour toutes les lignes de V donne la règle de mise à jour souhaitée pour V . La règle de mise à jour pour U peut être dérivée de la même manière. Et ces mises à jour sont les deux étapes alternées de l'algorithme 1.

Le terme supplémentaire dans (3.7) peut également être considéré comme une fonction de pénalité pour : empêcher l'annulation de la solution. De plus, la matrice $D_x - U^T U$ avec $x = \frac{[U^T U \bar{v}]}{[\bar{v}]}$ peut être vue comme une approximation de la matrice hessienne $U^T U$.

Algorithm 1 : Multiplicative Rules

1: Initialiser U^0, V^0 et $k = 0$

2: **repeat**

3: $U^{k+1} = U^k \circ \frac{[AV^k]}{[U^k(V^k)^T V^k]}$

4: $V^{k+1} = V^k \circ \frac{[A^t U^{k+1}]}{[V^k (U^{k+1})^T U^{k+1}]}$

5: $k = k + 1$

6: **until** Condition d'arrêt

III-4. Méthode des moindres carrés en alternance (ALS) :

Le premier algorithme proposé pour résoudre la factorisation matricielle non négative était la méthode des moindres carrés alternatifs [21]. Il est connu que, en fixant U ou V , le problème devient un problème des moindres carrés avec une contrainte de non-négativité.

Puisque les problèmes de moindres carrés de l'algorithme 2 peuvent être parfaitement découplés en problèmes plus petits correspondant aux colonnes ou aux lignes de A , nous pouvons appliquer directement des méthodes pour le problème des moindres carrés non négatifs à chacun des petits problèmes. Les méthodes pouvant être appliquées sont [4],[31], [25], etc.

Algorithm2 : Alternance de moindre carré (ALS)

Initialiser U et V

repeat

2 : Résoudre :

$$\min_{V \geq 0} \frac{1}{2} \|A - UV^t\|_F^2 \quad (3.3)$$

4 : Résoudre :

$$\min_{U \geq 0} \frac{1}{2} \|A^t - VU^t\|_F^2 \quad (3.4)$$

Until Condition d'arrêt

Algorithm3 : Inexacte moindres carrés alternés (IALS)

Initialiser U et V

repeat

3 : Résoudre pour U dans l'équation : $UV^tV = AV$
 $U = [U]_+$

6 : Résoudre pour V dans l'équation : $VU^tU = A^tV$
 $V = [V]_+$

Until Condition d'arrêt

Mais même avec l'implémentation rapide de ces algorithmes, ils ne peuvent pas correspondre aux autres méthodes en termes de temps d'exécution. Une modification a été faite en remplaçant une solution exacte du problème des moindres carrés non négatifs par la projection de la solution du problème des moindres carrés non contrainte dans l'orthant non négatif [12] comme dans l'algorithme 3. Ceci accélère l'algorithme en sacrifiant la propriété de convergence. La (figure III.2) est un exemple typique de la convergence des moindres carrés en alternance et des moindres carrés en alternance inexactes. On peut voir que même si le premier effectue toujours une mise à jour en descente, le dernier ne le fait pas. La méthode exacte produit également de meilleures erreurs d'approximation. Mais avec le même nombre d'itérations, il passe beaucoup plus de temps que la version inexacte (3.435s contre 0.02s). Notez que le solveur du problème des moindres carrés non négatifs dans cet exemple est la fonction standard de Matlab, 'lsqnonneg'. Pour un solveur plus rapide tel que [25], il est rapporté que la méthode exacte est encore loin derrière en terme de temps d'exécution. En pratique, les moindres carrés alternés exacts sont rarement utilisés car ils sont très inefficaces. Et sa version inexacte ne converge généralement pas vers un point stationnaire. Il est suggéré d'utiliser la version inexacte comme phase d'initialisation d'un algorithme hybride [20].

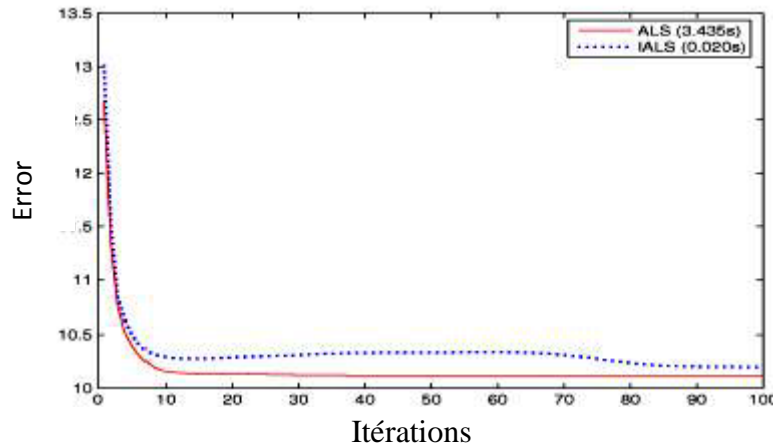


FIGURE III-2– Alternance des moindres carrés (ALS) vs Inexact Alternance des moindres carrés (IALS)

III-5- Approche NMF pour la compression d'image :

La compression d'image, un domaine important dans le domaine du traitement numérique de l'image, traite des techniques de réduction de stockage requis pour enregistrer une image ou la bande passante requise pour la transmission. L'objectif de la compression d'image vise à réduire le manque de pertinence et la redondance des données d'image, optimisant ainsi l'espace de stockage et augmenter le taux de transmission sur les pages Web. La non-négativité dans NMF garantit que les facteurs contiennent des parties cohérentes des données originales (images).

III-6. Reconnaissance faciale avec NMF :

III-6-1. Représentation et l'apprentissage :

La matrice de données F est construite de telle sorte que les images de visage d'apprentissage occupent les colonnes de la matrice F . L'ensemble des faces étant $\Gamma = \{f_1, f_2, \dots, f_m\}$, puis la matrice de données, $F = \{f_1, f_2, \dots, f_m\}$. Maintenant, l'apprentissage est fait en utilisant des équations (3.2) – (3.4) pour décomposer la matrice F en deux matrices, H et W . Soit les images de base être $W = [w_1, w_2, \dots, w_r]$ et les codages sont $H = [h_1, h_2, \dots, h_r]$. Chaque face f_i en F peut être approximativement reconstruite en combinant linéairement les images de base et les coefficients de codage correspondants $h_i = (h_{1i}, h_{2i}, \dots, h_{ri})^T$ comme indiqué à la figure 1. Par conséquent, une face peut être modélisée en termes de superposition linéaire de fonctions de base avec des codages comme suit:

$$f_i = Wh_i \quad (3.5)$$

Pour chaque face f_i de l'ensemble d'apprentissage et de l'ensemble de teste, nous calculons les coefficients de codage correspondants. Les images de base en W sont générées à partir de l'ensemble des faces d'apprentissage Γ^{train} . Les encodages, h_i de chaque visage d'apprentissage f_i est donné par :

$$h_i = W^{-1}f_i$$

Une fois formé, l'ensemble des images de visage, $\{f_1, f_2, \dots, f_m\}$ est représenté par un ensemble de codages $\{h_1, h_2, \dots, h_m\}$ de dimension réduite, r .

III-6-2. Le test :

Soit une image de face moyenne f , nous pouvons trouver un codage représentatif de f comme suit:

$$h = W^{-1}f$$

La (FIGURE III-3) illustre un codage d'un visage lorsque le rang est égal à 64. Une métrique de distance est utilisée pour calculer la similarité entre les codages d'une image d'apprentissage $h_i \in \Gamma^{train}$ et d'une image test $h \in \Gamma^{test}$. Le cosinus de l'angle entre les deux vecteurs de données est pris comme mesure de similarité:

$$S_i = \frac{h \cdot h_i}{\|h\| \cdot \|h_i\|} \quad (3.6)$$

La mesure de similarité S_i détermine le score de correspondance entre les codages h et h_i correspondant aux 2 faces f et f_i . Le codage de correspondance optimal d'une image peut être donné par h_{i^*} où :

$$i^* = \arg \max_i S_i$$

et $S_i > h_{thresh}$ indiquant que le visage f_{i^*} est identifié comme la correspondance la plus proche pour le visage f . Par conséquent, la meilleure image formée qui convient pour une image de test donnée est celle qui maximise S_i , à condition que le score soit supérieur à un seuil, h_{thresh} . S'il n'y a pas de valeur h_{i^*} pour laquelle le score est supérieur au seuil, l'image est rejetée. Le h_{thresh} est déterminé empiriquement comme le point auquel le taux d'acceptation fausse (classification erronée) et le taux de rejet est égal - le même taux d'erreur.



Figure III.3. L'image la plus à gauche est l'image originale du visage et l'image reconstruite à l'aide des images de base (W) et le codage d'image correspondant (h_i) est affiché directement sur l'image d'origine.

