

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la Recherche scientifique



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Telecommunication
Option : Telecommunication

Réf:

**Mémoire de Fin d'Etudes
En vue de l'obtention du diplôme:**

MASTER

Thème

Etude de la réduction de Tucker pour SRV3D

Présenté par :
Sara GUERRI

Soutenu le : 02 Juin 2014

Devant le jury composé de :

Mr .BAARIR-Z/E

Dr. BELAHCENE BENATIA Mébarka

Mr ZITOUNI SAMIR

M.C.A

M.C.B

M.C.B

Président

Encadreur

Examineur

Année universitaire : 2013 / 2014

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la recherche scientifique



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Telecommunication
Option : Telecommunication

Mémoire de Fin d'Etudes
En vue de l'obtention du diplôme:

MASTER

Thème

Etude de la réduction de Tucker pour SRV3D

Présenté par :

Sara GUERRI

Avis favorable de l'encadreur :

Dr. BELAHCENE BENATIA Mébarka

signature

Avis favorable du Président du Jury

Mr .BAARIR-Z/E

Signature

Cachet et signature



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Telecommunication
Option : Telecommunication

Thème :

Etude de la réduction de Tucker pour SRV3D

Proposé par : Sara GUERRI

Dirigé par : Dr. BELAHCENE BENATIA Mébarka

RESUMES (Français et Arabe)

La réduction de dimensionnalité (DR) est une question clé pour améliorer l'efficacité des classificateurs pour hyper-spectrale images (HSI) et aussi dans la manipulation des bases de données biométrique où la taille énorme pose un grand problème. L'Analyse en Composantes Principales et indépendantes (PCA, ICA), la poursuite de projection (PP)...sont des approches DR qui ont été étudiées. Ces méthodes matrice d'algèbre sont appliquées sur les images vectorisées. Dans ce cas le réarrangement spatial est perdu. Pour prendre conjointement de l'information spatiale et spectrale ou bien différentes modalités de l'image la représentation sous forme de tenseur a été récemment utilisée. Offrant de multiples façons de décomposer les données orthogonalement, nous avons introduit des méthodes de DR basé sur des outils d'algèbre multilinéaire. Le DR est effectué à l'aide PCA, LDA ou autres, défini par une projection orthogonale sur un sous-espace de dimension inférieure des moyens spatiaux. Au cours des dernières années, le rapprochement au tenseur de faible rang a été établi comme un nouvel outil de calcul scientifique pour répondre au problème linéaire à grande échelle et aux problèmes d'algèbre multilinéaire, qui serait intraitable avec des techniques classiques. Ce travail tente de donner un aperçu de la littérature sur les développements actuels dans ce domaine, avec un l'accent sur les tenseurs et la réduction de Tucker 3 relatifs à la fonction. Le tenseur un terme qui se réfère simplement à un tableau multidimensionnel ou *d'ordre N*, et nous considérons comment il est spécialement structuré permettent un calcul et un stockage efficace. Les méthodes étudiées sont originales de part l'introduction d'une nouvelle représentation des données par un tenseur d'ordre 3. Cette représentation implique l'utilisation d'outils d'algèbre multilinéaire. De ce fait, les méthodes développées sont dites *multidimensionnelles* ou *multimodales*. Basées sur la décomposition tensorielle de TUCKER. Dans l'image du visage, la classification nécessite la réduction de la dimension spectrale et spatiale. Bien que les méthodes de réduction de la dimensionnalité utilisent l'algèbre linéaire, nous proposons une méthode d'algèbre multilinéaire pour atteindre conjointement le débruitage et la réduction de la dimensionnalité. Le format Tucker hiérarchique est un système de stockage efficace permettant de rapprocher et représenter les tenseurs d'ordre élevé avec la méthodologie et les algorithmes, qui offrent un moyen commode de travailler avec ce format.

Mots clé : Techniques de réduction ; Tenseurs ; Tucker 3 ; Algèbre Multilinéaire ; Reconnaissance de Visage.

ملخص

كل ما تشمله هذه المذكرة هو معرفة النظام الامني المتمثل في كيفية التعرف على الوجه حيث ان هذه الاخيرة من بين المهام الصعبة في النظام الامني وهي تحتاج الى طرق عديدة لتسهيل عملية التعرف على هوية شخص ما ومن بين هذه الطرق نجد (Tucker) وهي طريقة تتميز بالسرعة الفائقة للتعرف على وجوه الاشخاص وبدقة تامة حيث ان طريقة (Tucker) هي طريقة رياضية تقوم على حد وتقليص في المصفوفات الخاصة بالصورة الخاضعة للكشف عن هوية شخص ما وكذا تتميز هذه الطريقة بعدم ضياع معلومات الصورة وهذا ما يميزها عن باقي طرق الحد الاخرى وسنرى توسع في دراسة هذه الطريقة في بحثنا هذا . ويتم التطبيق في هذا المشروع باستخدام Matlab.

Remerciements

*J'exprime mes vifs remerciements à mon encadreur **Dr. BELAHCENE BENATIA** Mébarka, pour la grande bienveillance qu'elle m'a accordé et pour ses efficaces et précieuses aides, tout le long de l'achèvement de ce mémoire.*

Je remercie également tous les membres du jury d'avoir accepté de juger ce travail.

Et enfin je remercie l'ensemble des enseignants et collègues de notre promotion.

Dédicaces

✚ *A mon père, à ma mère*

✚ *A mes frères et mes sœurs*

✚ *Mon neveu et mes nièces*

✚ *Mon fiancé*

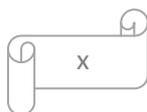
✚ *A toutes mes amies (surtout Meriem, Sarah ,
Ibtissam, Hakima, Naoual)*

✚ *A tout mes maîtres et professeurs : du primaire au
supérieur*

✚ *A toute ma famille*

*Introduction générale**Chapitre 1 Reconnaissance faciale 3D et Méthodes de Réduction*

Introduction	1
Partie A Reconnaissance Faciale 3D	1
1.1 La reconnaissance biométrique	1
1.2 Les caractéristiques biométriques	2
1.3 Mode de reconnaissance	4
1.4 Techniques 3D de reconnaissance de visages	4
1.5 Approches modèle	5
1.6 Approches 3D	5
• 1.6.1 Approches basées la surface 3D	5
• 1.6.2 Approches holistiques 3D	6
• 1.6.3 Approche géométrique ou locale 3D	7
1. <i>Approches basées sur les courbes faciales 3D</i>	7
2. <i>Approches basées sur les caractéristiques faciales</i>	8
1.7 Principales difficultés de la reconnaissance de visage	9
• 1.7.1 Changement d'illumination	9
• 1.7.2 Variation de pose	10
• 1.7.3 Expressions faciales	10
• 1.7.4 Présence ou absence des composants structurels	11
• 1.7.5 Occultations partielles	11
Partie B Méthodes d'analyse et de réduction de données	11
1.8.1 Analyse en composantes principales	11
• 1.8.1.1 Application de l'ACP à la reconnaissance de visage	12
• 1.8.1.2 EingenFace modulaire	13
1.8.2 Analyse discriminante linéaire	13
1.8.3 Analyse discriminante généralisée	14
1.8.4 Analyse discriminante linéaire directe	15
1.8.5 Analyse par noyau de Fisher	16
1.8.6 Analyse en Composantes Indépendantes	17
1.8.7 Méthode de Markov caché	19
1.8.8 Machine à vecteurs de support	20
1.8.9 La décomposition hiérarchique de Tucker (HTD)	22
Conclusion	24

*Chapitre 2 Etat de l'art sur la réduction de Tucker dans la SRV3D***Introduction** 26**2.1 Représentation des données hyperspectrales** 27

• 2.1.1 Espace spatial	28
• 2.1.2 Espace spectral	28
2.2 Définition et propriétés mathématiques d'un tenseur	29
• 2.2.1 Définition d'un tenseur	29
2.3 Décomposition de TUCKER3	31
• 2.3.1 Le modèle de TUCKER3	31
• 2.3.2 HOSVD	32
• 2.3.3 LRTA- (K_1, \dots, K_N)	33
• 2.3.4 Applications du modèle de TUCKER3	35
2.4 Autres décompositions tensorielles	35
• 2.4.1 PARAFAC/CANDECOMP, ParaCand	35
• 2.4.2 Les décompositions dérivées de PARAFAC/CANDECOMP	36
2.5 Etat de l'art d'application de tenseurs et réduction de Tucker	37
2.5.1 Calculs de MATLAB efficaces avec des tenseurs rares et pondérés	37
• 2.5.1.1 Tenseurs de Tucker	37
• 2.5.1.2 Propriétés du tenseur Tucker	38
• 2.5.1.3 Norme d'un tenseur de Tucker	38
• 2.5.1.4 Calcul $X_{(n)} X_{(n)}^T$ pour un tenseur Tucker	38
2.6.2 Tucker Décomposition non négatif	38
2.6.2.1 Contexte analyse multivoie	40
2.6.2.2 La différence entre deux modèles	41
2.6.2.3 Non négative décomposition Tucker	42
2.6.2.4 Les algorithmes de mise à jour multiplicatifs	42
2.6.2.5 Mise à jour des matrices de mode	43
2.6.2.6 Résultats	43
2.6.3 Contexte de l'imagerie hyperspectrale et problématiques abordées	46
2.6.3.1 Etat de l'art des méthodes de débruitage	46
1. Filtres de l'imagerie scalaire	47
2. Filtres hybrides	49
2.6.3.2 Etat de l'art des méthodes de réduction de dimension	50
1. Méthodes linéaires	51
2. Méthodes non-linéaires	51
3. Analyse discriminante	52
4. Réduction de dimension-Application pour l'amélioration de la classification	54
2.6.3.3 Application à l'imagerie hyperspectrale	55
2.6.3.4 Méthodes de réduction basées sur l'algèbre linéaire	56
1. Formulation tensorielle des transformations spectrales linéaires	58
2. Objectif : une analyse multimodale	59
2.6.3.5 Méthodes de réduction basées sur l'algèbre multilinéaire	59
1. Solution par approximation tensorielle, LRTA _{dr} - (K_1, K_2, P)	60

2. Etude d'un cas simple sur des données synthétiques	61
3. Evaluation sur les données réelles	62
2.7 Débruitage et réduction dimensionnelle à l'aide d'outils multilinéaires pour images hyper-spectrales	62
• 2.7.1 Outil de réduction de dimensionnalité $LRTA_{dr} - (K_1, K_2, D_3)$	63
2.8 Conclusion	64
Chapitre 3 Etude du SRV et réduction de Tucker3	
Introduction	68
Système de Reconnaissance de Visage	68
3.1 Détection de visage	68
• 3.1.1 Les difficultés liées à la détection de visage	69
• 3.1.2 Les méthodes de détection de visage	70
3.2 Prétraitement	71
• 3.2.1 Prétraitement basique	71
• 3.2.2 Exemple de prétraitement des bases d'apprentissage	72
3.3 Extraction des caractéristiques	73
• 3.3.1 Extraction des caractéristiques par patches	73
• 3.3.2 Extraction de la Forme du Visage	75
• 3.3.3 Extraction de la Texture du Visage	76
3.4 Réduction de donnée	77
• 3.4.1 Le format hiérarchique de Tucker	77
3.5 Classification	77
• 3.5.1 Classification à base de SVM	77
• 3.5.2 Evaluation des descripteurs et des fonctions noyaux	78
3.6 Conclusion	79
Chapitre 4 Conception de la réduction de Tucker et application au SRV3D	
Introduction	80
4.1 Environnement de la conception	81
• 4.1.1 Identification de l'ordinateur	81
• 4.1.2 Matlab	81
• 4.1.3 Quelles sont les particularités de MATLAB ?	81
4.2 Conception des méthodes multilinéaires	82
• 4.2.1 Comment calculer un tenseur ?	82
• 4.2.2 Conception de méthodes de réduction	84
• 4.2.3 Comment créer et afficher un tenseur ?	87

Sommaire

2014

- 4.2.4 Comment calculer le temps d'exécution de la méthode de réduction de Tucker ? 88

Conclusion	90
<i>Conclusion générale et perspectives</i>	91
<i>Bibliographie</i>	93
<i>Annexe</i>	97

Chapitre 1

Figure 1.1 Les principales caractéristiques biométriques a) forme de l'oreille, b) visage 2D, c) visage 3D, d) visage infrarouge, e) iris, f) rétine, g) empreinte de la main, h) thermo-gramme de la main, i) forme de la main, j) empreinte digitale, k) voix, l) signature et m) réseau veineux de la main.	3
Figure 1.2 Exemple de variation d'éclairage	10
Figure 1.3 Exemples de variation de poses	10
Figure 1.4 Exemples de variation d'expressions	11
Figure 1.5 Cinq premiers visages propres construits avec l'approche (a) ACP ; (b) LDA	14
Figure 1.6 Diagramme de l'algorithme de la LDA directe (DLDA)	16
Figure 1.7 Le modèle de synthèse d'image de l'architecture 1 d'ICA	17
Figure 1.8 La représentation d'ICA = (b_1, b_2, \dots, b_n)	18
Figure 1.9 Les deux architectures d'ICA : l'architecture 1 pour trouver des images de base statistiquement indépendantes (la séparation des images de visages produit des images composantes indépendantes), et l'architecture 2 pour trouver un code factoriel (la séparation des pixels produit un code factoriel)	19
Figure 1.10 Le modèle de synthèse d'image de l'architecture 2 d'ICA	19
Figure 1.11 La représentation factorielle d'ICA = (u_1, u_2, \dots, u_n)	19
Figure 1.12 Exemple de deux classes linéairement séparables. (L'hyperplan déterminé par le SVM, maximisant la marge, permet de séparer les deux classes de manière optimale)	21
Figure 1.13 Illustration de la HTD (τ) pour $d = 4$	23
Figure 1.14 Un arbre de dimension pour $d = 7$	24

Chapitre 2

Figure 2.1 Illustration du cube hyperspectral	27
Figure 2.2 Illustration de la représentation tensorielle des données hyperspectrales de HSI02_scene1b	27
Figure 2.3 Représentation spatiale : collection d'images quasi monochromatiques	28
Figure 2.4 Représentation spectrale : collection de spectres	29
Figure 2.5 Représentation matricielle des données hyperspectrales	29
Figure 2.6 Fibres d'un tenseur d'ordre 3	31
Figure 2.7 Tranches d'un tenseur d'ordre 3	37
Figure 2.8 Trace de l'erreur de Frobenius : NMF et NTD avec différents Méthodes d'initialisation.	44
Figure 2.9 Exemples d'images base. De haut en bas: (1) $\theta = 0$; (2) $\theta = 0,5$; (3) $\theta = 0,8$. Comme le θ augmente, les images de base deviennent plus clairsemées	44
Figure 2.10 De gauche à droite: (1) signal original ; (2) signal récupéré de la NTF ; (3) signal récupéré de la wihth de nsNTF $\theta = 0,8$	45