II-6-3. Étapes à suivre pour effectuer la reconnaissance faciale avec NMF

L'organigramme de la reconnaissance faciale avec SVD est présenté à la figure (III.2). Les explications de chaque étape comme suit :

1. Obtenir un ensemble des visages $\Gamma = \{f_1, f_2, \dots, f_m\}$ avec N image de visage de personnes connues.
2. une face modélisée en termes de superposition linéaire de fonctions de base avec des codages donné par $f_i = Wh_i$ (3.5).
3. Calculer les coefficients de codage correspondants pour chaque face f_i de l'ensemble d'apprentissage et de l'ensemble de teste. Les encodages, h_i de chaque visage d'apprentissage f_i est donné par h_i , solution du système (3.5).
4. Soit une image de face moyenne f (image test), nous pouvons trouver un codage représentatif de f comme suit: $h = (W' * W)^{-1} W' f$
5. calculer la similarité entre les codages d'une image d'apprentissage $h_i \in \Gamma^{Train}$ et d'une image test $h \in \Gamma^{Test}$. Le cosinus de l'angle entre les deux vecteurs de données est pris comme mesure de similarité: $S_i = \frac{h \cdot h_i}{\|h\| \cdot \|h_i\|}$
6. on a $i^* = \arg \max_i S_i$
7. et $S_{i^*} > \epsilon$ indiquant que le visage f_{i^*} est identifié comme la correspondance la plus proche pour le visage f . Par conséquent, la meilleure image formée qui convient pour une image de test donnée est celle qui maximise S_i , à condition que le score soit supérieur à un seuil, ϵ . S'il n'y a pas de valeur h_{i^*} pour laquelle le score est supérieur au seuil, l'image est rejetée.

L'organigramme de la reconnaissance faciale avec NMF est présenté dans la figure ci-dessous.

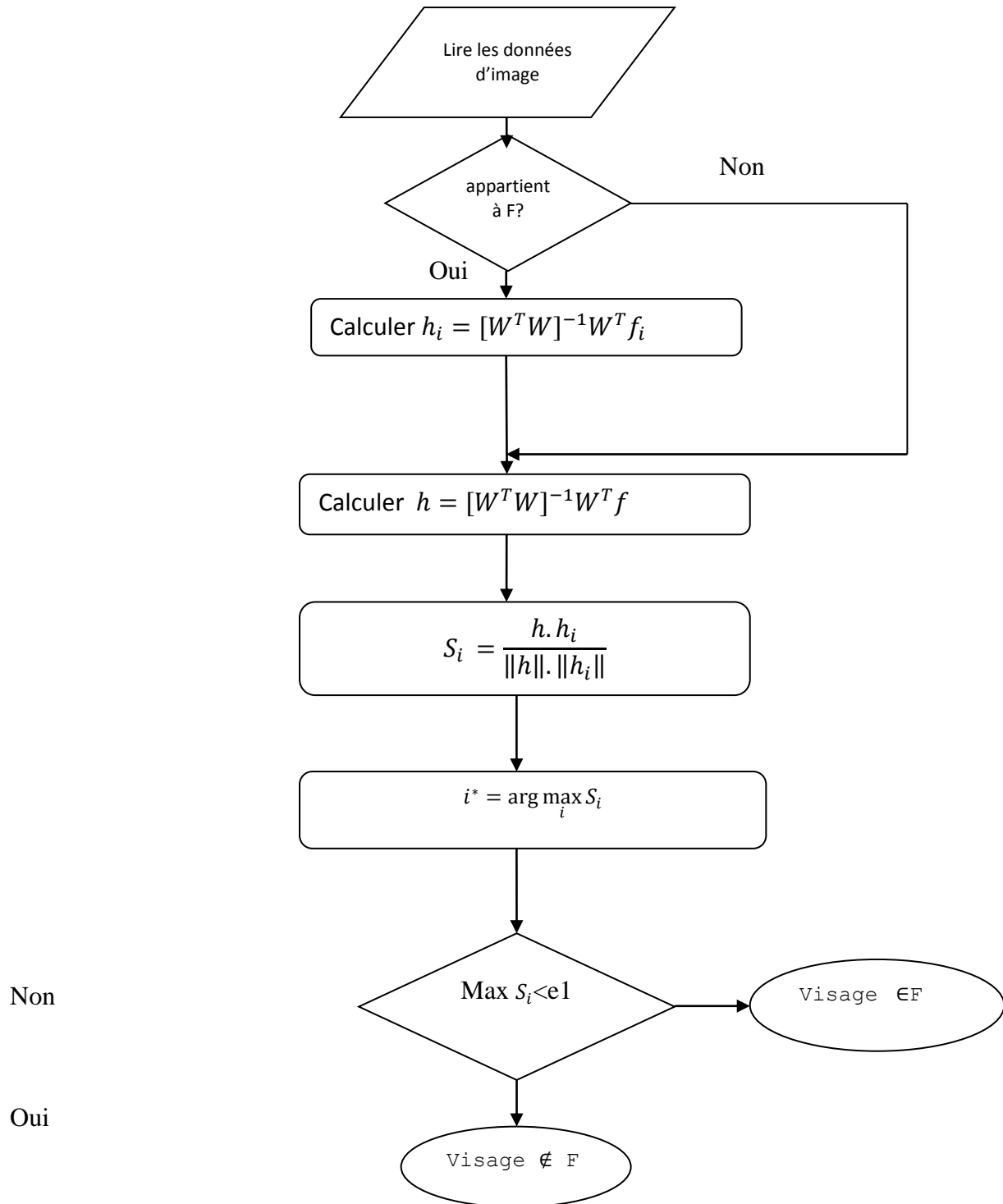


FIGURE.III.4: Organigramme de la reconnaissance faciale avec NMF

III-7-Conclusion :

La factorisation matricielle non négative contient des valeurs positives par contre que SVD contient des entrées négatives et donc du mal à interpréter. La factorisation matricielle non négative (NMF) présente de nombreux avantages par rapport à la norme Factorisation par SVD. Contrairement aux annulations dues à des entrées négatives dans les facteurs matriciels en mode SVD factorisations; la non négativité dans NMF garantit que les facteurs contiennent des parties cohérentes des données originales (images).

CHAPITRE IV:

Conception Et Implémentation

IV-1. Introduction :

Ce chapitre est consacré à la conception et la réalisation d'une application qui met en œuvre les techniques et algorithmes présentés pour la décompression d'image et la reconnaissance faciale. La première partie de ce chapitre est une application de la compression d'image par la méthode SVD et la méthode NMF. La deuxième partie est dédiée à l'implémentation des applications de la reconnaissance faciale avec les deux méthodes (SVD et NMF) ainsi qu'une comparaison des méthodes. Nous présenterons également les environnements matériel et logiciel qui ont été utilisés dans la réalisation de cette application.

IV-2. Bases de données :

Il y a plusieurs bases de données contenant des informations qui permettent l'évaluation des systèmes de reconnaissance de visages. Toutefois, ces bases de données sont généralement adaptées aux besoins de quelques algorithmes spécifiques de reconnaissance, chacune d'elle a été construite avec des conditions d'acquisition d'images de visages diverses (changements d'illumination, de pose, d'expressions faciales) ainsi que le nombre de sessions pour chaque individu.

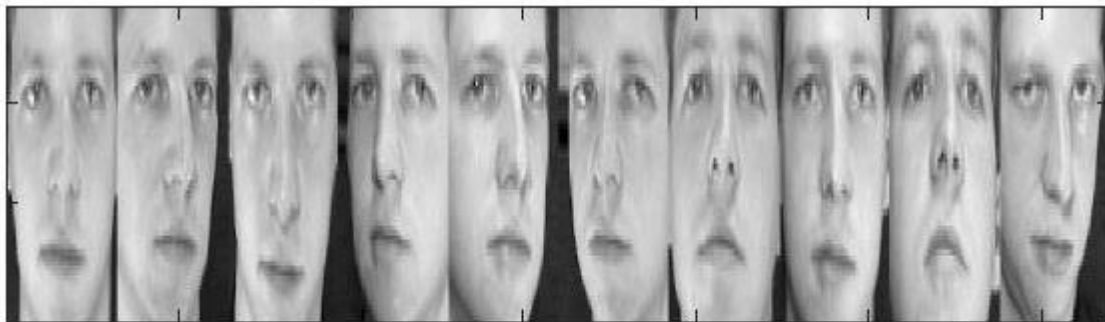
Les bases ORL et YALE ([10],[23]) ont été les plus utilisées et ont permis de comparer plusieurs méthodes de l'état de l'art. Cependant, d'autres bases de visages sont disponibles et destinées à des évaluations adaptées à certaines variabilités du visage telles que les bases (Color FERET, FRGC, CVL, AR et IV2).

IV-2-1. La Base de donnée « ORL » :

La base de données ORL a été élaborée entre avril 1992 et 1994 par le Laboratoire AT&T à l'université de Cambridge en Angleterre [10]. La base de données contient les visages de 40 personnes voir (FIGURE IV-1), chacune étant enregistrée dans 10 vues différentes comme le montre la (FIGURE IV-2). Les images sont à niveau de gris et de 112×92 pixels (92 pixels de colonne, 112 pixels de ligne). Pour certains sujets, les images ont été collectées à des dates différentes et avec des variations dans les conditions d'éclairage, les expressions faciales et par port des lunettes. Toutes les images ont été recueillies sur un fond sombre. Les formes de tête ont quelques différences de profondeur par rapport à la position frontale. Cependant, ces différences ne concernent que des personnes spécifiques et sont donc irrégulières.