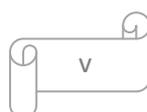
Figure 2.11 De haut en bas : des images originales (1) de visage ; (2) images contaminés par le bruit 'poivre & sel' ; (3) reconstruit des images par NMF ; (4) reconstruit des images par ATN	45
Figure 2.12 Images de base apprises par NMF (a) et (b) NTD, donné sur un ensemble de visages photographiés contaminés par poivre & bruit de sel	46
Figure 2.13 Illustration des 15 premières composantes indépendantes obtenues en utilisant la méthode de PP, avec réduction de dimension au préalable (≈ 30). Données utilisées HSI02_scene3	57
Figure 2.14 Schématisation de la réduction de dimension 3-modale d'un tenseur d'ordre 3 par transformation linéaire	58
Figure 2.15 Schématisation de la méthode de réduction multimodale proposée d'un tenseur d'ordre 3	58
Figure 2.16 a. Tenseur d'ordre 3 simulé $2 R^{50 \times 50 \times 150}$, b. diagramme de dispersion entre les bandes spectrales c. rapprochées et d. éloignées	61
Figure 2.17 Résultats de classification en fonction de la méthode de réduction de dimension spectrale utilisée pour plusieurs valeurs de P. Données utilisées : HSI02_scene3 (gauche) et HSI03_scene1 (droite)	62

Chapitre 3

Figure 3.1 Système simplifié – reconnaissance de visages (technique de Tucker)	68
Figure 3.2 Exemples de détection de visages dans des images	69
Figure 3.3 L'image avant (a) et après (b) découpage et décimation	71
Figure 3.4 Aperçu de l'approche	72
Figure 3.5 Etapes du prétraitement pour les images	72
Figure 3.6 Pyramide de patches multi-résolutions pour un pixel i . A chaque étape, différents niveaux d'informations sont mis en valeur.	74
Figure 3.7 Les exemples positifs sont extraits sur les 9 points les plus proches de la vérité-terrain. Les exemples négatifs sont ici représentés par des croix	74
Figure 3.8 Description de la forme d'un visage par les coordonnées 2D de 73 points caractéristiques	75
Figure 3.9 Représentation d'un visage sous la forme d'un maillage à facettes triangulaires	76
Figure 3.10 Scores 2AFC pour la détection des AU de la partie haute et basse du visage pour a) une fonction noyau intersection d'histogrammes et différents histogrammes et b) des histogrammes LGBP et différentes fonctions noyaux	79

Chapitre 4

Figure 4.1 Taux d'erreur pour différents échantillons d'apprentissage	87
Figure 4.2 (a) image originale, (b) histogramme d'image originale, (c) tenseur (affichage 2D)	88



Chapitre 2

Tableau 2.1 <i>Comparaison entre différents les décompositions tensorielles</i>	36
Tableau 2.2 <i>Notations impliquant tenseur algèbre</i>	41
Tableau 2.3 <i>Modèles pour NMF, nsNMF et NTF sont résumées dans le cadre de la décomposition de tenseur</i>	43
Tableau 2.4 <i>Précision de globalement du SAM en environnement bruyant. Résultats obtenus à partir de 30 conservé des composantes spectrales</i>	64

RESUME

La réduction de dimensionnalité (DR) est une question clé pour améliorer l'efficacité des classificateurs pour hyper-spectrale images (HSI) et aussi dans la manipulation des bases de données biométrique où la taille énorme pose un grand problème. L'Analyse en Composantes Principales et indépendantes (PCA, ICA), la poursuite de projection (PP)...sont des approches DR qui ont été étudiées. Ces méthodes matrice d'algèbre sont appliquées sur les images vectorisées. Dans ce cas le réarrangement spatial est perdu. Pour prendre conjointement de l'information spatiale et spectrale ou bien différentes modalités de l'image la représentation sous forme de tenseur a été récemment utilisée. Offrant de multiples façons de décomposer les données orthogonalement, nous avons introduit des méthodes de DR basé sur des outils d'algèbre multilinéaire. Le DR est effectué à l'aide PCA, LDA ou autres, défini par une projection orthogonale sur un sous-espace de dimension inférieure des moyens spatiaux. Au cours des dernières années, le rapprochement au tenseur de faible rang a été établi comme un nouvel outil de calcul scientifique pour répondre au problème linéaire à grande échelle et aux problèmes d'algèbre multilinéaire, qui serait intraitable avec des techniques classiques. Ce travail tente de donner un aperçu de la littérature sur les développements actuels dans ce domaine, avec un l'accent sur les tenseurs et la réduction de Tucker 3 relatifs à la fonction. Le tenseur un terme qui se réfère simplement à un tableau multidimensionnel ou *d'ordre N*, et nous considérons comment il est spécialement structuré permettent un calcul et un stockage efficace. Les méthodes étudiées sont originales de part l'introduction d'une nouvelle représentation des données par un tenseur d'ordre 3. Cette représentation implique l'utilisation d'outils d'algèbre multilinéaire. De ce fait, les méthodes développées sont dites *multidimensionnelles* ou *multimodales*. Basées sur la décomposition tensorielle de TUCKER. Dans l'image du visage, la classification nécessite la réduction de la dimension spectrale et spatiale. Bien que les méthodes de réduction de la dimensionnalité utilisent l'algèbre linéaire, nous proposons une méthode d'algèbre multilinéaire pour atteindre conjointement le débruitage et la réduction de la dimensionnalité. Le format Tucker hiérarchique est un système de stockage efficace permettant de rapprocher et représenter les tenseurs d'ordre élevé avec la méthodologie et les algorithmes, qui offrent un moyen commode de travailler avec ce format.

Mots clé : Techniques de réduction ; Tenseurs ; Tucker 3 ; Algèbre Multilinéaire ; Reconnaissance de Visage.

ملخص

كل ما تشمله هذه المذكرة هو معرفة النظام الامني المتمثل في كيفية التعرف على الوجه حيث ان هذه الاخيرة من بين المهام الصعبة في النظام الامني وهي تحتاج الى طرق عديدة لتسهيل عملية التعرف على هوية شخص ما ومن بين هذه الطرق نجد (Tucker) وهي طريقة تتميز بالسرعة الفائقة للتعرف على وجوه الاشخاص وبدقة تامة حيث ان طريقة (Tucker) هي طريقة رياضية تقوم على حد وتقليص في المصفوفات الخاصة بالصورة الخاضعة للكشف عن هوية شخص ما وكذا تتميز هذه الطريقة بعدم ضياع معلومات الصورة وهذا ما يميزها عن باقي طرق الحد الاخرى وسنرى توسع في دراسة هذه الطريقة في بحثنا هذا . ويتم التطبيق في هذا المشروع باستخدام Matlab.

Introduction générale

Le mot **biométrie** signifie « mesure + biologie » ou « mesure biologique », et désigne dans un sens très large l'étude quantitative des êtres vivants. Parmi les principaux domaines d'application de la biométrie, on peut citer l'agronomie, l'anthropologie, l'écologie et la médecine. L'usage de ce terme se rapporte de plus en plus à l'usage de ces techniques à des fins de reconnaissance, d'authentification et d'identification, le sens premier du mot biométrie étant alors repris par le terme bio-statistique.

• Avantages de la biométrie

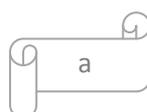
- *Sans contact voire distante, naturelle, bien acceptée.*
- *N'implique pas de coopération de l'utilisateur.*
- *Peut exploiter des bases de données existantes et un enregistrement distant.*
- *Très liée à d'autres applications (indexation, IHM, cadrage automatique, localisation).*
- *Capteurs très bon marché*

• Inconvénients de la biométrie

- *Sensible à l'environnement d'acquisition (segmentation du fond, illumination).*
- *Sensible aux expressions faciales et aux altérations de l'aspect (âge, pilosité).*
- *Potentiellement intrusif du point de vue éthique (vie privée) du fait de la non coopération.*

Le visage en 3 dimensions : Le système contient une source lumineuse infrarouge et une caméra. Cette lumière se projette sous une forme particulière, par exemple une grille, sur le visage. Puis la caméra enregistre les déformations de cette grille, ce qui permet au logiciel de reconstruire le visage en 3 dimensions. **Les avantages** : L'enregistrement est moins dépendant de la position du visage face à la caméra que lors de la reconnaissance du visage 2 D. Les variations de la lumière visible n'ont pas d'incidence car la caméra n'est sensible qu'à la lumière infrarouge.

La reconnaissance de visage : La reconnaissance de visage 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, brevets, et de conférences spécialisées. La reconnaissance de visage a de nombreuses applications en vidéosurveillance, biométrie, robotique, indexation d'images et de vidéos, recherche



Introduction générale

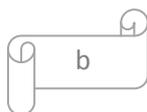
d'images par le contenu, etc...La reconnaissance de visage fait partie du domaine du traitement du signal. La reconnaissance de visage consiste à déterminer l'identité d'une personne à partir d'une image de son visage. Pour cela, il est nécessaire que l'identité de la personne soit connue au préalable, au moyen d'une ou plusieurs images de son visage, ou d'un modèle 3D de son visage.

On distingue deux types de tâches:

- **L'identification**, appelée aussi one-to-many (1:N), consiste à déterminer l'identité d'un individu parmi N identités connues, présentes dans une base.
- La **vérification**, appelée aussi one-to-one (1:1), consiste à vérifier que l'identité prétendue est bien la bonne.

Méthodes de réduction pour reconnaissance de visage : Il existe plusieurs méthodes comme PCA (*Principal Component Analysis*)(Moon et Phillips en 2002),LDA (*Linear Discriminant Analysis*)(Zhao et Chapella en 1998),GDA (*Generalized Discriminant Analysis*),DLDA(*Direct Linear Discriminant Analysis*),KFA(*Kernel Fisher Analysis*),ICA (*L'analyse en composantes indépendantes*) (M. S. Bartlett, H. M. Lades, and T. J. Sejnowski en 1998),HMM (*Hidden Markov Models*),SVM(*Machine à vecteurs de support*) mais l'importante pour ma travaille est la méthode de Tucker c'est une méthode de la reconnaissance faciale de visage, donc il ya de nombreux scientifiques et chercheurs ont étudié et nous verrons certains des chercheurs et les résultats qui les reçoivent sur la méthode de Tucker ci-dessous :

Brett W. Bader en 2006 ont appliqués les méthodes Tenseur Tucker et SVD (*Singular Value Decomposition*) [L.L&BM00].. DANIEL KRESSNER, MARTIN PLE SINGER, AND CHRISTINE TOBLER en 2007 ont appliqués la méthode Tucker et HOSVD (*Higher Order Singular Value Decomposition*) [L.L&BM00].Nadine Renard, Salah Bourennane et Jacques Blanc –Talon en 2007 ont appliqués la méthode PCA 3D (*Principal Component Analysis 3 Démétielle*),PCADR (*Dimensionality reduction*) et LRTADR (*Lower rank-(K_1, \dots, K_N) tensor approximation*) selon la base de données SAM [NR08]. Nadine Renard et Salah Bourennane en 2008 ont aussi appliqués la méthode ICA (*L'analyse en composantes indépendantes*),PCA 3D selon la base de donnée HYDICE [NR08]. L. Grasedyck en 2010 s'est intéressé à la méthode HTD (*La décomposition hiérarchique de Tucker*) et le résultat de



Introduction générale

la mise en œuvre affinée est à haute performance moderne selon la base de donnée NESTEDNESS [L.G10]. T. G. Kolda and B.W. Bader en 2012 ont appliqués les méthodes HTD selon la base de donnée HOSVD, le résultat obtenu est le même que celui de L. Grasedyck.

Nous avons choisi ce sujet en raison de son importance en tant que le processus de reconnaissance du visage est le processus d'intervention dans le système de sécurité, par conséquent, n'est pas un processus facile pour cela nous avons besoin de précision et de sophistication et de la méthode de Tucker est un moyen peu efficace et distincte du reste des méthodes mentionnées précédemment, il est caractérisé par la vitesse et de gagner du temps dans un processus d'identification pour le visage de quelqu'un et connaître son identité. Et vers nous prenons pour ce thème nous nous opposons à plusieurs questions et sont :

C'est quoi la reconnaissance de visage 3D ?

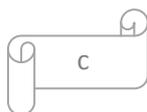
Quels sont les méthodes de réduction pour la reconnaissance de visage ?

C'est quoi le principe de la réduction de Tucker ? Et pourquoi ce choix de réduction de données ?

Pour répondre à ces questions nous parvenons à quatre chapitres présentés comme suit:

- *Chapitre 1 : Reconnaissance faciale 3D et méthodes de réduction.
- *Chapitre 2 : Etat de l'art sur la réduction de Tucker dans le SRV3D
- *Chapitre 3 : Etude de SRV basé sur réduction de Tucker
- *Chapitre 4 : Conception de la réduction de Tucker et tentative d'application au SRV3D

Finalement nous terminons notre mémoire par une conclusion générale sur notre contribution et la proposition de perspectives pour des travaux futurs.



Chapitre

Reconnaissance faciale 3D et Méthodes de Réduction

Introduction

Nous examinerons dans ce chapitre un bref aperçu sur la biométrie d'une façon générale, la reconnaissance de visage ainsi que les domaines de recherche associés qui font face à beaucoup d'obstacles et présentent des défis pour de nombreux chercheurs. Il ya beaucoup de méthodes importantes pour surmonter ces obstacles dans le domaine de la reconnaissance de visage, nous essayons dans ce chapitre de mettre en revue les plus utilisées. La biométrie consiste à identifier une personne à partir d'une ou de plusieurs caractéristiques physiologiques (empreintes digitales, visage, iris, contour de la main, etc.), ou comportementales (signature, démarche, etc.). Etymologiquement, la biométrie humaine est synonyme d'anthropologie physique [BM13]. Une autre définition de la biométrie est donnée par Roethenbaugh [BM13] : « La biométrie s'applique à des particularités ou des caractères humains uniques en leur genre et mesurables, permettant de reconnaître ou de vérifier automatiquement l'identité ».