Figure IV-1 : Base de donnée des images « ORL » (40 visages)



(a)



FIGURE IV-2 : La 1^{ère} et la 35^{ième} personnes de "ORL".

7IV-2-2. La base de données "YALE" :

Elle se compose de 165 images frontales en niveau de gris de 15 personnes voir (FIGUREIV-3), [23]avec 11 images pour chacune voir (FIGUREIV-4). On trouve trois angles d'éclairage différents : gauche, centre et droit, et il existe des images avec lunette et sans lunettes. La base offre des images incluant différentes expressions faciales : normale, triste, heureux, somnolant, étonnant, et clignotement de l'œil.



Figure IV-3 :Base de données des images « YALE » (15 visages)



(a)

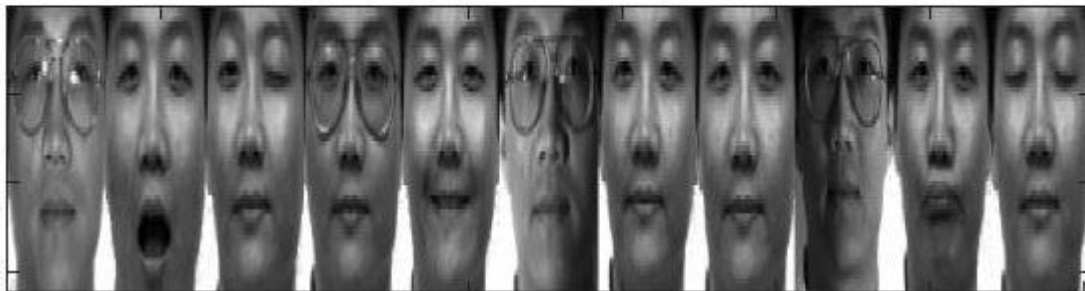


Figure IV-4 La 2è e et la 14ième personnes de la base « YALE ».

IV-3. Aspect matériel :

Notre projet a été développé sur un micro portable HP:

- * Processeur: Intel(R) Core (TM) i5-4210U CPU@ Capacité Mémoire(RAM) :8.00 Go
- * Vitesse d'horloge : 2.40 Ghz
- * Capacité disque dur : 700 Go
- * Système d'exploitation : Windows 7 édition intégrale 32 bits service pack 1.

IV-4. Outils de développement :

Pour la réalisation de notre système nous avons choisi le langage de programmation MATLAB (R2015b). MATLAB est un environnement de calcul scientifique et de visualisation de données. Sa facilité d'apprentissage et d'utilisation (due à une syntaxe très claire) en ont fait un standard adapté pour les divers problèmes l'ingénierie.

Parmi les raisons qui nous ont poussés à l'utiliser, on trouve :

- " Ses très nombreuses fonctions prédéfinies et prêtes à l'emploi.
- " Sa simplicité à l'implémentation et rapidité de calculs.
- " Sa fiabilité et sa robustesse.

MATLAB offre un certain nombre de fonctionnalités pour la documentation et le partage du travail. On peut intégrer le code MATLAB avec d'autres langages et applications, et distribuer les algorithmes et applications MATLAB.

IV-5.La Compression d'image :

La compression d'image, une discipline importante dans le domaine du traitement numérique de l'image, traite des techniques de réduction de stockage requis pour enregistrer une image ou la bande passante requise pour la transmission. L'objectif de la compression d'image vise à réduire le manque de pertinence et la redondance des données d'image, optimisant ainsi l'espace de stockage et augmenter le taux de transmission sur les pages Web.

IV-5-1.La Compression d'image par SVD :

Soit A la matrice associée à l'image originale, en utilisant la décomposition en valeurs singulières (SVD) (1.10) et (1.11), nous allons compresser l'image avec différentes valeurs singulières de K .

K : Le nombre de valeurs singulières



Image originale

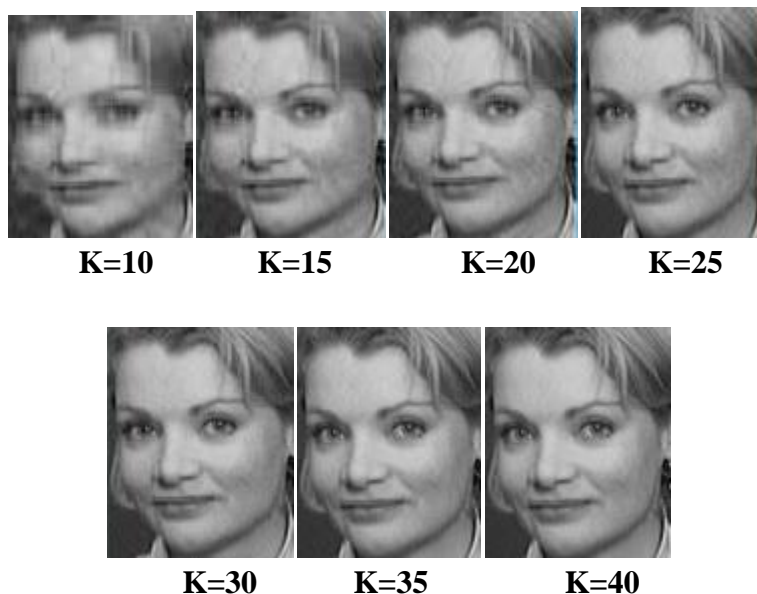


Figure IV-5. Exemples d'images utilisées pour tester la Compression d'image par SVD

La figure IV-5 montre les résultats obtenus en compressant l'image d'origine avec SVD avec différentes valeurs de K .

Nous avons trouvé quand ($k_1 \leq 25$), les images sont floues et avec l'augmentation des valeurs singulières on a une meilleure approche de l'image originale.

Cette étude nous montre que les valeurs singulières contiennent les informations de l'image. Cependant les valeurs singulières de faible amplitude contiennent peu d'information et leur influence n'est visible que si on en omet un grand nombre. Dans ce cas, la présence des valeurs singulières à forte amplitude permet la bonne compréhension de l'image.

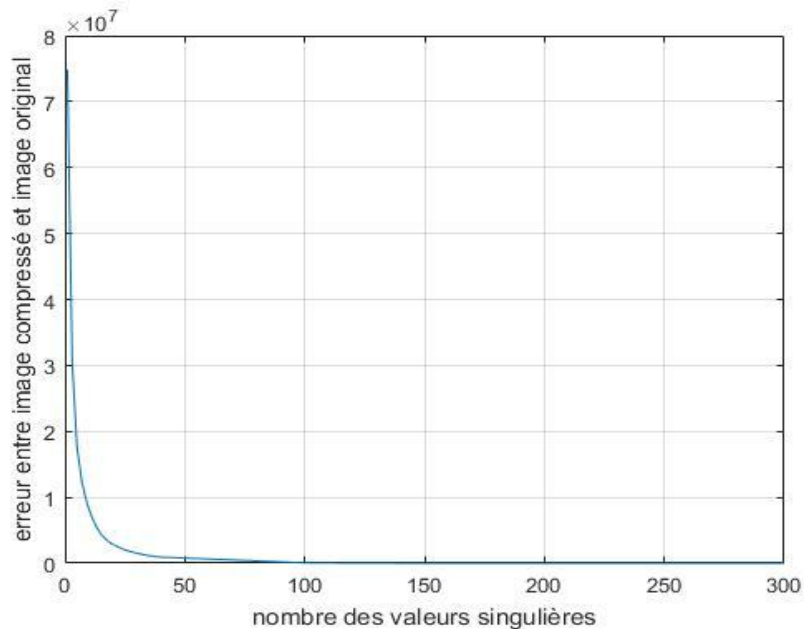


Figure IV.6 : erreur entre l'image originale et l'image compressée

Le Tableau IV-1 présente un résumé des résultats obtenus avec la mesure de l'espace de stockage, et la mesure d'erreur pour les images testées.

K	Espace de Stockage	CR	MSE
10	2050	5.01	96.84
15	3075	3.35	53.62
20	4100	2.51	31.95
25	5125	2.01	19.82
30	6150	1.67	12.74
35	7175	1.43	08.94
40	8200	1.25	05.69
Image original	10304	1	

Tableau IV-1. Résumé des résultats de la compression d'image par SVD

Avec les résultats de tableau IV-1, nous avons les observations suivantes:

- En utilisant une petite valeur singulière (plus petite K), le meilleur taux de compression est atteint.

- Cependant, la valeur la plus singulière est utilisée (plus grand K), mesure de qualité MSE est plus petit (meilleure qualité d'image), et les images reconstruites sont plus égales à l'original, mais en utilisant plus de stockage espace.
- Pour l'image testée, l'acceptable la qualité d'image est d'environ $k = 25$, et le taux de compression est $CR = 2.01$.
- L'image se rapproche de l'image d'origine lorsque $k = 40$. À ce stade, $CR = 1,26$ et $MSE = 9,07$.

Les résultats obtenus par la méthode de décomposition en valeurs singulières sont plus ou moins satisfaisants. Certes, il est toujours possible de récupérer une image assez proche de l'originale, mais pour cela il faut jouer sur le nombre de valeurs singulières conservées ; si on prend en considération un nombre plus restreint de valeurs singulières il faut alors envisager l'utilisation d'une autre méthode.

Le traitement sur l'image témoin permet de vérifier la validité de notre modèle. En effet, nous avons mis en évidence que si l'ensemble des valeurs singulières était conservé, l'écart avec l'image témoin était quasi-nul et que la qualité de l'image restituée se détériore au fur et à mesure que l'on prend en considération un nombre plus restreint de valeurs singulières. Il faut alors envisager l'utilisation d'une autre méthode.

IV-5-2. La Compression d'image par NMF:

Soit A la matrice associée à l'image originale, en utilisant la factorisation matricielle non négative (NMF)(3.1) Nous allons compresser l'image avec différentes valeurs de rang K .



Image originale

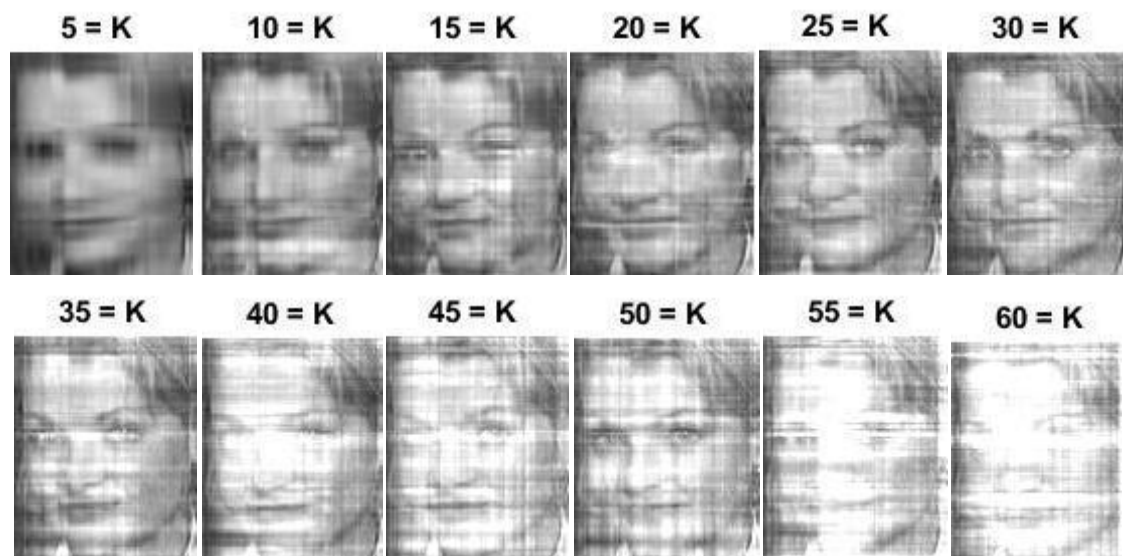


Figure IV-7 : Exemples d'images utilisées pour tester la Compression d'image par NMF

La figure IV-7 montre les résultats obtenus en compressant l'image d'origine avec NMF avec différentes valeurs de rang K . À travers les résultats obtenus, nous notons que lorsque la valeur de K est petite, l'image compressée est plus proche de l'image d'origine.

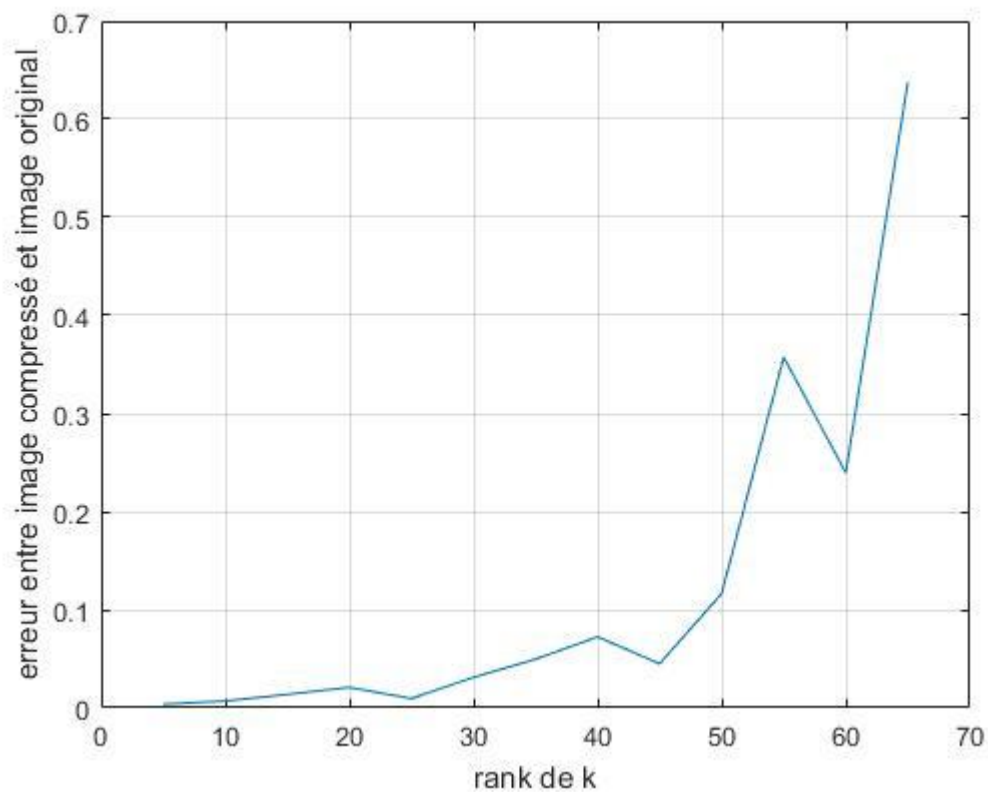


Figure IV.8 : Erreur entre l'image originale et l'image compressée

K	Espace de Stockage	CR	MSE
05	1025	10.053	0.0040
10	2050	5.0263	0.0032
15	3075	3.350	0.0087
20	4100	2.513	0.0182
25	5125	2.010	0.0167
30	6150	1.675	0.0417
35	7175	1.436	0.0365
40	8200	1.256	0.0771
45	9225	1.116	0.0373
50	10250	1.005	0.0920
55	11275	0.913	0.2609
60	12300	0.837	0.274

Tableau IV-2. Résumé du résultat pour la Compression d'image par NMF

Avec les résultats de tableau IV-1, nous avons les observations suivantes :

- En utilisant une valeur de K moins (plus petit K), le meilleur taux de compression est atteint.
- Cependant, la valeur de K est utilisée (plus petit K), mesure de qualité MSE est plus petit (meilleure qualité d'image), et les images reconstruites sont plus égales à l'original, mais en utilisant moins d'espace de stockage.
- Pour l'image testée, l'acceptable la qualité d'image est d'environ $k = 5$, et Le taux de compression est $CR = 10.053$

IV-6.La reconnaissance faciale :

La reconnaissance faciale qui est considérée comme l'un des problèmes les plus difficiles à résoudre dans le monde. Jusqu'à présent, plusieurs méthodes et approches sophistiquées ont été développées pour obtenir les meilleurs résultats de reconnaissance en utilisant des bases de données de visage spécifiques.

IV-6-1.La reconnaissance faciale par SVD:

Soit S ensemble d'apprentissage avec $N=40$ images (nombres des individus de taille $112*92=10304$ pixels.ifférentsConditions: toute inclinaison frontale et légère de la tête, différentes expressions faciales.



Figure IV-9 Ensemble des images d'apprentissage S

- Calcule le visage moyen \bar{f} de S



Figure IV-10. Image du visage moyen calculé d'après d'ensemble des images d'apprentissage S

-Création la matrice A et calculer le SVD(A).

- Pour chaque visage connu, calculez les coordonnées de vecteur X_i .

On a Choisis un seuil $\varepsilon_1 = 50$ qui définit la distance maximale autorisée à partir de l'espace. Déterminez un seuil $\varepsilon_0 = 15$ définissant la distance maximale autorisée par rapport à tout visage connu de l'ensemble S.

- on a identifier une nouvelle image d'entrée f (image test), calcule les coordonnées de son vecteur x , la projection vectorielle f_p et la distance ε_f .

Si $\varepsilon_f > \varepsilon_1$, l'image d'entrée n'est pas un visage.

Si $\varepsilon_f < \varepsilon_1$, calculez la distance ε_i à chaque visage connu.

Si tout $\varepsilon_i > \varepsilon_0$, l'image d'entrée peut être classée comme visage inconnu.

Si $\varepsilon_f < \varepsilon_1$ et un certain $\varepsilon_i < \varepsilon_0$, classifiez l'image d'entrée en tant que visage connu associé au minimum ε_i , et cette image peut éventuellement être ajoutée à l'ensemble d'apprentissage original.