Partie A Reconnaissance Faciale 3D

1.1 La reconnaissance biométrique [A.C09]

La reconnaissance biométrique fait référence à l'utilisation de différentes caractéristiques physiologiques (empreintes digitales, visage, rétine, iris, voix) et comportementales (manière de marcher, dynamique de la signature), appelées caractéristiques biométriques, ou simplement biométrie, pour reconnaître automatiquement les individus. Toutes les caractéristiques biométriques d'une personne sont, en fait, une combinaison de caractéristiques physiologiques et comportementales. Par exemple, les empreintes digitales sont de nature physiologique, mais la façon dont un utilisateur présente son doigt au capteur dépend du comportement de la personne. Ainsi, l'entrée d'un système de reconnaissance biométrique est une combinaison de caractéristiques physiologiques et comportementales. De même, la voix est en partie déterminée par la structure biologique qui produit la parole chez un individu et aussi par la façon dont une personne parle. Souvent, une similitude peut être constatée entre les parents, les enfants, et les frères et sœurs dans leur voix, leur démarche, et même dans leur signature. Le même argument s'applique aux visages: les visages de jumeaux peuvent être très similaires à la naissance, mais au cours du développement, les visages changent en fonction du mode de vie et du comportement de la personne. De plus, les sources de variabilité entre deux vues d'un même visage sont multiples, et peuvent même engendrer

des dis-similarités plus importantes que celles observés entre deux visages différents. Ceci se traduit par une baisse significative des performances de la reconnaissance automatique dans les conditions réelles pour le visage ainsi que pour la majorité des modalités biométriques. Ainsi la reconnaissance automatique des caractéristiques biométriques reste un problème complexe et non efficacement résolu. [A.C09]

1.2 Les caractéristiques biométriques [A.C09]

Les caractéristiques biométriques ne peuvent pas être facilement volées, falsifiées, ou partagées. Ainsi, elles sont plus fiables et sécurisées pour la reconnaissance de personne que les méthodes traditionnelles basées sur la connaissance ou la possession. Cependant ces caractéristiques physiques et comportementales doivent satisfaire plusieurs contraintes pour une grande fiabilité des systèmes biométriques. En effet, les objectifs de la reconnaissance biométrique sont la facilité d'utilisation par une reconnaissance sans carte ou code PIN, la sécurité accrue qui est traduite par la difficulté de contourner le contrôle d'accès ainsi que la plus grande performance quant à la précision et la rapidité des traitements. Ainsi, chaque caractéristique physiologique et/ou comportementale peut être utilisée comme une biométrie pour reconnaître une personne dès lors qu'elle répond à ces exigences [A.C09]:

1. ***l'universalité***, ce qui signifie que chaque personne devrait avoir cette caractéristique biométrique ;
2. ***l'unicité***, qui indique que la caractéristique biométrique est représentative d'un et un seul individu;
3. ***la facilité de mesure***, qui se réfère à la facilité avec laquelle une personne peut interagir avec un système biométrique pendant qu'elle est identifiée ou authentifiée par ce système. Cette mesure est objective et dépend du type de dispositif utilisé pour capturer l'échantillon biométrique (caméra, scanner, etc.) ;
4. ***la performance***, qui se réfère principalement à la précision de la reconnaissance, à la vitesse d'exécution et à la robustesse face aux variations intrinsèques des caractéristiques biométriques, aux bruits et aux déformations des appareils de captures. La performance d'un système dépend aussi des ressources nécessaires pour atteindre la vitesse et la précision escomptées, ainsi que les facteurs opérationnels et environnementaux qui influent sur la vitesse et la précision de la reconnaissance; Cependant, dans un système biométrique pratique, il existe un certain nombre d'autres paramètres qui doivent être examinés, notamment:
5. ***L'acceptation*** par l'utilisateur qui reflète l'ampleur des inquiétudes et des objections que l'utilisation d'une technologie biométrique donnée tend à susciter. Dans certains

pays, la reconnaissance faciale est mal acceptée ; dans d'autres pays, la reconnaissance des empreintes digitales a des connotations de criminalité. La mesure de l'acceptation reste très subjective et varie d'une personne à l'autre et d'un pays à l'autre, suivant le régime de protection des données en vigueur, le contexte culturel et les attentes personnelles des utilisateurs ;

6. **Le contournement**, qui reflète combien il est facile de tromper le système par des méthodes frauduleuses.
7. **La permanence** ou la stabilité qui est définie par la constance d'une caractéristique biométrique au cours du développement normal et du vieillissement d'une personne. En principe, plus une caractéristique est stable, moins il est nécessaire de mettre à jour les caractéristiques personnelles ou de réenregistrer la personne. Un système biométrique pratique doit avoir une précision acceptable et une vitesse de reconnaissance raisonnable vis-à-vis des ressources requises, inoffensif pour les utilisateurs, accepté par la population, et suffisamment robuste à l'encontre des méthodes frauduleuses. De nombreuses modalités biométriques sont utilisées dans diverses applications (cf. Figure 1.1). Chaque modalité biométrique a ses forces et ses faiblesses et le choix dépend généralement de l'application à traiter. Aucune modalité biométrique ne répond effectivement aux exigences de toutes les applications. Ainsi, l'utilisation de plusieurs modalités (multi-modalités) biométriques pour l'identification d'un individu augmente aussi bien la précision que la couverture de la population tout en laissant décroître la vulnérabilité à l'imposture. [A.C09]

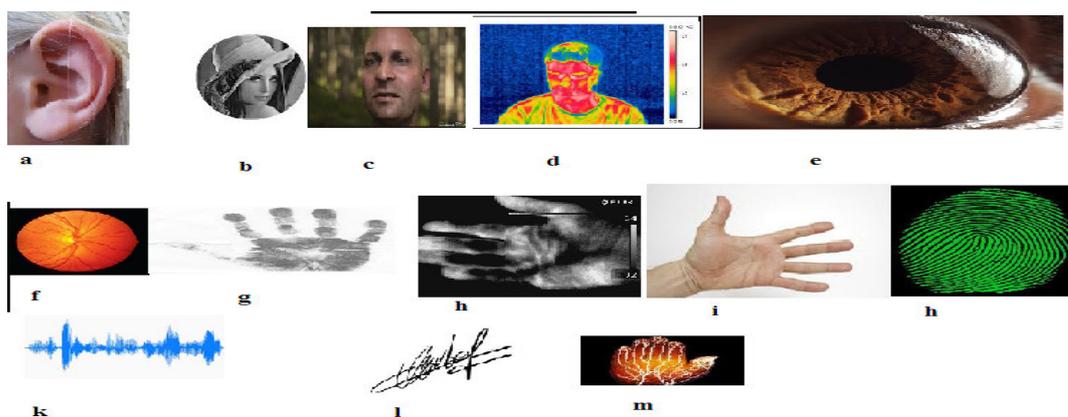


Figure 1.1 Les principales caractéristiques biométriques
 a) forme de l'oreille, b) visage 2D, c) visage 3D, d) visage infrarouge, e) iris, f) rétine, g) empreinte de la main, h) thermo-gramme de la main, i) forme de la main, j) empreinte digitale, k) voix, l) signature et m) réseau veineux de la main .

1.3 Mode de reconnaissance [A.C09]

Suivant l'application traitée, un système biométrique peut fonctionner suivant un mode de reconnaissance positif ou négatif:

Dans une reconnaissance *positive*, le système permet de déterminer si la personne est celle qu'elle prétend être, de manière implicite ou explicite. Le but d'une reconnaissance positive est d'empêcher plusieurs personnes d'utiliser la même identité. Par exemple, si une personne unique est autorisée à entrer dans une certaine zone sécurisée, le système lui donnera exclusivement l'accès. Si le système arrive à faire correspondre la signature enregistrée de cette personne à la signature extraite de la biométrie acquise, cela correspond à une acceptation, et sinon à un rejet.

Pour une demande de reconnaissance négative, le système permet de déterminer si la personne correspond à ce qu'elle nie être de manière explicite ou implicite. L'objet de la reconnaissance est négatif pour empêcher une seule personne d'utiliser de multiples identités. Par exemple, pour une application de prestations sociales, le système enregistre dans sa base de données les personnes qui ont déjà bénéficié de prestations. Si une personne malhonnête souhaite recevoir les prestations une deuxième fois en se proclamant comme étant une tierce personne, le système doit vérifier si celle-ci correspond à un des bénéficiaires enregistrés dans la base de données. Si le système arrive à faire correspondre la signature extraite de la biométrie acquise de cette personne avec une des signatures de la base de données, cela correspond à un rejet, et sinon à une acceptation. Notons que les méthodes traditionnelles d'authentification, telles que les mots de passe, les codes PIN, les clés, les jetons peuvent être utilisées uniquement pour une reconnaissance positive. La reconnaissance négative, quant à elle, ne peut être établie que par la biométrie. En outre, la reconnaissance positive peut fonctionner aussi bien en mode de vérification ou d'identification, mais les applications qui nécessitent une reconnaissance négative ne peuvent être déployées qu'en mode d'identification: en fait, le système doit parcourir la totalité de ses bases de données pour prouver que les données saisies ne sont pas présentes.

1.4 Techniques 3D de reconnaissance de visages [BM13]

Nous avons passé en revue les méthodes de reconnaissance 2D de visages. Malgré les avancées réalisées ces dernières années, les techniques de reconnaissance 2D de visages robustes aux différents facteurs de variabilité (éclairage, pose, occultation) sont loin d'être développées. La reconnaissance 3D de visages constitue une alternative prometteuse pour surmonter ces problèmes, surtout depuis l'apparition de dispositifs d'acquisition 3D

performant. L'avantage principal des approches basées modèle 3D réside dans le fait que le modèle 3D conserve toutes les informations sur la géométrie de visage, ce qui permet d'avoir une représentation réelle de ce dernier. Dans cette section, après avoir rapidement évoqué les systèmes d'acquisition 3D, nous présenterons les travaux récents sur la reconnaissance 3D de visages. [BM13]

1.5 Approches modèle [BM13]

Ces approches construisent, à partir des points 3D, des modèles de visages qu'elles utilisent par la suite pour la reconnaissance. Blanz et al ont proposé une méthode basée sur un modèle 3D « *morphable* » du visage. L'ensemble des visages est représenté par un espace vectoriel. La base de données contient 100 visages d'hommes et 100 visages de femmes acquis avec un *scanner laser CyberwareTM 3030PS*. Les points 3D des modèles de visages générés sont représentés par leurs coordonnées cylindriques définies par rapport à un axe vertical. Pour chaque visage de référence, les coordonnées et les valeurs de texture de tous les *sommets* ($n = 75\ 972$) sont regroupées pour former deux vecteurs : un vecteur de forme et un vecteur de texture. Une fois le modèle générique créé, l'étape suivante consiste à l'ajuster sur l'image 2D à partir des paramètres de forme et de texture. La synthèse d'image permet de rendre les nouvelles positions projetées des sommets du modèle 3D, à l'aide l'illumination et la couleur extraites. Enfin, l'étape de reconnaissance est réalisée en mesurant la distance de Mahalanobis entre la forme et les paramètres de texture des modèles dans la galerie et le modèle d'ajustement. L'identification a été évaluée sur deux bases de données d'images à accès libre : CMU-PIE et FERET. Un taux de reconnaissance de 95 % sur l'ensemble de données CMU-PIE et 95.9 % sur l'ensemble de données FERET a été obtenu. [BM13]

1.6 Approches 3D [BM13]

Elles sont subdivisées en deux catégories : les approches basées surface qui utilisent la géométrie de la surface du visage et les approches holistiques 3D.

1.6.1 Approches basées la surface 3D [L.B12]

Dans ce cas, le problème de la reconnaissance 3D de visages est celui de l'alignement de deux surfaces 3D qui modélisent les deux visages à appateler-rier. L'algorithme généralement utilisé est l'algorithme du plus proche voisin itéré, ou ICP (Iterative Closest Point). Il consiste en une optimisation alternée d'appariements et de transformations. Ainsi, à partir d'une transformation initiale, les deux étapes suivantes sont réitérées :

(a) Mise en correspondance (plus proche voisin) : on apparie chaque primitive du modèle transformé avec la primitive la plus proche dans la scène.

(b) Recalage : la transformation (translation + rotation) est généralement calculée aux sens des moindres carrés, surtout si l'on travaille avec des points. Si l'on possède une information d'incertitude, on peut l'utiliser dans les étapes terminales pour affiner la solution. Une approche basée sur l'algorithme ICP est donnée par Cook et al . Ils utilisent l'algorithme ICP pour établir la correspondance entre les surfaces 3D afin de compenser des erreurs dues à la nature non rigide des visages. En effet, les visages sont comparés en employant un modèle statistique, en l'occurrence un Mélange de Modèles Gaussiens (MMG). La distribution des erreurs est alors paramétrée. Les auteurs ont réalisé des expériences sur la base de données 3D RMA une base créée au sein du Signal et Image Center du Royal Military Academy (Bruxelles - Belgique). Un dispositif de vision active (structured lightbased) est utilisé pour l'acquisition. Cette base contient les images de 120 individus, capturées en deux sessions, sous formes de points 3D non maillés. Irfanoglu et al proposent une autre approche qui utilise l'algorithme ICP pour mettre en correspondance les points caractéristiques de visages. Ils ont développé une méthode pour obtenir un appariement dense de points du maillage contenant les points présents sur tous les visages, de sorte que l'alignement de visage soit trivialement obtenu. Une fois que l'appariement dense est établi, la distance PSD "*Point Set Distance*", qui représente une approximation discrète du volume entre les surfaces faciales, est employée pour calculer la distance entre deux nuages différents de points. Bien qu'ICP soit un outil puissant pour évaluer la similitude entre deux visages, les méthodes basées ICP ne sont pas capables de traiter le problème de changement d'expression, car elles considèrent le visage 3D comme un objet rigide, ce qui n'est pas le cas dans la réalité. Pour résoudre ce problème, plusieurs approches ont été développées. Moreno et al proposent de segmenter le visage en plusieurs régions en utilisant la courbure gaussienne. Ainsi, chaque modèle facial 3D est subdivisé en 86 régions différentes grâce à un algorithme de segmentation qui exploite les signes de la médiane et des courbures gaussiennes pour isoler les régions ayant des courbures significatives. Cet espace de caractéristiques est ensuite réduit pour augmenter l'efficacité de la méthode. Enfin, un vecteur de caractéristiques est créé pour chaque sujet. Les expérimentations ont été menées sur un ensemble de 420 modèles 3D de visages appartenant à 60 sujets, comprenant des images avec des variations de lumière, de rotation et d'expressions de visages. [L.B12]

1.6.2 Approches holistiques 3D [BM13]

Les techniques holistiques comme l'ACP ont été largement utilisées dans la reconnaissance faciale 2D. Plus récemment, ces techniques ont été aussi étendues aux données 3D de visage. Ainsi, Heshner et al ont appliqué l'ACP sur une base de données

contenant 222 images de profondeur (37 personnes), avec six expressions faciales différentes. Ils ont obtenu un taux de reconnaissance de 83%. Heseltine et al. ont développé deux approches qui appliquent l'ACP sur des représentations tridimensionnelles du visage. La première méthode utilise l'approche basée ACP « eigensurface » sur des modèles 3D maillés de visage de la base de données de l'Université de York (The 3D Face Database, 2003). Elle arrive à un taux de reconnaissance de 87%. La deuxième approche est une extension de la méthode Ficherface 2D de Belhumeur aux données surfaciques de visage 3D. Les tests ont été effectués sur une base de données contenant 1770 modèles 3D maillés de 280 sujets ayant différentes poses et expressions faciales. Le taux de reconnaissance obtenu est de 88%.

Les techniques basées ACP ont également été combinées avec d'autres méthodes de classification, comme le modèle caché de Markov (EHMM) puis appliquées à la reconnaissance 3D de visages. Enfin, d'autres approches basées sur l'Analyse Discriminante Linéaire ou l'Analyse des Composantes Indépendantes ont aussi été développées pour l'analyse des données 3D de visages. [BM13]

1.6.3 Approche géométrique ou locale 3D [L.B12]

L'introduction de la géométrie 3D du visage dans la reconnaissance de personnes est considérée comme une voie très prometteuse pour améliorer sa performance. Cependant, l'un des défis ouverts est celui de la robustesse de ces approches 3D aux expressions faciales. Un exemple de ces approches est présenté par Bronstein et al. qui propose une nouvelle représentation basée sur la nature isométrique de la surface faciale. Sous cette hypothèse, les auteurs construisent une représentation invariante à l'expression appelé formes canonique et utiliser la distance géodésique comme métrique des surfaces faciales pour les comparer. Pour conserver la consistance des géodésiques en cas de bouche ouverte, les mêmes auteurs ont amélioré la solution dans [1], en détectant et en supprimant les lèvres, à la fois pour l'expression neutre et non-neutres. Les approches géométriques ou locales 3D sont subdivisées en deux catégories principales : les approches basées courbes faciales 3D qui utilisent la géométrie des courbes du visage et les approches basées sur les caractéristiques faciales. [L.B12]

1 Approches basées sur les courbes faciales 3D [L.B12]

Approches basées sur les courbes. Dans cette catégorie, nous nous sommes intéressés aux méthodes d'extraction des courbes faciales 3D. Ces courbes peuvent être explicitement analysées et comparées. Plusieurs représentations de courbes ont été utilisées dans la littérature. Samir et al. ont utilisé les courbes planes de niveaux. Les auteurs ont utilisé l'analyse de forme des courbes planes afin d'analyser les déformations du visage. En revanche, aucune étude n'a été faite pour la sélection des courbes pertinentes. Dans les mêmes

auteurs ont utilisé la distance géodésique pour extraire les courbes de niveaux 3D. Selon l'étude de Bronstein et al les changements de la distance géodésique dus aux expressions faciales est négligeable comparée aux changements de la distance euclidienne. Dans les auteurs ont utilisé des points échantillons prélevés de l'intersection entre les courbes de niveaux et les profils radiaux, qui ont comme origine le bout du nez, ensuite ils ont calculé les distances euclidiennes entre les points correspondants des différents visages. Cette approche, permet de sélectionner des sous ensembles de courbes faciales pour une mise en correspondance efficace du visage. Drira et al ont exploré l'utilisation de la forme de différents types de nez. Les auteurs extraient les courbes iso-géodésiques, riété Riemannienne de courbes fermées. La même approche permet de faire des calculs statistiques (moyennes) pour une recherche efficace dans des bases de visages 3D. Dans les mêmes auteurs ont proposé une approche similaire afin d'analyser la forme des courbes radiales 3D. Dans Berretti et al. Segmentent la surface faciale en bandes géodésiques (en se basant sur une distance intrinsèque) et utilisent une approche d'appariement de graphes pour comparer les visages. Mpiperis et al proposent une paramétrisation polaire, au sein de cette nouvelle paramétrisation intrinsèque, les attributs du visage sont invariants aux déformations isométriques. Quand la bouche est ouverte, cette paramétrisation n'est plus valable. les auteurs proposent donc de déconnecter les lèvres sur toutes les surfaces faciales et changent cette paramétrisation en se basant sur trois points de références. [L.B12]

2 Approches basées sur les caractéristiques faciales

Approches basées sur la sélection des caractéristiques. Plusieurs méthodes ont été proposées pour analyser le pouvoir discriminant des différentes parties du visage. Daniyal et al ont proposé un algorithme dans lequel le visage est représenté comme un vecteur de distance entre les points d'intérêts (landmarks) du visage. Ils ont sélectionné les points par brute-forcing des combinaisons possibles des points utilisés/nonutilisés et ils ont comparé les taux de reconnaissance. Ils ont conclu que la meilleure sélection correspond aux points situés autour des yeux et du nez. Kakadiaris et al utilisent un modèle de visage annoté qui est déformé élastiquement pour s'adapter à chaque visage, ainsi lui permettre automatiquement l'annotation de ses différentes zones comme le nez, yeux, bouche, etc. Pour résoudre le problème des expressions faciales, les auteurs classent les visages en utilisant les coefficients d'ondelettes qui représentent les zones non affectées par les expressions, telles que les yeux et le nez. Cependant, le meilleur taux de reconnaissance est atteint pour le visage entier, ce qui implique que le rejet de certaines zones, affectées par les déformations du visage, conduit à la perte de l'information discriminante. Dans les auteurs proposent un système de

reconnaissance de visage 3D utilisant deux dispositifs différents d'acquisition de données 3D : un scanner laser 3D et une lumière structurée. A partir des mesures 3D, huit points caractéristiques du visage (géométriquement invariables) sont extraits puis utilisés pour calculer un vecteur caractéristique comprenant les distances et les angles entre ces points. Pour la phase de reconnaissance, les auteurs ont implémenté deux algorithmes différents : le premier basé profondeur (depth-based DP) et le second utilise les SVM (feature-based SVM). Huang et al. [1] présentent une nouvelle représentation géométrique des visages 3D afin d'améliorer la spécificité des images de profondeur. Cette représentation du visage est basée sur la multi-échelle des ELBP (Extended Local Binary Patterns) et permet une description précise et rapide de la variation de la forme locale du visage. Récemment, Wang et al. [2] calculent la carte de différence de forme signée (SSDM), calculée entre deux visages alignés, comme une représentation intermédiaire pour la comparaison de formes. Basé sur les SSDMs, trois types d'opérateurs ont été utilisés pour coder à la fois la similarité locale et le changement des caractéristiques de différentes formes du visage. Les caractéristiques locales les plus discriminantes ont été sélectionnées, par AdaBoost, en les utilisant comme des classifieurs faibles. Les caractéristiques individuelles sont du type : Haar-like, Gabor et local binary pattern (LBP). En utilisant les caractéristiques similaires, Li et al. [3] proposent de concevoir un dispositif commun et un système de classement afin de collecter différentes caractéristiques géométriques de bas-niveau, telles que la courbure, et les classer selon leurs sensibilités aux expressions faciales. [L.B12]

1.7 Principales difficultés de la reconnaissance de visage [BM13]

Pour le cerveau humain, le processus de la reconnaissance de visages est une tâche visuelle de haut niveau. Bien que les êtres humains puissent détecter et identifier des visages dans une scène sans beaucoup de peine, construire un système automatique qui accomplit de telles tâches représente un sérieux défi. Ce défi est d'autant plus grand lorsque les conditions d'acquisition des images sont très variables. Il existe deux types de variations associées aux images de visages : inter et intra sujet. La variation inter-sujet est limitée à cause de la ressemblance physique entre les individus. Par contre la variation intra-sujet est plus vaste. Elle peut être attribuée à plusieurs facteurs que nous analysons ci-dessous. [BM13]

1.7.1 Changement d'illumination

L'apparence d'un visage dans une image varie énormément en fonction de l'illumination de la scène lors de la prise de vue (voir figure 1.2). Les variations d'éclairage rendent la tâche de reconnaissance de visage très difficile. En effet, le changement d'apparence d'un visage dû à l'illumination, se révèle parfois plus critique que la différence

physique entre les individus, et peut entraîner une mauvaise classification des images d'entrée. Ceci a été expérimentalement observé dans Adini et al où les auteurs ont utilisé une base de données de 25 individus. L'identification de visage dans un environnement non contrôlé reste donc un domaine de recherche ouvert. Les évaluations FRVT ont révélé que le problème de variation d'illumination constitue un défi majeur pour la reconnaissance faciale. [BM13]



Figure 1.2 Exemple de variation d'éclairage. [B.S12]

1.7.2 Variation de pose

Le taux de reconnaissance de visage baisse considérablement quand des variations de pose sont présentes dans les images. Cette difficulté a été démontrée par des tests d'évaluation élaborés sur les bases FERET et FRVT. La variation de pose est considérée comme un problème majeur pour les systèmes de reconnaissance faciale. Quand le visage est de profil dans le plan image (orientation $< 30^\circ$), il peut être normalisé en détectant au moins deux traits faciaux (passant par les yeux). Cependant, lorsque la rotation est supérieure à 30° , la normalisation géométrique n'est plus possible (voir figure 1.3). [BM13]



Figure 1.3 Exemples de variation de poses. [B.S12]

1.7.3 Expressions faciales

Un autre facteur qui affecte l'apparence du visage est l'expression faciale (voir figure 1.4). La déformation du visage qui est due aux expressions faciales est localisée principalement sur la partie inférieure du visage. L'information faciale se situant dans la partie supérieure du visage reste quasi invariable. Elle est généralement suffisante pour effectuer une identification. Toutefois, étant donné que l'expression faciale modifie l'aspect du visage, elle entraîne forcément une diminution du taux de reconnaissance. L'identification de visage avec expression faciale est un problème difficile qui est toujours d'actualité et qui reste non résolu.

L'information temporelle fournit une connaissance additionnelle significative qui peut être utilisée pour résoudre ce problème [BM13].



Figure 1.4 Exemples de variation d'expressions. [Sit1]

1.7.4 Présence ou absence des composants structurels [BM13]

La présence des composants structurels telle que la barbe, la moustache, ou bien les lunettes peut modifier énormément les caractéristiques faciales telles que la forme, la couleur, ou la taille du visage. De plus, ces composants peuvent cacher les caractéristiques faciales de base causant ainsi une défaillance du système de reconnaissance. Par exemple, des lunettes opaques ne permettent pas de bien distinguer la forme et la couleur des yeux, et une moustache ou une barbe modifie la forme du visage. [BM13]

1.7.5 Occultations partielles

Le visage peut être partiellement masqué par des objets dans la scène, ou par le port d'accessoire tels que lunettes, écharpe... Dans le contexte de la biométrie, les systèmes proposés doivent être non intrusifs c'est-à-dire qu'on ne doit pas compter sur une coopération active du sujet. Par conséquent, il est important de savoir reconnaître des visages partiellement occultés. Gross et al ont étudié l'impact du port de lunettes de soleil, et du cache-nez occultant la partie inférieure du visage sur la reconnaissance faciale. Ils ont utilisé la base de données AR. Leurs résultats expérimentaux semblent indiquer que, dans ces conditions, les performances des algorithmes de reconnaissance restent faibles. [BM13]

Partie B Méthodes d'analyse et de réduction de données

1.8.1 Analyse en composantes principales (*Principal Component Analysis* ou PCA) est dérivé de la transformation de Karhunen-Loeve. Soit une représentation vectorielle de m dimensions de chaque visage dans un ensemble des images d'apprentissage, PCA a tendance à trouver un sous-espace de n dimensions dont les vecteurs de base correspondent à la direction la plus variable dans l'espace original. Ce nouveau sous-espace est normalement plus faible en dimension ($n \ll m$). Si les éléments d'image sont considérés comme des variables aléatoires, les vecteurs de base de PCA sont définis comme les vecteurs propres de la matrice de

dispersion Les vecteurs de caractéristiques pour l'algorithme PCA sont formés par concaténation des valeurs des pixels à partir des images. Ces vecteurs bruts de caractéristiques ne sont pas seulement grands, mais aussi fortement corrélés. La méthode PCA transforme les vecteurs de caractéristiques à partir d'un grand espace fortement corrélé à un petit sous-espace qui n'a pas de covariance entre les caractéristiques. La méthode PCA possède deux propriétés utiles lorsqu'elle est utilisée en reconnaissance de visages. La première est qu'elle peut être utilisée pour réduire la dimension des vecteurs de caractéristiques. Cette réduction de dimension peut être effectuée d'une façon avec perte ou sans perte. Lorsqu'elle est appliquée d'une façon avec perte, les vecteurs de base sont tronqués de la matrice de transformation. Il est supposé que ces vecteurs correspondent aux informations inutiles telles que les variations d'éclairage ou le bruit. Si aucun vecteur de base n'est supprimé, la méthode est une transformation sans perte et il devrait être possible d'obtenir une reconstruction parfaite pour les données d'apprentissage basée sur les vecteurs comprimés de caractéristiques. La seconde propriété utile est que PCA élimine toutes les covariances statistiques dans les vecteurs transformés de caractéristiques. Cela signifie que la matrice de covariance pour les vecteurs transformés de caractéristiques sera toujours diagonale. Cette propriété est exploitée pour certaines mesures de distance telles que L1, les classifieurs bayésiens. [PHTN10]

1.8.1.1 Application de l'ACP à la reconnaissance de visage

La comparaison entre les deux images est faite par une mesure métrique entre les projetés des images dans l'espace réduit : $d(\Phi^1, \Phi^2) = \|\psi(\Phi^1), \psi(\Phi^2)\|$, avec $\|\ \ \|$ la mesure métrique considérée. Différentes mesures ont été proposées dans la littérature, comme la distance L1, la distance euclidienne L2, la similarité cosinus (ou "mesure cosinus"), la distance de Mahalanobis. Voici leurs définitions :

- **L1 :** $d(\Phi^1, \Phi^2) = |\psi(\Phi^1), \psi(\Phi^2)| = \sum_{i=1}^r |a_i - b_i|$ (1.1)

avec a_i et b_i les coordonnées de projection de $\psi(\Phi^1)$ et $\psi(\Phi^2)$

- **L2 :** $d(\Phi^1, \Phi^2) = \|\psi(\Phi^1), \psi(\Phi^2)\| = \sum_{i=1}^r (a_i - b_i)^2$ (1.2)

- **Angle :** $d(\Phi^1, \Phi^2) = \frac{\psi(\Phi^1) - \psi(\Phi^2)}{\|\psi(\Phi^1)\| \|\psi(\Phi^2)\|} = \frac{\sum_{i=1}^r a_i b_i}{\sqrt{\sum_{i=1}^r (a_i)^2 \sum_{i=1}^r (b_i)^2}}$ (1.3)

- **Mahalanobis :** $d(\Phi^1, \Phi^2) = \sum_{i=1}^r \frac{1}{\sqrt{\lambda_i}} a_i b_i$ (1.4)

Moon et Phillips ont étudié dans [HM&PJPH98] les performances de ces 4 mesures sur la base FERET [PHPJ,WH,HJ&RP98]. Une autre étude [BDWY&RB02], faite en 2002, a repris les travaux de Moon et Phillips et a confirmé aussi que la distance de Mahalanobis dépassait

les autres distances en termes de performance et qu'une fusion de ces mesures améliorerait la reconnaissance. Ces évaluations ont été faites sur la même base FERET.