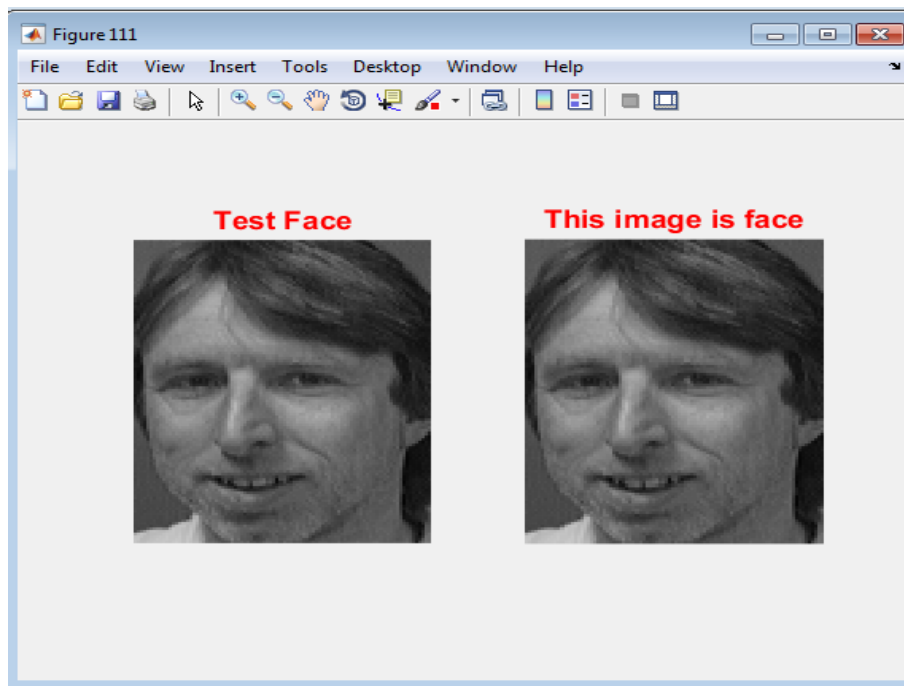


Figure IV-11. Image test et résultat de reconnaissance faciale par SVD

- This image is face#390
- Le temps écoulé est 15.657171 seconds.
- 390 c'est le numéro de l'image trouvé dans la bases d'apprentissage

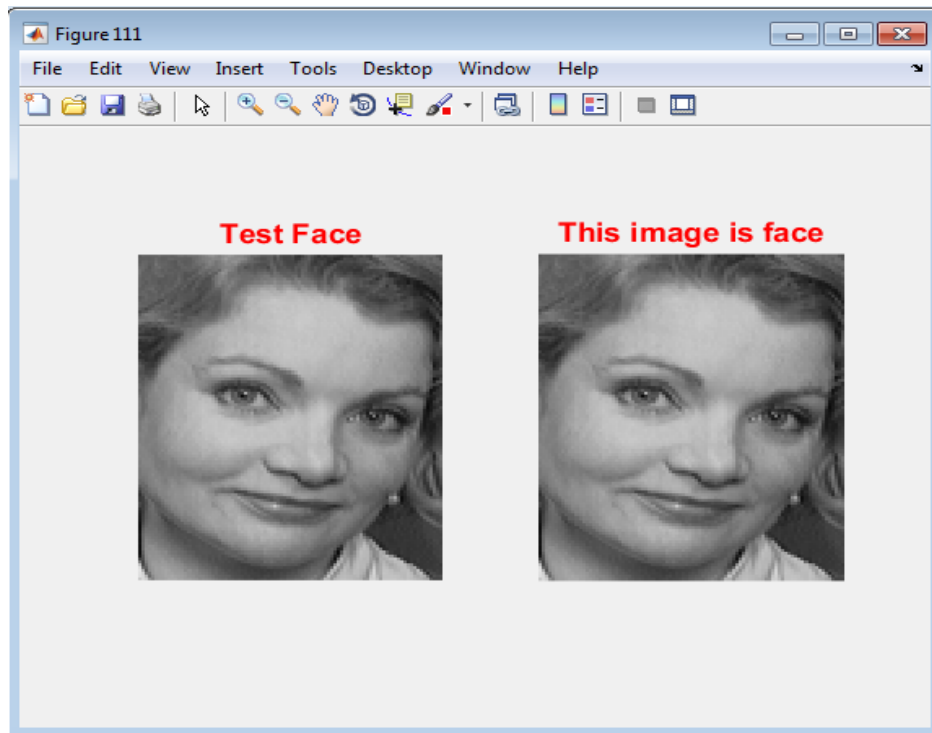


Figure IV-12. Image test et résultat de reconnaissance faciale par SVD

- This image is face #390
- Le temps écoulé est is 14.66385seconds.
- 347 c'est le numéro de l'image trouvé dans la bases d'apprentissage

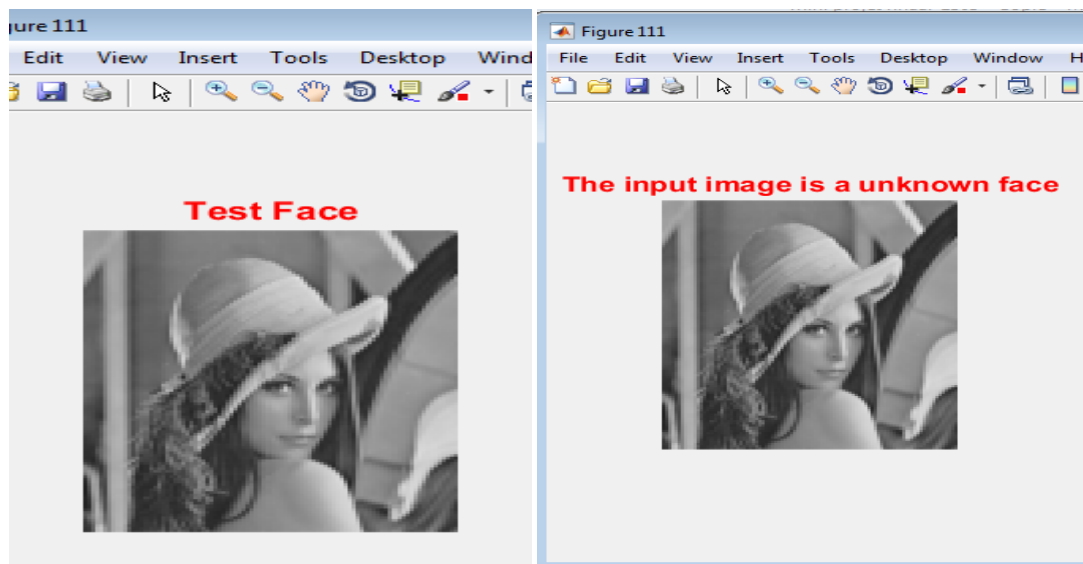


Figure IV-13. Image test et résultat de reconnaissance faciale par SVD

- The input image is a unknown face
- Le temps écoulé est 46.482068 seconds

IV-6-2. La reconnaissance faciale par NMF:

Soit $F = \{f_1, f_2, \dots, f_m\}$ ensemble d'apprentissage avec $N=40$ images (nombres des individus de taille $112 \times 92 = 10304$ pixels). Conditions: toute inclinaison frontale et légère de la tête, différentes expressions faciales.



FIGURE-IV-14. Ensemble des images d'apprentissage f_i

Pour chaque face f_i de l'ensemble d'apprentissage et de l'ensemble de teste, nous calculons les coefficients de codage correspondants. Les images de base en W sont générées à partir de l'ensemble des faces d'apprentissage Γ^{train} . Les encodages, h_i de chaque visage d'apprentissage f_i est donné par :

$$f_i = Wh_i \text{ alors } h_i = W^{-1}f_i = (W^T * W) \setminus (W^T * f_i)$$



Figure IV-15 L'image la plus à gauche est l'image originale du visage et l'image reconstruite à l'aide des images de base (W) et le codage d'image correspondant (h_i) est affiché directement sur l'image d'origine.