1.8.1.2 EigenFace modulaire [Sit2]

Cette méthode possède le même principe que les EigenFaces, mais appliquée à des parties précises du visage comme les yeux. Mais elle rencontre le problème de non précision lors de la localisation des points caractéristiques du visage avant l'application de la méthode.

1.8.2 Analyse discriminante linéaire (*Linear Discriminant Analysis* ou LDA) trouve dans l'espace sous-jacent les vecteurs qui discriminent le mieux parmi des classes. Pour tous les échantillons de toutes les classes, la matrice de dispersion inter-classe S_b et la matrice de dispersion intra-classe S_w sont définies. L'objectif est de maximiser S_b en minimisant S_w , en d'autres termes, de maximiser le rapport $det | S_b | / det | S_w |$. Ce rapport est maximisé lorsque les vecteurs colonnes de la matrice de projection sont les vecteurs propres de $(S_w^{-1} \times S_b)$. LDA a tendance de produire une transformation linéaire qui souligne les différences entre classes en réduisant les différences au sein des classes. L'objectif est de former un sous-espace qui est linéairement séparable entre classes. Lorsque la méthode LDA est utilisée en reconnaissance de visages, chaque identification humaine forme une classe. LDA nécessite des données d'apprentissage qui devraient posséder plusieurs images par identification. LDA peut être utilisée d'une façon combinée avec PCA : la méthode PCA est d'abord utilisée pour réduire la dimensionnalité des vecteurs de caractéristiques. Après, la méthode LDA est effectuée sur les données pour réduire encore la dimension de telle sorte que les caractéristiques qui distinguent bien les classes sont conservées. Une matrice finale de transformation est obtenue en multipliant les vecteurs de PCA et de LDA pour produire une matrice pleine de transformation. [PHTN10]. Le résultat final de LDA est le même que celui de PCA. L'algorithme produit un ensemble des vecteurs LDA de base. Ces vecteurs de base produisent une transformation des vecteurs de caractéristiques. Comme l'algorithme PCA, les mesures de distance peuvent être utilisées sur les vecteurs de caractéristiques LDA. La première mesure est la matrice d'éparpillement intra-classes S_w ("within-class scatter matrix") qui est définie par :

$$S_w = \sum_{j=1}^c \sum_{i=1}^{N_j} (x_i^j - \mu_j)(x_i^j - \mu_j)^T \tag{1.5}$$

Avec x_i^j le i ème échantillon de la classe j , μ_j la moyenne de la classe j , c le nombre de classes et N_j le nombre d'échantillons de la classe j . La deuxième mesure est la matrice d'éparpillement inter-classes S_b ("between-class scatter matrix") qui est définie par :

$$S_b = \sum_{j=1}^c (\mu_j - \mu)(\mu_j - \mu)^T \tag{1.6}$$

Avec μ la moyenne de tout l'échantillon. Le but est de maximiser les distances inter-classes tout en minimisant les distances intra-classes, ce qui revient à retrouver la matrice de transformation \mathbf{W} qui maximise le critère $J(\mathbf{W}) = \frac{WTSbW}{WTSwW}$ donc \mathbf{W} est optimale pour : $\mathbf{W}_{opt} = \mathit{arg\ max}_w \left(\frac{WTSbW}{WTSwW} \right) = [w_1, w_2, \dots, w_m]$. Ce problème est ramené à un problème de recherche des vecteurs propres de la matrice $S^{-1}_w S_b$.

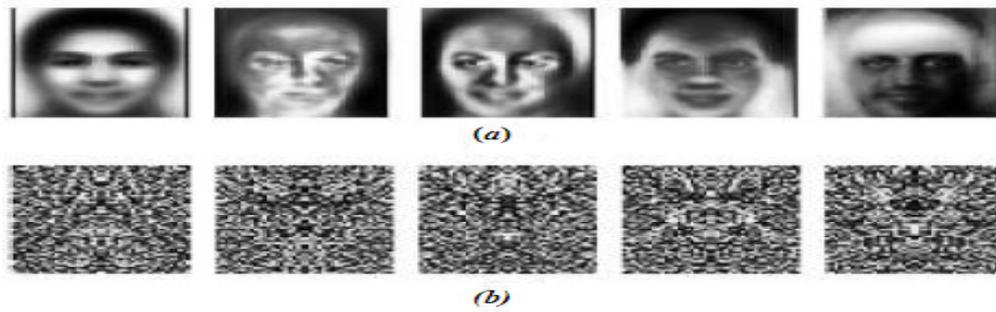


Figure 1.5 Cinq premiers visages propres construits avec l'approche [A.M09]
(a) ACP ; (b) LDA

1.8.3 Analyse discriminante généralisée

(Generalized Discriminant Analysis ou GDA) [PHTN10]

C'est une généralisation de la méthode LDA classique pour résoudre des problèmes non linéaires . Soit $\Phi : Z \in R^N \rightarrow \Phi (Z) \in F$ est une cartographie non linéaire de l'espace d'entrée a un espace de caractéristiques de grande dimension ou les différentes classes d'objets sont censées être linéairement séparables. L'idée de GDA est d'effectuer une analyse LDA classique dans l'espace de caractéristiques F au lieu de l'espace d'entrée R^N . Soient S_b et S_w respectivement les matrices de dispersion inter-classe et intra-classe dans l'espace de caractéristiques F, comme suivant :

$$S_b = \frac{1}{L} \sum_{i=1}^c C_i (\overline{\Phi_i} - \overline{\Phi}) (\overline{\Phi_i} - \overline{\Phi})^T \quad (1.7)$$

$$S_w = \frac{1}{L} \sum_{i=1}^c \sum_{j=1}^{C_i} C_{ij} (\Phi_{ij} - \overline{\Phi_i}) (\Phi_{ij} - \overline{\Phi_i})^T \quad (1.8)$$

Où $\Phi_{ij} = \Phi (Z_{ij})$, $\overline{\Phi_i} = \frac{1}{C_i} \sum_{j=1}^{C_i} \Phi (Z_{ij})$ est la moyenne de la classe Z_i , $\overline{\Phi} = \frac{1}{L} \sum_{i=1}^c \sum_{j=1}^{C_i} \Phi (Z_{ij})$ est la moyenne de l'ensemble, et C_i est le nombre d'élément dans Z_i , avec $L = \sum_{i=1}^c C_i$. LDA détermine un ensemble des vecteurs discriminants optimaux, notes par $\{\Psi_k\}$, $k = 1..M$, afin que le rapport des dispersions inter-classe et intra-classe soit maximisé. En supposant que $\Psi = [\Psi_1, \dots, \Psi_M]$, la maximisation peut être obtenue en résolvant le problème suivant de valeur propre :

$$\Psi = \mathit{arg\ max}_\Psi \frac{|\Psi Tsb\Psi|}{|\Psi Tsw\Psi|} \quad (1.9)$$

L'espace de caractéristiques F peut être considère comme un espace de linéarisation. Cependant, sa dimension peut être arbitrairement grande, et peut-être infinie. Heureusement,

$\Phi(Z)$ n'est pas nécessaire et l'espace de caractéristiques peut devenir implicite en utilisant les méthodes de noyau dans lesquelles les produits dans F sont remplacés par une fonction de noyau dans l'espace d'entrée R^N de sorte que la cartographie non linéaire soit effectuée implicitement dans R^N . Dans les tâches de reconnaissance de visages, dans la plupart des cas, le nombre d'échantillons d'apprentissage L est beaucoup plus faible que la dimension de R^N (pour LDA) ou F (pour GDA), cela conduit à une matrice de dispersion dégénérée M_{intra} . [PHTN10]

1.8.4 Analyse discriminante linéaire directe (*Direct Linear Discriminant Analysis* ou DLDA) est un algorithme développé par Yang afin d'éviter les lacunes existantes dans les solutions traditionnelles pour les problèmes d'échantillon de petite taille. L'idée de l'algorithme est que l'espace nul de S_w peut contenir des informations discriminantes importantes si la projection de S_b n'est pas nulle en cette direction, et qu'aucune information importante sera perdue si l'espace nul de S_b est jeté. En supposant que A et B représentent respectivement les espaces nuls de S_b et S_w , les espaces de complément de A et B peuvent être écrits comme $A' = R^N - A$ et $B' = R^N - B$. Par conséquent, le sous-espace discriminant optimal recherché par l'algorithme DLDA est l'espace d'intersection $(A \cap B)$ [PHTN10].

L'idée de la LDA est de maximiser le critère de Fisher. Cela revient à rechercher une matrice A qui diagonalise simultanément les deux matrices d'éparpillement S_w et S_b :

$$AS_wA^T = I; AS_bA^T = \Lambda \quad (1.10)$$

Avec Λ matrice diagonale ordonnée par ordre décroissant. Dans [L.F.C, H.Y.M. L&al00], Chen a démontré que l'espace nul de S_w (l'espace nul de $S_w = \{x | S_w x = 0; x \in R^n\}$) contient de l'information discriminante et qu'un passage par une réduction d'espace par ACP supprime cette information. Concrètement, si pour un vecteur " a " de l'espace d'entrée, $S_w a = 0$, et $S_b a \neq 0$, alors $\frac{|a^T S_b a^T|}{|a^T S_w a^T|}$ est maximisée. Il suffit alors de garder l'espace nul de la matrice d'éparpillement inter-classes S_w . Cette méthode est bien meilleure que la LDA classique dans le sens où les vecteurs de projection obtenus maximisent le critère de Fisher sans supprimer l'information discriminante. Une nouvelle approche, appelée la DLDA directe, a été proposée dans afin de maximiser le critère de Fisher. La première étape de cette approche est la diagonalisation de la matrice S_b par le calcul de la matrice V : $V^T S_b V = \Lambda$. Avec $V^T V = I$. Le problème est alors ramené à la résolution d'un problème de valeurs propres. Chaque vecteur de la matrice V est un vecteur propre de la matrice S_b et Λ contient toutes les valeurs propres de S_b . Comme la matrice S_b peut être singulière, Λ peut contenir des valeurs propres nulles ou très faibles. Ces valeurs ainsi que les vecteurs propres associés doivent être écartés car une

projection dans la direction de ces vecteurs n'apporte aucune information discriminante entre les classes. Soit Y la sous-matrice composée des m premières colonnes de V (Y étant de dimension $m \times n$ et n étant la taille de l'échantillon d'entrée) : $Y^T S_b Y = D_b > 0$

Avec D_b la sous-matrice de Λ , de dimension $m \times m$. En multipliant par $D^{-1/2T}_b$ et $D^{-1/2}_b$ de part et d'autre on aboutit à :

$$(YD^{-1/2}_b)^T S_b (YD^{-1/2}_b) = I$$

En posant $Z = Y^T D^{-1/2}_b$, nous obtenons : $Z^T S_b Z = D_b$ (1.11)

On constate que Z permet de réduire la dimension de S_b de $n \times m$ à $m \times m$. Considérons la diagonalisation de la matrice $Z^T S_w Z$ par la résolution du problème des valeurs propres :

$$U^T Z^T S_w Z U = D_w$$
 (1.12)

Avec $U^T U = I$. D_w peut contenir des valeurs nulles sur la diagonale. L'objectif est de minimiser l'éparpillement des inter-classes. Il est donc important de garder les vecteurs de projection associés aux valeurs propres les plus faibles, spécialement les valeurs nulles, et d'écarter ceux associés aux valeurs propres les plus fortes. En posant la matrice $A = U^T Z^T$, A permet la diagonalisation du numérateur et du dénominateur du critère de Fisher :

$$A S_w A^T = D_w; A S_b A^T = I$$
 (1.13)

En posant $\Phi = D^{-1/2}_w A$, nous obtenons la matrice de projection qui répond au critère de Fisher. La figure 1.6 montre la chaîne de l'algorithme de la DLDA. [PHTN10]

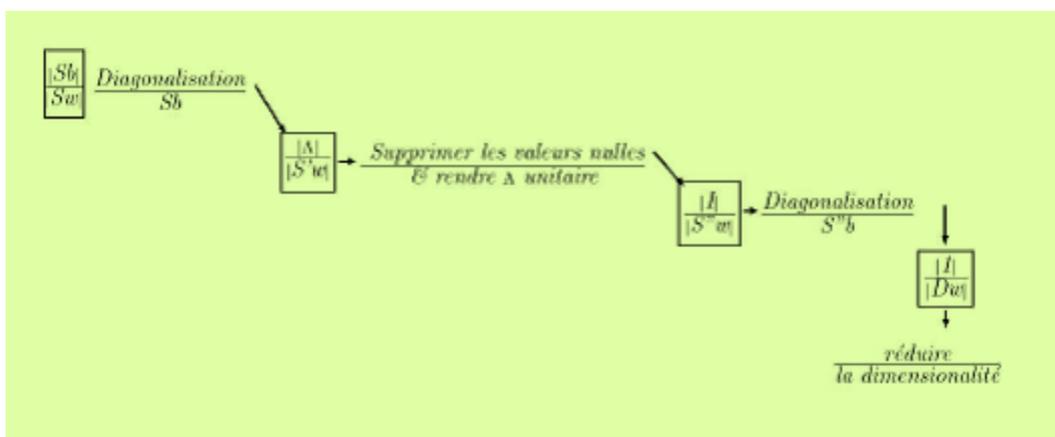


Figure 1.6 Diagramme de l'algorithme de la LDA directe (DLDA). [HY&JY01].

1.8.5 Analyse par noyau de Fisher [PHTN10] (Kernel Fisher Analysis ou KFA) est une méthode utilisée pour trouver un espace discriminant de transformation, les données d'entrée sont d'abord projetées dans un espace implicite de caractéristiques F à travers la cartographie non linéaire $\Phi : x \in R^N \rightarrow \Phi(x) \in F$. Dans sa mise en œuvre, Φ est implicite et on calcule simplement le produit scalaire de deux vecteurs dans F en utilisant une fonction noyau :

$$K(x,y) = (\Phi(x) \cdot \Phi(y)) \tag{1.14}$$

La matrice de dispersion inter-classe S_b et intra-classe S_w sont définies comme suivant :

$$S_b = \sum_{i=1}^c p(\omega_i) (u_i - u) (u_i - u)^T \tag{1.15}$$

$$S_w = \sum_{i=1}^c p(\omega_i) E\{(\Phi(x_i) - u_i)(\Phi(x_i) - u_i)^T | \omega_i\} \tag{1.16}$$

Où $u_i = (\frac{1}{n_i}) \sum_{j=1}^{n_i} \Phi(x_{ij})$ est la moyenne de la classe i , u est la moyenne de toutes les images d'apprentissage et $p(\omega_i)$ est la probabilité a priori. [PHTN10]

Dans la méthode KFA, $w \in F$ se situe dans l'espace de tous les échantillons dans F , on définit les vecteurs de support (SVs) :

$$SVs = \Phi(x_1), \Phi(x_2), \dots, \Phi(x_N) \tag{1.17}$$

$$W = \alpha SVs^T \tag{1.18}$$

Où N est le nombre d'échantillons d'apprentissage, et les matrices noyau sont définies comme suivant :

$$K_w = \sum_{i=1}^c p(\omega_i) E(\eta_j - m_i) (\eta_j - m_i)^T \tag{1.19}$$

$$K_w = \sum_{i=1}^c p(\omega_i) E(m_i - \bar{m}) (m_i - \bar{m})^T \tag{1.20}$$

Où $\eta_j = (k(x_1, x_j), k(x_2, x_j), \dots, k(x_n, x_j))$

$m_i = (\frac{1}{n_i} \sum_{j=1}^{n_i} k(x_1, x_j), \frac{1}{n_i} \sum_{j=1}^{n_i} k(x_2, x_j), \frac{1}{n_i} \sum_{j=1}^{n_i} k(x_1, x_j), \dots, \frac{1}{n_i} \sum_{j=1}^{n_i} k(x_n, x_j))^T$ et \bar{m} est la moyenne de tous η_j . [A.M09]

1.8.6 Analyse en Composantes Indépendantes [B.S12]

PCA est une technique optimale de recherche de représentation réduite qui minimise l'erreur de reconstruction, cependant les vecteurs de base tenant compte de l'erreur de reconstruction peuvent ne pas être optimaux pour coder l'information appropriée de l'image à la classification. L'analyse en composantes indépendantes (ICA) est une généralisation de PCA qui utilise en plus des statistiques d'ordre 2, les statistiques d'ordre plus supérieur, ce qui peut produire une représentation de données plus puissante. Le but d'ICA est de trouver des vecteurs de base (images de base) localisés dans l'espace et qui sont statistiquement indépendants, en réduisant au minimum la dépendance statistique. [B.S12]

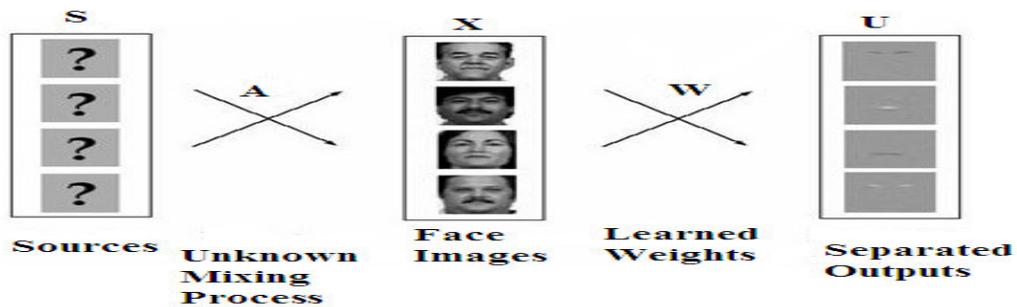


Figure 1.7 Le modèle de synthèse d'image de l'architecture 1 d'ICA [B.S12]

Avant l'apprentissage, on soustrait l'image moyenne des images d'apprentissage X et on les filtre par un filtre de blanchiment (whitening filter) $W_Z = Z(XX^T)^{1/2}$ (1.21)

Pour supprimer les statistiques d'ordre 1 et 2 (la moyenne et les covariances sont mises à zéro et les variances sont égalisées). Pour trouver un ensemble d'images composantes indépendantes, les images d'apprentissage X sont considérées comme étant une combinaison linéaire d'images de base statistiquement indépendantes S, où A est une matrice de mixage inconnue. Les images de base sont retrouvées par une matrice de filtre d'apprentissage $W_i = W_x W_z$ qui produit des sorties U statistiquement indépendantes. Ces images de base sont considérées comme un ensemble de caractéristiques faciales statistiquement indépendantes, où les valeurs des pixels de chaque image caractéristique sont statistiquement indépendantes des valeurs des pixels des autres images caractéristiques [MB,HL&TS98].



Figure 1.8 La représentation d'ICA = (b1, b2, ..., bn) [B.S12]

La représentation d'ICA est constituée des coefficients b de la combinaison linéaire des images de base U qui forment chaque image de visage. Le nombre de composantes indépendantes extraites par l'algorithme d'ICA est égal à la dimension de l'entrée [MB,HL&TS98] (le nombre d'images d'apprentissage), et si on veut contrôler ce nombre, au lieu d'exécuter ICA sur les n images originales, on l'exécute sur un ensemble de m combinaisons linéaires de ces images, où $m < n$. Communément, on choisit pour ces combinaisons linéaires un sous ensemble formé des premières composantes principales de la matrice de donnée, ainsi l'exécution d'ICA sur l'ensemble de ces m composantes principales produit m images sources U_i statistiquement indépendantes. Dans cette première architecture d'ICA, les images de base U_i sont statistiquement indépendantes, mais les coefficients b_i ne le sont pas. Au lieu de séparer les images de visages en ensembles d'images indépendantes, il existe une deuxième architecture d'ICA, dans laquelle les coefficients sont statistiquement indépendants, en séparant les pixels en un ensemble de variables indépendantes codant les images.

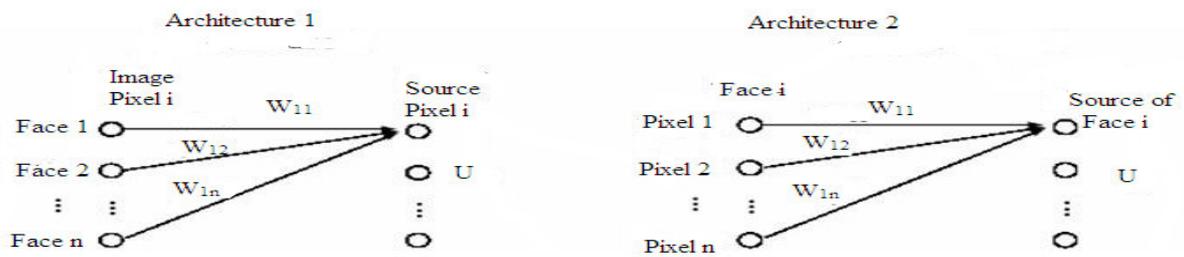


Figure 1.9 Les deux architectures d’ICA : l’architecture 1 pour trouver des images de base statistiquement indépendantes (la séparation des images de visages produit des images composantes indépendantes), et l’architecture 2 pour trouver un code factoriel (la séparation des pixels produit un code factoriel) [B.S12]

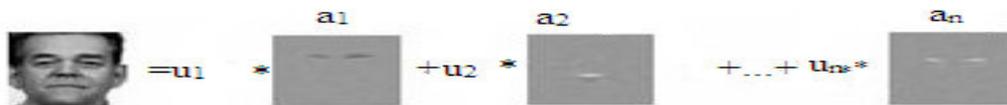


Figure 1.10 Le modèle de synthèse d’image de l’architecture 2 d’ICA [B.S12]

Dans l’architecture 2 d’ICA, chaque image de l’ensemble de données est considérée comme étant créée à partir d’un ensemble d’images de base en A, et d’un vecteur de causes d’images “image causes” fondamentales statistiquement indépendantes en S. L’algorithme d’ICA essaye d’inverser les images de base en trouvant un ensemble de filtres W_i qui produisent des sorties statistiquement indépendantes. [B.S12]

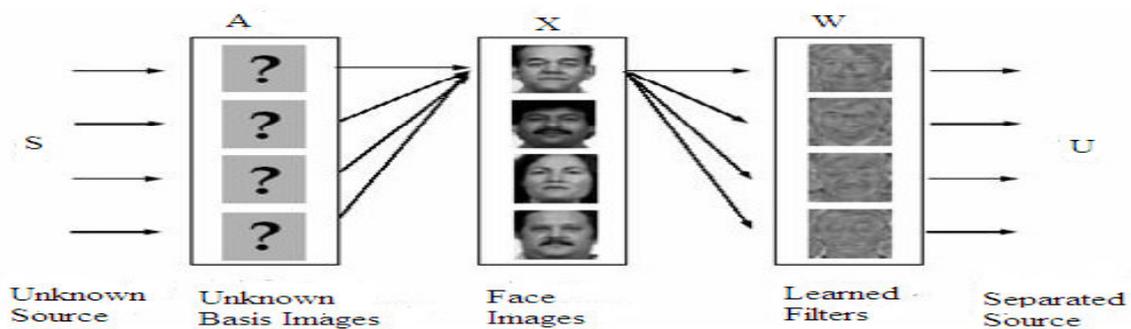


Figure 1.11 La représentation factorielle d’ICA = (u_1, u_2, \dots, u_n) [B.S12]

1.8.7 Méthode de Markov caché [Sit2]

Les HMMs (Hidden Markov Models) sont appliqués à la reconnaissance du visage en considérant l’information du visage comme étant une séquence variable dans le temps.

L'avantage des méthodes locales, est qu'elles prennent en compte la particularité du visage en tant que forme naturelle à reconnaître et un nombre réduit de paramètres en exploitant les résultats de la recherche en neuropsychologie et psychologie cognitive sur le système visuel humain. La difficulté éprouvée c'est quand il s'agit de prendre en considération plusieurs vues du visage ainsi que le manque de précision dans la phase « extraction » des points qui constitue leur inconvénient majeur.

1.8.8 Machine à vecteurs de support

Une machine à vecteurs de support est une technique de discrimination, c'est une méthode d'apprentissage supervisé utilisée pour la classification et la régression. Elle consiste à séparer deux ou plusieurs ensembles de points par un hyperplan. Selon les cas et la configuration des points, la performance de la machine à vecteurs de support peut être supérieure à celle d'un réseau de neurones ou d'un modèle de mixture gaussienne [Sit3].

- 1. Le principe de SVM :** L'idée originale des SVMs est basée sur l'utilisation de fonctions noyau kernel, qui permettent une séparation optimale des points du plan en différentes catégories. La méthode fait appel à un ensemble de données d'apprentissage, qui permet d'établir un hyperplan séparant au mieux les points [Sit3].
- 2. La séparation optimale :** Dans un plan à deux dimensions dans lequel sont répartis deux groupes de points associés à deux classes : les points (+) pour $y > 0$ et les points (-) pour $y < 0$, on peut trouver un séparateur linéaire évident, qui est l'axe des abscisses et qui sépare parfaitement les points. Le problème est dit *linéairement séparable*. Pour des problèmes plus compliqués, la caractérisation d'un séparateur linéaire peut être très compliquée et tout à fait non optimale. Par exemple dans un plan dans lequel les points (+) sont regroupés en un cercle, avec des points (-) tout autour, aucun séparateur linéaire en deux dimensions ne pourra correctement séparer les groupes. Le problème dans ce cas n'est pas *linéairement séparable*. La SVM utilise les *trucs du noyau* "kernel trick" pour appliquer une classification linéaire à un problème de classification non linéaire [Sit3].
- 3. Le concept d'hyperplan :** Afin de remédier au problème de l'absence de séparateur linéaire, le concept des SVMs est de reconsidérer le problème dans un espace de dimension supérieure. Dans ce nouvel espace, il existe un séparateur linéaire qui permet de classer au mieux les points dans les deux groupes qui conviennent, avec une distance maximale entre l'hyperplan et les points les plus proches des deux classes, afin de pouvoir classifier plus exactement les nouveaux points, cette distance s'appelle la *marge*. Cet hyperplan est connu sous le nom de *hyperplan à marge maximum* ou

l'hyperplan optimal et les vecteurs (points de dimension n) les plus proches de cet hyperplan s'appellent les *vecteurs de support*. On pourra ensuite projeter le séparateur linéaire dans l'espace d'origine pour visualiser le résultat de la classification. Le séparateur linéaire obtenu peut être un hyperplan, c'est à dire la généralisation à n dimensions d'une ligne (1D) séparant un espace 2D, ou d'un plan (2D) séparant un espace 3D. Le changement d'espace se fait au moyen d'une fonction répondant au critère de Mercer. Ce critère permet un changement dans les deux sens, ce qui permet à partir de l'expression de l'hyperplan dans l'espace complexe de classer les éléments dans l'espace de description initial [Sit3].

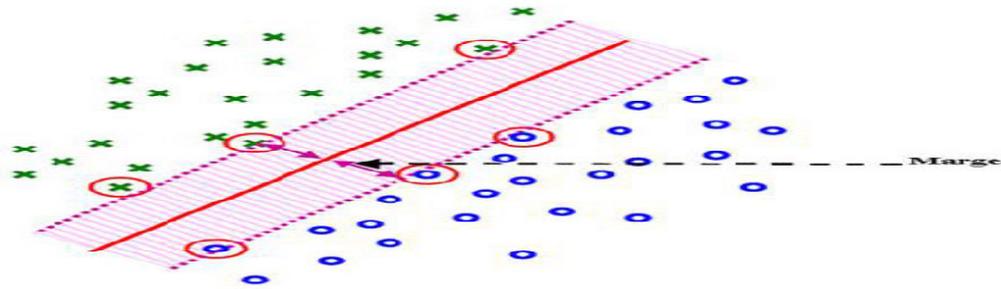


Figure 1.12 Exemple de deux classes linéairement séparables. L'hyperplan déterminé par la SVM, maximisant la marge, permet de séparer les deux classes de manière optimale. [B.S12]

1. La classification linéaire : On considère un ensemble de données $\{(x_1, c_1), (x_2, c_2), \dots, (x_p, c_p)\}$ où c_p est la classe à laquelle appartient le point x_p , c_p est égale à (-1) ou à (+1) et x_p est un vecteur de dimension n de valeurs appartenant à l'intervalle [0,1] ou [-1,1]. Ces *données d'apprentissage* désignent la classification correcte qu'on voudra que la SVM fasse par la suite, au moyen de l'hyperplan de division, qui obéit à la forme :

$$w \cdot x - b = 0 \tag{1.22}$$

On s'intéresse aussi aux hyperplans parallèles à l'hyperplan optimal qui sont tangents aux vecteurs de support des classes, et qui sont décrits par les équations : $w \cdot x - b = 1$ et $w \cdot x - b = -1$. On veut que ces hyperplans maximisent la distance avec l'hyperplan de division et qu'il n'y ait aucun point de données entre les deux. En employant la géométrie, on trouve que la distance entre les hyperplans est $2 / |w|$, ainsi le but est de réduire au minimum $|w|$. Pour exclure les points de données, on doit s'assurer que pour chaque i on a $w \cdot x_i - b \geq 1$ où $w \cdot x_i - b \leq -1$, ou sous une autre équation $c_i(w \cdot x_i - b) \geq 1$ avec $1 \leq i \leq p$. Le problème maintenant est de minimiser $|w|$ sous cette dernière contrainte, ceci est un problème d'optimisation de *programme quadratique QP*). Il existe plusieurs algorithmes spécialisés pour résoudre rapidement les problèmes *QP* et la méthode la plus commune est l'algorithme *SMO*. Après

l'apprentissage, la SVM peut classifier les nouvelles données (données de test) en utilisant la

règle de décision suivante :

$$C = \begin{cases} 1 & \text{if } w \cdot xi + b \geq 0 \\ 1 & \text{if } w \cdot xi + b \leq 0 \end{cases} \tag{1.23}$$

On peut constater d'après la règle de classification qu'elle est seulement en fonction des vecteurs de supports (les données d'apprentissage se trouvant sur la marge) [Sit3].