On calcule :

$$S_i = \frac{h \cdot h_i}{\|h\| \cdot \|h_i\|}$$

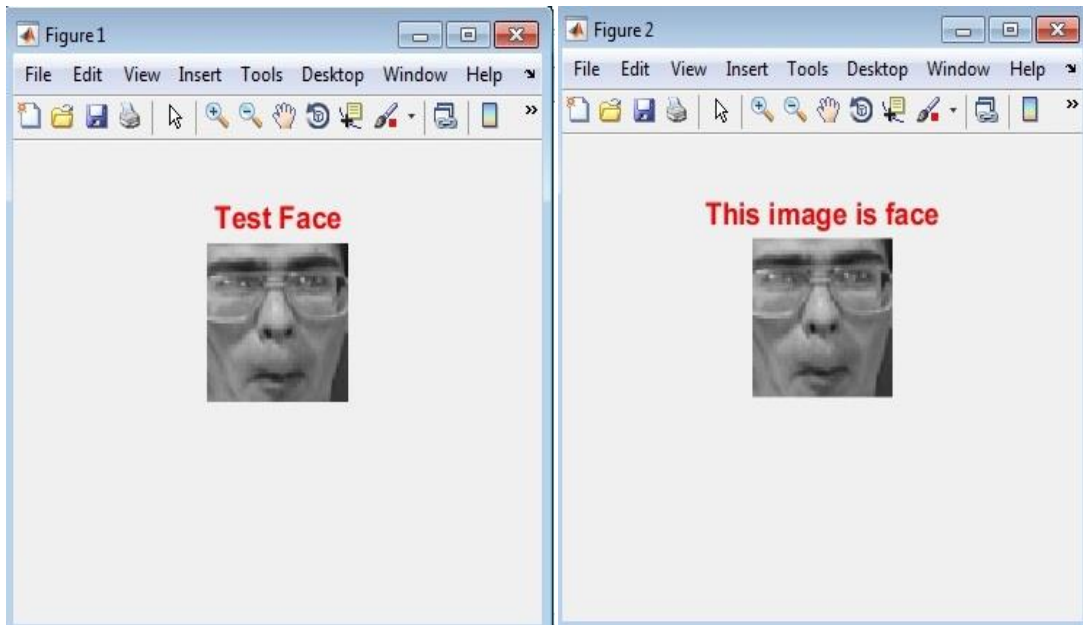


Figure IV-16. Image test et résultat de reconnaissance faciale par NMF

- This image is face#200
- Le temps écoulé est 2.295184seconds.
- 200 c'est le numéro de l'image trouvé dans la bases d'apprentissage

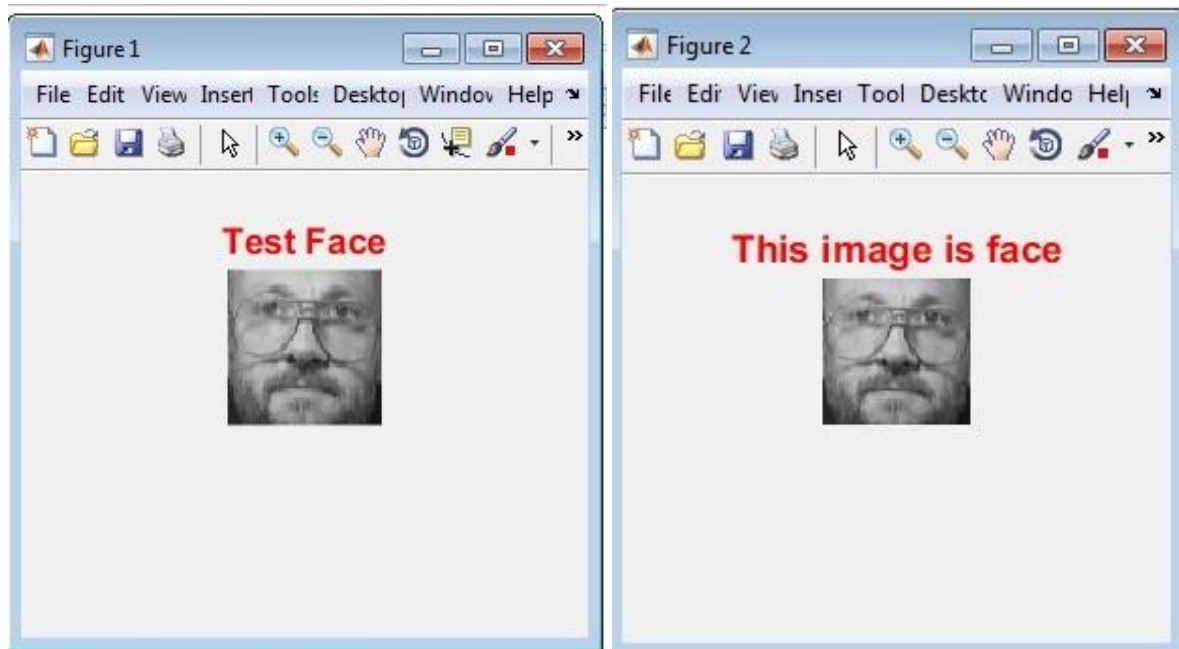


Figure IV-17. Image test et résultat de reconnaissance faciale par NMF

- This image is face#140
- Le temps écoulé est 2.166691seconds.
- 140 c'est le numéro de l'image trouvé dans la bases d'apprentissage



Figure IV-18. Image test et résultat de reconnaissance faciale par NMF

- This image is not a face in the dataset
- Le temps écoulé est 1.822869 seconds.

IV-7-Conclusion :

Dans ce chapitre nous avons évalué les applications de compression d'image et la reconnaissance faciale et D'après les résultats obtenue de test du plusieurs images de la reconnaissance faciale par NMF, on constate que la méthode NMF est plus mieux que la méthode SVD et le temps de la reconnaissance dans NMF est plus petit que la SVD.

La factorisation matricielle non négative contient de valeurs positives par contre que SVD contient des entrées négatives eta donc du mal à interpréter. La factorisation matricielle non négative (NMF) présente de nombreux avantages par rapport à la normeFactorisation par SVD. Contrairement aux annulations dues à des entrées négatives dans les facteurs matriciels en mode SVDfactorisations; la non négativité dans NMF garantit que les facteurs contiennent des parties cohérentes des données originales (images).

CONCLUSION GÉNÉRALE

Conclusion générale :

L'objectif de ce projet est de concevoir et implémenter une application de la compression d'image et de reconnaissance faciale en utilisant les méthodes SVD et NMF. La compression d'image résout le problème de stockage excessif des images ainsi que le problème de transfert d'images à travers les réseaux. Des travaux de recherches récents ont mis en évidence l'efficacité des méthodes SVD et NMF. Le but de ce travail était donc d'expérimenter ces méthodes en utilisant des bases de données tests.

Au bout de ce projet, nous avons pu implémenter les techniques SVD et NMF pour la compression d'image et la reconnaissance faciale. Des expériences ont été menées sur les bases de données d'images ORL et YALE. Les résultats obtenus confirment l'intérêt des recherches récentes en ces deux méthodes, en particulier dans la reconnaissance faciale. Les tests menés ont été concluants mais nous pensons qu'il faut aller au-delà de ces tests simples afin de confirmer les performances des méthodes étudiées dans le cas de bases de données d'images de grandeur nature.

Ce projet a été une occasion d'apprentissage intense que ce soit sur le plan théorique que pratique. En effet, la compréhension des méthodes a nécessité une revue intensive des notions mathématiques sur lesquels sont fondées les deux techniques. Ensuite il fallait mettre en œuvre les algorithmes en prenant en considération leur caractère numérique. Il est clair que dans le cas d'algorithmes de calcul numérique il faut faire attention aux problèmes d'erreurs numériques.

Les expériences menées sont encourageantes mais il faut pousser l'expérimentation plus loin encore, en particulier sur des bases de données d'images plus importantes afin de juger les performances. Ceci concerne plus particulièrement la méthode NMF compte tenu de sa complexité. Sur un autre plan, on peut penser à l'application de ces techniques mathématiques sur d'autres domaines de reconnaissance de formes, et plus particulièrement en biométrie (empreinte digitale, iris). En effet, ces domaines sont en pleine expansion dans notre pays, et le développement d'applications apportera des solutions précieuses.

Références Bibliographiques :

- [1] **Abriham R. et al**, « A Variation on SVD Based Image Compression », Third Workshop on Computer Vision, Graphics, and Image Processing, 2006, 1-6.
- [2] **A. L. Yuille, D. S. Cohen, and P.W. Hallinan.** (June 1989), *Feature Extraction from Faces Using Deformable Templates*, **Proc. CVPR, San Diego.**
- [3] **C.-J. Lin.** (2007), *Projected gradient methods for non-negative matrix factorization*, Neural Computation, To appear.
- [4] **C.L. Lawson and R.J.Hanson.** (1974), *Solving least squares problems.*, Prentice-Hall Englewood Cliffs, NJ.
- [5] **D. Guillaumet, M. Bressan, and J. Vitrià.** (2001), *A weighted nonnegative matrix factorization for local representations*, **In IEEE Computer Society Conference on Computer Vision and Pattern Recognition**, 942 – 947.
- [6] **Gérard Blanchet Maurice Charbit.** (2006), *Digital Signal and Image Processing using MATLAB*, ISTE
- [7] **Guoliang Zeng.** (2006), *Face Recognition with Singular Value Decomposition*, **CISSEI, Proceeding.**
- [8] **G.Sadashivappa, and K.V.S. AnandaBabu,**“ Performance analysis of Image Coding Using Wavelets”, IJCSNS International Journal of Comp. Science and Network Security, Oct.2008.
- [9] **H.T. Gao, T.H. Li, K. Chen, W.G. Li, and X. Bi.** (2005), *Overlapping spectra resolution using non-negative matrix factorization.*, *Talanta*, **66(1)**, 65 – 73.
- [10] **Laboratoire.AT.** (1992), *La base données "ORL"*, L'université de Cambridge, <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>
- [11] **Lee.D.D.,and Seung. H.S.** (1999), *Learning the parts of objects by nonnegative matrix factorization.*, *Nature*, , 401 .
- [12] **M. Cooper and J. Foote.** (2002), *Summarizing video using non-negative similarity matrix factorization*, **In IEEE Multimedia Signal Processing Workshop**, 25 – 28.
- [13] **M.W. Berry, M. Browne, A.N. Langville, V.P. Pauca, and R.J. Plemmons.** (2007), *Algorithms and applications for approximate nonnegative matrix factorization.*, **Computational Statistics and Data Analysis**, **52(1)**, 155 – 173.
- [14] **Matthew A. Turk, Pentland P. Alex.** (1991), *Face Recognition using Eigenface method*, **IEEE Conference on Computer Vision and Pattern Recognition**, pages 586 – 591.
- [15] **Mrak M., Grgic S. and Grgic M.** (September 2003), *Picture Quality Measures in image compression systems*, **IEEE EUROCON, Ljubljana, Eslovenia.**