2. La classification non linéaire : L'algorithme original de l'hyperplan optimal est un classificateur linéaire, et pour avoir un classificateur non linéaire on applique le *kernel trick* à l'hyperplan à marge maximum, l'algorithme résultant est formellement semblable, sauf que chaque produit scalaire est remplacé par une fonction noyau non linéaire.

Parmi les noyaux les plus communs, on trouve :

3. le noyau polynomial (homogène) : $k(x, \bar{x}) = (x \cdot \bar{x})^\alpha$ (1.24)

4. le noyau polynomial (inhomogène) : $k(x, \bar{x}) = (x \cdot \bar{x} + 1)^\alpha$ (1.25)

5. la fonction radiale de base RBF : $k(x, \bar{x}) = e^{-(\gamma \|x - \bar{x}\|^2)}$ avec $\gamma > 0$ (1.26)

6. la fonction radiale de base de type gaussien : $k(x, \bar{x}) = e^{-\gamma \|x - \bar{x}\|^2}$ (1.27)

7. la fonction sigmoïde: $k(x, \bar{x}) = \tanh(kx \cdot \bar{x} + c)$ (1.28)

1.8.9 La décomposition hiérarchique de Tucker (HTD)

Contrairement à la décomposition de Tucker, HTD emploie une hiérarchie des matricizations, motivées par la propriété d'emboîtement suivant.

1. **Lemme 1** [LG10, Lemme 17] : Laisser $x \in \mathbb{C}^{n_1 \times \dots \times n_d}$ et $t = t_1 \cup t_r$ pour $t_1 = \{i_1, i_1+1, \dots, i_m\}$ et $t_r = \{i_m+1, \dots, i_r\}$. Puis la durée $X^{(t)} \subset (X^{(t_1)} \otimes X^{(t_r)})$.

2. **Preuve :** N'importe quelle colonne d' $X^{(t)} = X^{(t_1)} \otimes X^{(t_r)}$ peut être considéré comme la vectorisation d'un tenseur $C \in \mathbb{C}^{n_1 \times \dots \times n_r}$. Les colonnes de la matricization $C^{(t_1)}$ sont clairement contenues dans la durée $(X^{(t)})$ et par conséquent $C^{(t_1)} = X^{(t_1)} (X^{(t_r)})^+ + C^{(t_r)}$.

Où M^+ désigne le pseudo-inverse de Moore-Penrose d'une matrice M. Par analogie $(C^{(t_r)})^T = X^{(t_r)} (X^{(t_1)})^+ + C^{(t_1)}$. Ces deux relations impliquent

$$C^{(t_1)} = \underbrace{(X^{(t_1)} (X^{(t_r)})^+ + C^{(t_r)})}_{=: V} = (X^{(t_1)} (X^{(t_r)})^+ + C^{(t_r)}) \Rightarrow \text{vec}(C) = (X^{(t_1)} \otimes X^{(t_r)}) \text{vec}(V)$$

Compte tenu des bases U_t, U_{t_1}, U_{t_r} pour les espaces de colonne de $X^{(t)}, X^{(t_1)}, X^{(t_r)}$ le résultat de la lemme 1 implique l'existence de ce qu'on appelle matrice de transfert B_t telle que :

$$U_t = (U_{t_r} \otimes U_{t_1}) B_t \tag{1.29}$$

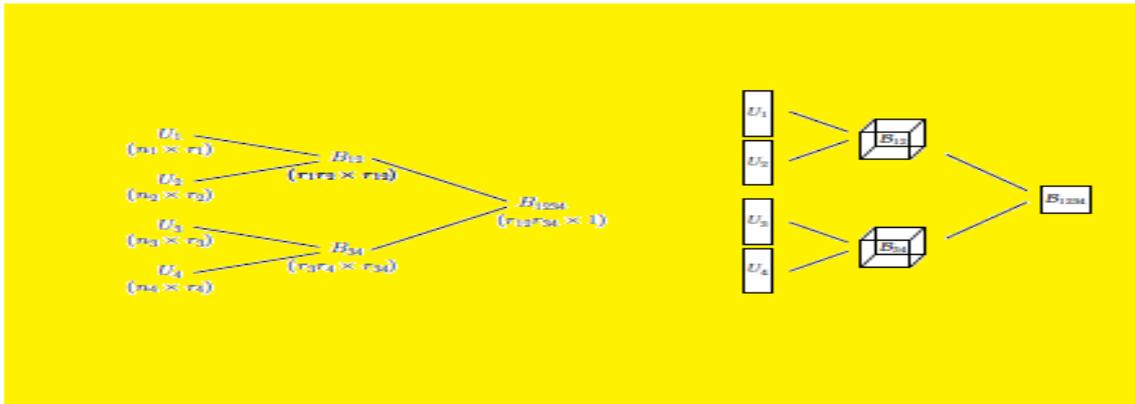


Figure 1.13 Illustration de la HTD (τ) pour $d = 4$. [DK&CT12]

Où r_i, r_{tl}, r_{tr} désigner les rangs des matricizations correspondantes. Appliquant cette relation de manière récursive, jusqu'à ce que t_l et t_r deviennent des singletons, conduit à la HTD

1. Exemple 1

Répété l'application de (1.29) pour $d = 4$: $vec(X) = X^{(1234)} = (U_{34} \otimes U_{12}) B_{1234}$

$$U_{12} = (U_1 \otimes U_2) B_{12}; U_{34} = (U_3 \otimes U_4) B_{34} \Rightarrow vec(X) = (U_3 \otimes U_4 \otimes U_1 \otimes U_2) (B_{34} \otimes B_{12}) B_{1234} \quad (1.30)$$

Il est souvent avantageux de remodeler les matrices de transfert :

$$B_{1234} \in \mathbb{C}^{r_{12} r_{34} \times 1} \Rightarrow \beta_{1234} \in \mathbb{C}^{r_{12} r_{34} \times 1}; B_{12} \in \mathbb{C}^{r_{12} \times r_{12}} \Rightarrow \beta_{12} \in \mathbb{C}^{r_{12} \times r_{12}}; B_{34} \in \mathbb{C}^{r_{34} \times r_{34}} \Rightarrow \beta_{34} \in \mathbb{C}^{r_{34} \times r_{34}}$$

Une illustration de la structure hiérarchique et les données à stocker pour le HTD (τ) est donnée à la figure 1.13. La construction générale d'une HTD nécessite une division hiérarchique des modes $1, \dots, d$.

2. Définition Un arbre binaire T avec chaque nœud représenté par un sous-ensemble du $\{1, \dots, d\}$ est appelé un arbre de dimension si le nœud racine est $\{1, \dots, d\}$, chaque nœud terminal est un singleton, et chaque nœud parent est l'union disjointe de ses deux enfants. Dans ce qui suit, on note : $\mathcal{L}(\tau)$ ensemble de tous les nœuds terminaux ; $\mathcal{N}(\tau)$ ensemble de tous les nœuds non-feuilles, $\mathcal{N}(\tau) = \tau \setminus \mathcal{L}(\tau)$

Remarque Pour plus de commodité de notation, nous imposons l'hypothèse suivante supplémentaire sur les enfants de gauche et de droite t_l, t_r d'un t de nœud dans l'arborescence de la dimension : chaque élément de t_l est plus petit que tout élément de t_r . Note que cette hypothèse peut toujours être satisfaite par une remise en ordre appropriée des modes

Il n'est pas difficile de voir que le nombre de nœuds non-feuilles est toujours $d-1$. Un exemple d'un arbre de dimension est donné à la figure 1.14.

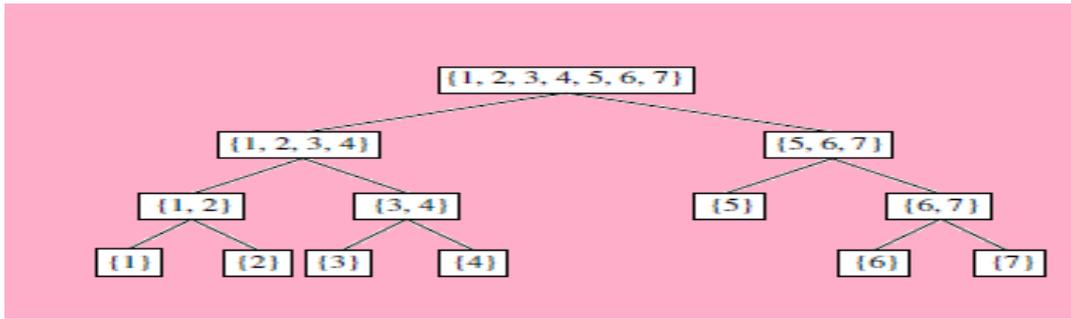


Figure 1.14 Un arbre de dimension pour $d = 7$. [DK&CT12]

Ayant prescrit un k_t de rang maximal pour chaque nœud $t \in \tau$, l'ensemble des tenseurs de Tucker hiérarchiques de rang hiérarchique au maximum (k_t) est défini comme

$H\text{-Tucker}((k_t)_{t \in \tau}) = \{X \in \mathcal{C}^{n_1 \times \dots \times n_d} : \text{rang}(X^{(t)}) \ll k_t \text{ pour tout } t \in \tau\}$. Tel un tenseur Tucker

hiérarchique X est stocké dans le format de Tucker hiérarchique comme suit. À chaque nœud

terminal $\{\mu\}$ une base $U_\mu \in \mathcal{C}^{n_\mu \times r_\mu}$ où $r_\mu := \text{rang}(X^{(\mu)}) \ll k_\mu$ est stocké. À chaque parent nœud t

avec enfants t_l et t_r , le troisième ordre de transfert tenseur $\beta_t \in \mathcal{C}^{r_{t_l} \times r_{t_r} \times r_t}$ satisfaisant (1.29)

avec $B_t = B_t^{(1,2)}$ est stocké. De manière équivalente, (1.29) peut être écrite comme :

$$(U_\mu)_{:,q} = \sum_{i=1}^{k_{t_l}} \sum_{j=1}^{k_{t_r}} ((U_{t_l})_{:,j} \otimes (U_{t_r})_{:,i}) (\beta_t)_{i,j,q}, \quad q=1,2,\dots,r_t \tag{1.31}$$

Le HTD pour X est obtenu de manière récursive insertion (1.29) comme illustré dans

l'exemple 1. En résumé, le format hiérarchique de Tucker est représenté par d des matrices

U_μ et $(d - 1)$ transfert tenseurs B_t . Par conséquent, si $r = \max\{r_t : t \in \tau\}$ et $n = \max\{n_1, \dots, n_d\}$ les

conditions de stockage sont limitées par : $dnr + (d-2)r^3 + r^2$. Où nous avons utilisé que le

tenseur de transfert à la racine peut effectivement être considéré comme une matrice de taille

au plus $r \times r$.

Conclusion

La technique linéaire la plus connue et sans aucun doute l'analyse en composantes principales (PCA), également appelée transformée de Karhunen-Loeve. Le PCA fut d'abord utilisé afin de représenter efficacement des images de visages humains [L.S&M.K87]. En 1991, cette technique a été reprise dans le cadre plus spécifique de la reconnaissance faciale sous le nom de méthode des Eigenfaces [MT&AP91].

Il existe d'autres techniques également construites à partir de décompositions linéaires comme l'analyse discriminante linéaire (LDA) [PB,JH&DK97] ou encore l'analyse en composantes indépendantes (ICA) [MB,JM&TS02]. Et les machines à vecteurs de support (SVM) [GG,SL&KC00], cette dernière méthode ont l'avantage de pouvoir modéliser plus facilement les variations de pose, d'éclairage et d'expression par rapport aux méthodes

globales. Toutefois, elle est plus lourdes à utiliser puisqu'il faut souvent placer manuellement un assez grand nombre de points sur le visage alors que les méthodes globales ne nécessitent de connaître que la position des yeux afin de normaliser les images, ce qui peut être fait automatiquement et de manière assez fiable par un algorithme de détection [SA,PC&RL06].

Grâce à notre étude, nous constatons que toutes ces méthodes de réduction de données restent valable à la reconnaissance de visage 2D et 3D et leurs applications reste limitées à des bases de données de tailles réduites à ceci s'ajoute l'inconvénient d'analyser l'image ou dans le domaine spatial ou spectral séparément. Ce qui motive notre envie d'aller chercher dans l'analyse multilinéaire et multidimensionnel en nous intéressant à des réductions de données adaptées entre autre la réduction de Tucker qui est proche de la PCA optimale et fondamentale dans les SRV2D 3D. Nous développerons notre étude sur la réduction de Tucker ainsi que son application à la reconnaissance de visage dans les chapitres suivants. Un état de l'art approprié est présenté dans l'étude le deuxième chapitre.

Chapitre 2

Etat de l'art sur la réduction de l'écoulement de Tucker dans la SIRE

Introduction

L'émergence des images hyper-spectrales (IHS) implique l'exploration et la collecte d'une grande quantité de données. Capteurs d'imagerie hyper-spectrale fournissent un grand nombre de bandes spectrales, généralement jusqu'à plusieurs centaines. Cette déraisonnablement grande dimension d'IHS non seulement augmente la complexité du calcul, mais aussi dégrade la précision de la classification. Parce qu'IHS contient une grande quantité de redondance spectrale et manque de données sur la formation, la réduction de la dimension spectrale s'est avérée nécessaire d'appliquer des algorithmes de classification. Un travail antérieur [NEzbaed 08] a montré que les espaces de données de grande dimension sont presque vides, ce qui indique que la structure de données impliqués existe principalement dans un sous-espace. En raison de sa simplicité et sa facilité d'utilisation, l'approche de réduction la plus populaire de la dimensionnalité (DR) est l'APC, Dénommé PCADR, ce qui maximise la variance des données par projection orthogonale. Un raffinement de PCADR est l'analyse en composantes indépendantes (ICA), dénommé ICADR qui utilise les statistiques d'ordre supérieur. Mais ces deux méthodes de DR linéaires exigent un accord préalable des données. En effet, lorsqu'il s'agit de données à trois voies une première étape consiste à vectoriser les images rendement de données bidirectionnelles, permettant l'utilisation du traitement du signal, mais en négligeant le réarrangement spatial. Pour surmonter cela, on propose une méthode mathématique basée multi-canalopérateur morphologique DR qui incorpore la représentation d'image. Une méthode de DR multilinéaire basée sur l'algèbre est proposée pour extraire les composantes principales spectrales en tenant compte de l'information spatiale. En outre, certains bruits radiométriques dégradent les résultats de la classification. Ce bruit est dû à un capteur, les effets de photons et de l'erreur d'étalonnage. Ensuite, en plus de DR, une réduction de bruit (NR) de traitement est également nécessaire pour diminuer la variabilité spectrale, ce qui est utile pour d'autres algorithmes de classification. Dans ce contexte, la fraction bruit maximal, Wiener et ondelettes seuil, le retrait sont des techniques de débruitage utiles. Généralement, le système hybride consiste à la décorrélation de la bande spectrale par l'APC, Fourier ou transformée en cosinus discrète à appliquer 2D - filtrage sur chaque bande décorrélée. La représentation intuitive d'une collection d'images est un tableau à trois voies, ou troisième ordre tenseur. Ainsi, au lieu d'adapter les données à des techniques à base de matrices classiques, la représentation 3D a prouvé son utilité pour les applications de compression. L& méthode d'algèbre multilinéaire réduit simultanément la dimension spectrale et débruite tous les moyens en commun. Sur la base de la décomposition Tucker 3, on

montre la capacité du rang inférieur (K1, K2, K3) tenseur rapprochement comme un outil de débruitage [NEzbaed 08].