BIBLIOGRAPHIE

- [16] **Nom1. P1., Nom2. P2., and Nom3. P3.** (2006), *Document clustering using nonnegative matrix factorization.*, **Information Processing and Management**, **42(2)**, 373 – 386.
- [17] **Cooper I., Lorenc C.**, « Image Compression Using Singular Value Décomposition », Collège of the Redwoods, 2006, 1-22.
- [18] **Guoliang Zeng**, “Face Recognition with Singular Value Decomposition”, CISSE
- [19] **H.Gene Golub et F.Charles Van Loan**, *Matrix Computations*, Third Edition, 1996.
- [20] **N.-D. Ho, V. Blondel, and P. Van Dooren.** (2007), *Weighted nonnegative matrix factorization and face feature extraction. Submitted to Image and Vision Computing.*
- [21] **Paatero.P and Tapper. U.** (1994), *Positive matrix factorization : a nonnegative factormodel with optimal utilization of error estimates of data values*, *Environmetrics*, **5(1)**, 111 – 126.
- [22] **P. Smaragdis and J.C. Brown.** (2003), *Non-negative matrix factorization for polyphonic music transcription*, **IEEE Workshop on Applications of Signal Processing to Audio and Acoustics**, 177 – 180.
- [23] **geoghiades.AT.Yale DOT** (1997), *La base donnée "YALE"*, <http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html>
- [24] **R. Albright, J. Cox, D. Duling, A.N. Langville, and C.D. Meyer.** (2006), *Algorithms, initializations, and convergence for the nonnegative matrix factorization.*, Preprint
- [25] **R. Bro and S. De Jong.** (1997), *A fast non-negativity constrained least squares algorithm.*, *Journal of Chemometrics*, **11(5)** :, pages 393 – 401.
- [26] **S.Z. Li, X.W. Hou, H.J. Zhang, and Q.S. Cheng.** (2001), *Learning spatially localized, parts-based representation*, **In Proceedings of IEEE Conf. Computer Vision and Pattern Recognition**, 1 – 6.
- [27] **Steve J. Leon.** (1996), *Linear Algebra with Applications*, Macmillan Publishing Company, New York.
- [28] **S.G. Kong, J. Heo, B.R. Abidi, J. Paik, and M. A. Abidi.** (2005), *Recent advances in visual and infrared face recognition a review.*, **Computer Vision and Image Understanding**, **97(1)** :, pages 103 – 135.
- [29] **Smith, Lindsay I.** (2002), *A Tutorial on Principal Component Analysis*, <http://csnet.otago.ac.nz/cosc453>
- [30] **T. Kanade.** (Nov.1973), *Picture Processing System by Computer Complex and Recognition of Human Faces*, Department of Information Science, **Kyoto University**.
- [31] **V. Franc, V. Hlaváč, and M. Navara.** (2005), *Sequential coordinate-wise algorithm for the non-negative least squares problem.*, **In CAIP, Computer Analysis of Images and Patterns**, volume **3691**, pages 407 – 414, September 2005.

BIBLIOGRAPHIE

- [32] **W. Xu, X. Liu, and Y. Gong.** (2003), *Document clustering based on nonnegative matrix factorization*, In **Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval**, pages 267–273. ACM Press New York, NY, USA,
- [33] **Y. Gao and G. Church** (2005), *Improving molecular cancer class discovery through sparse non-negative matrix factorization.*, *Bioinformatics*, **21(21)**, 3970– 3975.
- [34] **C.F. Van Loan** (2000), *The ubiquitous Kronecker product.*, *Journal of Computational and Applied Mathematics*, *123(1-2)*, pages 85 – 100
- [35] **A. Berman and R.J. Plemmons.** (1994), *Nonnegative matrices in the mathematical sciences.*, Siam.
- [36] **Topics in matrix analysis.** (1991), *Cambridge*, University Press. viii, 607 p.

Annexe

Cette partie présente les résultats de base et les concepts utilisés tout au long de cette thèse. Les résultats connus ne sont indiqués que sans preuve.

1-Théorie des matrices et algèbre linéaire :

A est la matrice réelle de taille $m \times n$ tel que m le nombre des lignes et n nombre des colonnes. Nous avons une matrice carrée lorsque le nombre de lignes est égal au nombre de colonnes. L'ensemble de $m \times n$ matrices réelles est noté $\mathbb{R}_{m \times n}$. Dans cette thèse, toutes les matrices sont réelles. Nous utilisons des lettres majuscules pour les matrices. La $i^{\text{ème}}$ ligne de la matrice A est notée $A_{i.}$. La $j^{\text{ème}}$ colonne de la matrice A est notée $A_{.j}$. L'élément à l'intersection de la ligne i et de la colonne j de la matrice A est noté A_{ij} ou $[A]_{ij}$.

Un vecteur de colonne est une matrice d'une seule colonne. De même, un vecteur ligne est une matrice d'une seule ligne. Sauf indication explicite, un vecteur est toujours un vecteur colonne. L'ensemble de tous les vecteurs de taille n est \mathbb{R}^n . Les vecteurs sont désignés par des lettres minuscules, sauf lorsqu'ils font partie d'une matrice, comme décrit dans le paragraphe précédent.

On dit qu'une matrice carrée A est symétrique si $A_{ij} = A_{ji}$, pour tout i, j . Une matrice diagonale est une matrice carrée comportant des éléments non nuls uniquement sur sa diagonale principale (c'est-à-dire, $A_{ij} = 0$ pour $i \neq j$). Nous utilisons D_x pour désigner une matrice diagonale avec le vecteur x sur sa diagonale principale (c.-à-d. $A_{ij} = x_i, i = 1, \dots, n$).

Voici quelques matrices spéciales :

- Matrices dont les éléments sont tous 1 : $\mathbb{1}_{1 \times n}, \mathbb{1}_{m \times 1}, \mathbb{1}_{m \times n}$

$$\mathbb{1}_{1 \times n} = (1, 1, \dots, 1) \quad \mathbb{1}_{m \times 1} = (1, 1, \dots, 1)^T \quad \mathbb{1}_{m \times n} = \mathbb{1}_{m \times 1} \mathbb{1}_{1 \times n}$$

- Vecteurs base

$$e_i = \left(0, 0, \dots, \overset{i^{\text{ème}} \text{ position}}{\hat{1}}, \dots, 0 \right)^T$$

- Matrices d'identité I_n : matrice diagonale où les éléments diagonaux sont égaux à 1.
- Matrices de permutation : matrices carrées comportant sur chaque ligne et chaque colonne un seul élément non nul, égal à 1.
- Matrices de sélection : toutes sous-matrices de matrices de permutation.

1.1 Manipulation matricielle :

Voici quelques opérations de base sur la matrice

- Matrice transposée $A^T : [A^T]_{ij} = A_{ji}$. A est une matrice symétrique $\Leftrightarrow A^T = A$
- Addition matricielle $C = A+B : C_{ij} = A_{ij} + B_{ij}$.
- Produit matricielle $C = A.B : C_{ij} = \sum_k A_{ik}B_{kj}$. Le point de produit est souvent supprimé.
- Vectorisation matricielle de $A \in \mathbb{R}^{m \times n}$

$$vec(A) = \begin{pmatrix} A_{:1} \\ \vdots \\ A_{:n} \end{pmatrix} \in \mathbb{R}^{m \times n}$$

- Produit de Kronecker de la matrice $A \in \mathbb{R}^{m \times n}$ et de la matrice B

$$A \otimes B = \begin{pmatrix} A_{11}B & \cdots & A_{1n}B \\ \vdots & \ddots & \vdots \\ A_{m1}B & \cdots & A_{mn}B \end{pmatrix}$$

Une relation importante entre le produit matrice et le produit Kronecker est la suivante [34] :

$$vec(AXB^T) = (B \otimes A)vec(X)$$

On écrit $A < B$ si $A_{ij} < B_{ij}$ pour tous i, j et ainsi pour $A \leq B$ et $A > B$ et $A > B$.

On utilise $A < \alpha, A \leq \alpha$ et $A \geq \alpha$, où $\alpha \in \mathbb{R}$, comme abréviations de $A < \alpha \mathbb{1}_{m \times n}, A > \alpha \mathbb{1}_{m \times n}, A \geq \alpha \mathbb{1}_{m \times n}, A \leq \alpha \mathbb{1}_{m \times n}$. La matrice absolue $|A|$ est définie comme suit : $[|A|]_{ij} = |A_{ij}|$ pour tout i, j .

Nous définissons le produit scalaire des deux vecteurs réels $x, y \in \mathbb{R}^n$ comme un réel :

$$\langle x, y \rangle = \sum_i x_i y_i = x^T y$$

Les vecteurs $x, y \in \mathbb{R}^n$ non nuls sont dits orthogonaux si leur produit scalaire est égal à zéro :

$$\langle x, y \rangle = 0$$

On considère une matrice $A \in \mathbb{R}^{m \times n}$ et sa vectorisation $vec(A) \in \mathbb{R}^{m \times n}$, nous pouvons également définir le produit intérieur de deux matrices réelles de même taille :

$$\langle A, B \rangle = vec(A)^T vec(B) = \sum_{ij} A_{ij} B_{ij} = tr(A^T B)$$

où la trace de A ($tr(A)$) est la somme de tous les éléments diagonaux de A . Ceci implique la relation utile suivante :

$$\langle I, ABC \rangle = \langle A^T, BC \rangle = \langle B^T A^T, C \rangle = \langle C^T B^T A^T, I \rangle = tr(ABC)$$

Une matrice carrée A est dite inversible s'il existe une matrice B telle que

BIBLIOGRAPHIE

$$AB = BA = I,$$

où B est appelé l'inverse de A et est noté $B = A^{-1}$.

La somme matricielle $C = A+B$ est définie par $C_{ij} = A_{ij} + B_{ij}$. Cette opération est dite élément par élément ou par entrée puisque chaque entrée de la matrice de résultat C dépend uniquement des entrées de A et B au même endroit. Ceci est contraire au produit matriciel habituel $C = AB$ où les relations ne sont plus locales. Un produit matriciel plus simple, élément par élément, est appelé produit Hadamard ou produit Schur $C = A \circ B$ où $C_{ij} = A_{ij}B_{ij}$ et A, B et C sont des matrices $m \times n$. Cela aide considérablement à simplifier les formules matricielles dans de nombreux cas. Voici quelques propriétés du produit

Hadamard [36] :

- $A \circ B = B \circ A$
- $A^T \circ B^T = (A \circ B)^T$
- $(a \circ b)(c \circ d)^T = (ac^T) \circ (bd^T) = (ad^T) \circ (bc^T)$

A.2 Sous-espaces de vecteurs

Un sous-espace linéaire E de \mathbb{R}^n est l'ensemble de toutes les combinaisons linéaires d'un ensemble de vecteurs $V = \{v_1, v_2, \dots, v_k\}$ de \mathbb{R}^n :

$$E = \left\{ \sum_{i=1}^k \alpha_i v_i \quad ; \alpha_i \in \mathbb{R} \right\}$$

E est également appelé l'espace engendré par V et V est appelé un ensemble générant de E . Dans un sous-espace E , il existe de nombreux ensembles générant. Parmi eux, un ensemble dont aucun vecteur ne peut être supprimé et sans modifier l'espace engendré est dit linéaire, indépendant et base de E . Le cardinal d'une base de E est fixé et est appelé dimension de E .

Par exemple :

$$E = \text{vect}(\{(1,2,1)^T, (1,0,0)^T\})$$

est un sous-espace de \mathbb{R}^3 et $\dim(E) = 2$ parce que $\{(1,2,1)^T, (1,0,0)^T\}$ est linéaire indépendant.

Suite à cela, le rang d'une matrice A de taille $m \times n$ peut également être défini comme la dimension du sous-espace engendré par les colonnes de A : est linéaire indépendant. Suite à cela, le rang d'une matrice A de taille $m \times n$ peut également être défini comme la dimension du sous-espace engendré par les colonnes de A :

$$\text{rg}(A) = \dim(\text{vect}(A_{:,1}, A_{:,2}, \dots, A_{:,n})) \leq \min(m, n).$$

Un sous-espace linéaire est fermé sous addition et multiplication scalaire, c'est-à-dire :

$$u, v \in E \Rightarrow u + v \in E$$

$$u \in E, \alpha \in \mathbb{R} \Rightarrow \alpha u \in E$$

1.2 Valeurs et Vecteurs Propres :

Les concepts centraux de l'analyse matricielle sont les valeurs propres et les vecteurs propres d'une matrice carrée. Ils fournissent des informations essentielles sur la matrice. Les concepts associés aux matrices rectangulaires sont ce qu'on appelle les valeurs singulières et les vecteurs. Ils jouent un rôle crucial dans les approximations de rang minimal qui conservent les caractéristiques dominantes de la matrice d'origine.

Définition Le scalaire $\lambda \in \mathbb{C}$ est une valeur propre de la matrice $A \in \mathbb{C}^{n \times n}$ s'il existe un vecteur non nul $x \in \mathbb{C}^n$ tel que $Ax = \lambda x$. Le vecteur x est appelé le vecteur propre associé à la valeur propre λ .

Une matrice A de taille $n \times n$ a exactement n valeurs propres. L'ensemble de toutes les valeurs propres est noté $\sigma(A)$. Le module maximum de $\sigma(A)$ est le rayon spectral de A et est noté $\rho(A)$:

$$\rho(A) = \max\{|\lambda|, \lambda \in \sigma(A)\}$$

Dans cette thèse, seules les valeurs propres et les vecteurs propres de certaines matrices symétriques sont examinés. Pour ces matrices, les résultats bien connus suivants peuvent être établis :

Théorème 1.1: (Théorème Spectral) Soit A une matrice réelle symétrique. Tous les vecteurs et les valeurs propres de A sont réels.

De plus, pour une matrice réelle symétrique A , si toutes les valeurs propres de A sont non négatives (respectivement non positives), A est dite semi-définie positive (respectivement semi-définie négative). Si toutes les valeurs propres sont positives (respectivement négatives), A est dite définie positive (respectivement définie négative).

La décomposition en valeurs singulières définie dans le théorème suivant est un outil très utile en analyse matricielle :

Théorème 4.2 Pour une matrice quelconque $A \in \mathbb{R}^{m \times n}$, il existe deux matrices orthogonales $U \in \mathbb{R}^{m \times m}$ et $V \in \mathbb{R}^{n \times n}$ tel que :

$$A = U \Sigma V^T$$

où les valeurs singulières σ_i sont des scalaires réels et décroissants

1.3 Les Normes :

Une norme est utilisée pour mesurer la magnitude d'un vecteur ou d'une matrice. Une norme sur \mathbb{R}^n (où $\mathbb{R}^{m \times n}$) on la note $\|\cdot\|$ qui vérifie les quatre conditions suivantes :

- 1- $\|x\| \geq 0, \forall x \in \mathbb{R}^n$ (où $\mathbb{R}^{m \times n}$)
- 2- $\|x\| = 0 \Leftrightarrow x = 0$;
- 3- $\|\alpha x\| = |\alpha| \|x\|, \forall x \in \mathbb{R}^n$ (où $\mathbb{R}^{m \times n}$) et $\forall \alpha \in \mathbb{R}$
- 4- $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in \mathbb{R}^n$ (où $\mathbb{R}^{m \times n}$)

La norme la plus courante est la norme Euclidienne ou la norme de Frobenius dérivée du produit scalaire :

$$\|x\|_F = \sqrt{\langle x, x \rangle}$$

1.4 Matrices non négatives :

Les matrices dont tous les éléments sont non négatifs sont appelées matrices non négatives. Nous utilisons \mathbb{R}_+^n et $\mathbb{R}_+^{m \times n}$ pour désigner l'ensemble des vecteurs non négatifs à n dimensions et l'ensemble des $m \times n$ matrices non négatives, respectivement. En effet, ces sous-ensembles sont des cônes polyédriques et sont généralement appelés orthant non-négatifs.

Une matrice non négative est appelée "ligne admissible" si elle n'a pas de ligne nul. De la même manière, une matrice non négative est appelée "colonne admissible" si elle n'a pas de colonne nul. Une matrice non négative est dite stochastique colonne (ligne) si toutes les sommes de colonne (ligne) sont égales à 1. Une matrice non négative est dite doublement stochastique si elle est stochastique en colonne et stochastique en ligne.

Le résultat le plus important pour les matrices non négatives est le suivant :

Théorème 1.2 (Perron-Frobenius, voir [35])

Soit A une matrice carrée non négative. Il existe une plus grande valeur propre de module de A qui est non négative et un vecteur propre non négatif qui lui correspond.

Ce vecteur est généralement appelé le vecteur de Perron de la matrice non négative. Pour une matrice non négative rectangulaire, des résultats similaires peuvent être établis pour la plus grande valeur singulière et ses vecteurs singuliers correspondants.

Étant donné un sous-ensemble $V \subset \mathbb{R}^{m \times n}$ et une matrice $A \in \mathbb{R}^{m \times n}$, l'élément le plus proche de V à A (par rapport à une distance) est appelé la projection de A sur V , notée $P_V(A)$. Lorsque le sous-ensemble V est l'orthant non négatif et que la distance considérée est la distance Euclidienne, la projection de A est notée $[A]_+$ et définie comme suit :

$$[[A]_+]_{ij} = \begin{cases} A_{ij} & \text{si } A_{ij} > 0 \\ 0 & \text{sinon} \end{cases} = \max(0, A_{ij}).$$