2.1 Représentation des données hyperspectrales

Le cube hyperspectral ainsi acquis contient une source d'information considérable. Alors que la visualisation d'une image couleur, acquise suivant trois bandes spectrales² (rouge, vert, bleu), est facilement réalisable, il n'en est pas de même pour l'image hyperspectrale. La représentation de telles données nécessite donc de chercher d'autres espaces de représentation qui sauront tirer parti de l'information la plus appropriée en fonction de l'application visée. L'espace cubique peut être représenté par une image volumique, mais à part pour des applications de compression cet espace n'a pas été valorisé. Les espaces couramment utilisés sont ici présentés [NR08]

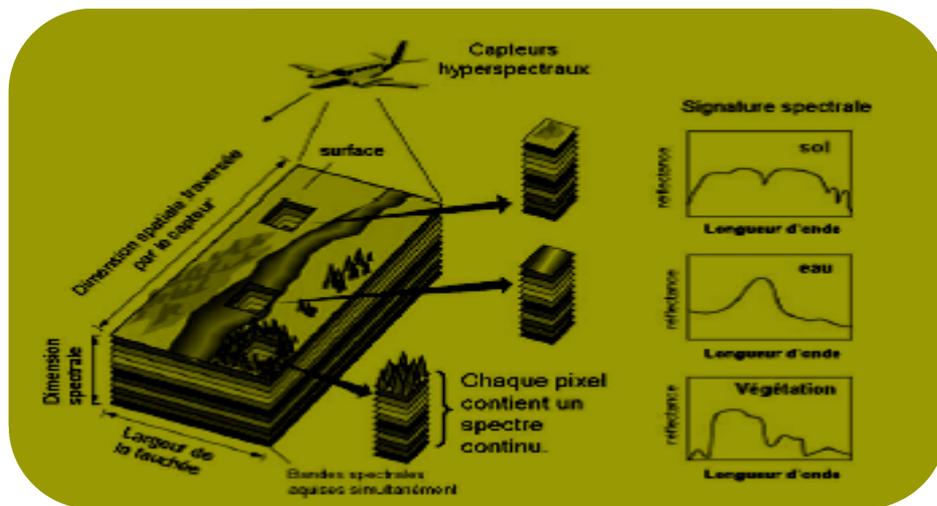


Figure 2.1 Illustration du cube hyperspectral.

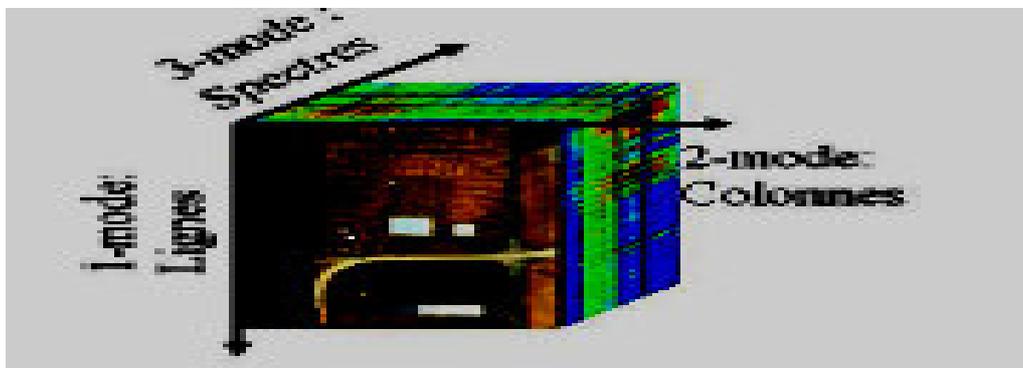


Figure 2.2 Illustration de la représentation tensorielle des données hyperspectrales de HSI02_scene1b.

2.1.1 Espace spatial

L'espace spatial (ou espace image [M.L&L.H02]) des données hyperspectrales est valorisé en considérant le cube hyperspectral³ (3-voir la figure 2.1) comme étant une concaténation de I_3 images de dimension $I_1 \times I_2$. Une telle représentation est illustrée à la figure 2.3. On note la matrice $\mathbf{R}(:, :, i_3) = \{r_{i_1 i_2} : 1 \leq i_1 \leq I_1, 1 \leq i_2 \leq I_2\}$ la $i_3^{\text{ième}}$ image spectrale, avec $i_3=1, \dots, I_3$. Chacune des images obtenues est une image quasi-monochromatique qui contient l'information associée à une longueur d'onde donnée.

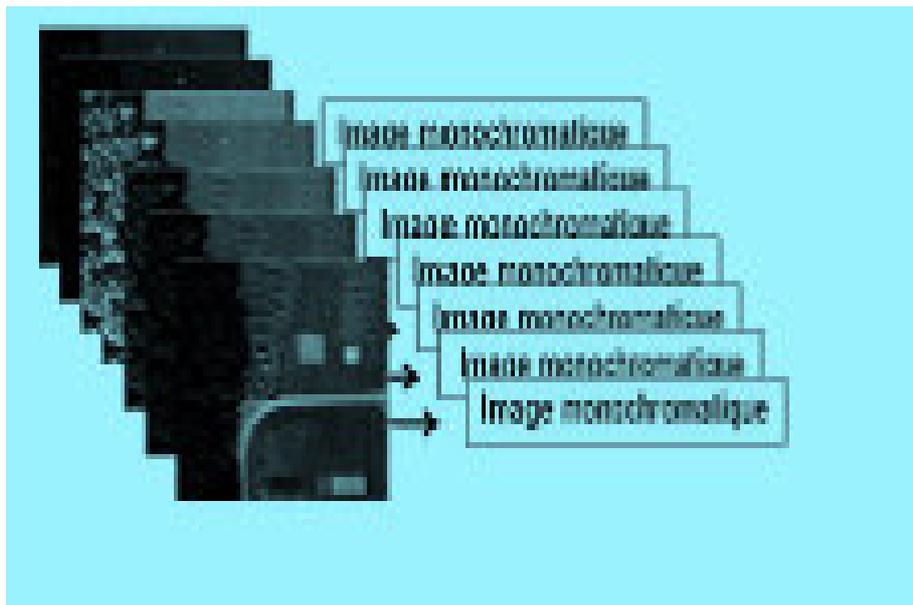


Figure 2.3 Représentation spatiale : collection d'images quasi monochromatiques.

2.1.2 Espace spectral

L'espace spectral des données hyperspectrales est valorisé en considérant le cube hyperspectral comme étant une concaténation de $I_1 \cdot I_2$ spectres. Une telle représentation est illustrée à la figure 2.4. Ainsi chaque vecteur pixel est une réalisation d'un vecteur de variable aléatoire $\mathbf{r} = (r_1, r_2, \dots, r_{I_3})^T$ avec I_3 le nombre de bandes spectrales

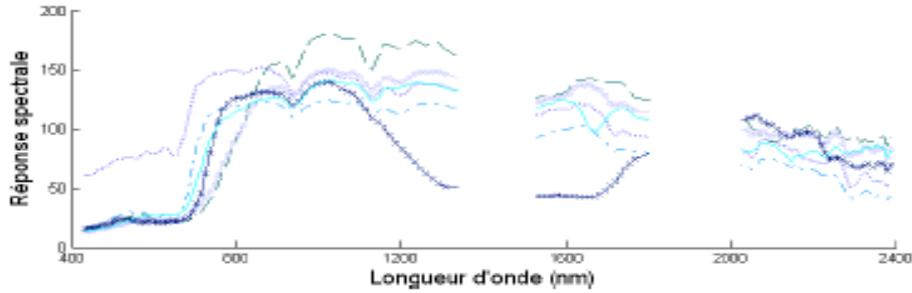


Figure 2.4 Représentation spectrale : collection de spectres.

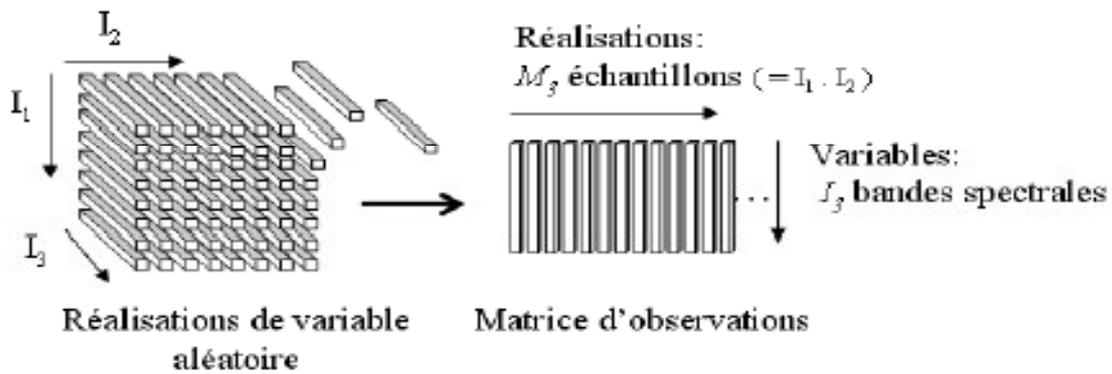


Figure 2.5 Représentation matricielle des données hyperspectrales.

Nous notons $M_3 = I_1 \cdot I_2$ le nombre d'échantillons présents dans la scène. Il est alors possible de représenter les données sous forme d'une matrice en concaténant l'ensemble des M_3 spectres. La matrice résultante est schématisée à la figure 2.5, et est notée :

$$\mathbf{R} = [\mathbf{r}_1 \mathbf{r}_2 \dots \mathbf{r}_{M_3}] \quad (2.1)$$

Où $\mathbf{r}_m = (r_1, r_2 \dots r_B)^T$, $\forall m = 1, \dots, M_3$, est une réalisation d'une variable aléatoire dont on peut estimer :

- la moyenne :
$$\hat{\boldsymbol{\mu}} = \frac{1}{M_3} \sum_{m=1}^{M_3} \mathbf{r}_m \quad (2.2)$$

- la variance :
$$\hat{\boldsymbol{\theta}}^2 = \frac{1}{M_3} \sum_{m=1}^{M_3} (\mathbf{r}_m - \hat{\boldsymbol{\mu}})^2 \quad (2.3)$$
 la matrice de

covariance :
$$\hat{\boldsymbol{\tau}}^2 = \frac{1}{M_3} \sum_{m=1}^{M_3} (\mathbf{r}_m - \hat{\boldsymbol{\mu}})(\mathbf{r}_m - \hat{\boldsymbol{\mu}})^T \quad (2.4)$$

2.2 Définition et propriétés mathématiques d'un tenseur

2.2.1 Définition d'un tenseur

Dans tout ce manuscrit, les données étudiées sont supposées appartenir au corps des réels \mathbb{R} . La définition d'un tenseur est explicitée en détails par DE SILVA *et al*

[V.S&L.L08] qui rappelle que ce dernier ne doit pas être confondu avec le tenseur utilisé en physique et en ingénierie (tel que le tenseur des contraintes¹ (1. *stress tensor*)) qui sont souvent désignés par le tenseur champ² (2. *fieldtensor*) en mathématique. Nous considérons qu'un tenseur d'ordre N représente un tableau multidimensionnel dont chaque élément est accessible via N indices (i_1, \dots, i_N) . Ce tenseur est noté par :

$$A \in \mathbb{R}^{I_1 \times \dots \times I_N} \tag{2.5}$$

et chacun de ses éléments est noté $a_{i_1 \dots i_N}$. Chaque dimension du tenseur est appelée « n mode » en référence au $n^{\text{ième}}$ indice du tenseur. Un tenseur d'ordre zéro est un scalaire, un tenseur d'ordre un est un vecteur et un tenseur d'ordre deux est une matrice.

1. ORDRE DEUX

Une matrice réelle $A \in \mathbb{R}^{I_1 \times I_2}$ est la représentation, sous la forme d'un tableau de valeurs, d'une forme bilinéaire φ construite sur deux espaces vectoriels $E^{(1)}$ et $E^{(2)}$, de dimensions respectives I_1 et I_2 [J.L.F98, P.L83] :

$$\begin{aligned} \varphi : E^{(1)} \times E^{(2)} &\rightarrow \mathbb{R} \\ (e^{(1)}_{i_1}, e^{(2)}_{i_2}) &\rightarrow \varphi(e^{(1)}_{i_1}, e^{(2)}_{i_2}) = a_{i_1 i_2}, \end{aligned}$$

dans laquelle $a_{i_1 i_2}$ est l'élément de la ligne i_1 et de la colonne i_2 de la matrice A . À un tenseur d'ordre deux sont donc associés deux espaces vectoriels correspondant à chacun des n -modes (lignes, colonnes). Leur dimension respective (I_1 et I_2) est égale au nombre d'éléments du n -mode auquel ils sont associés.

2. ORDRE N (>2)

Un tenseur réel $A \in \mathbb{R}^{I_1 \times \dots \times I_N}$ est la représentation sous la forme d'un tableau multidimensionnel de valeurs, d'une forme multilinéaire φ construite sur N espaces vectoriels $E^{(1)}, \dots, E^{(N)}$ de dimension respectives I_1, \dots, I_N [J.L.F98, P.L83] :

$$\begin{aligned} \varphi : E^{(1)} \times \dots \times E^{(N)} &\rightarrow \mathbb{R} \\ (e^{(1)}_{i_1}, \dots, e^{(N)}_{i_N}) &\rightarrow \varphi(e^{(1)}_{i_1}, \dots, e^{(N)}_{i_N}) = a_{i_1 \dots i_N}, \end{aligned}$$

dans lequel $a_{i_1 \dots i_N}$ est l'élément du tenseur A situé aux indices n -modaux $(i_1 \dots i_N)$. Il est donc possible d'associer à un tenseur d'ordre N , N espaces vectoriels $E^{(1)}, \dots, E^{(N)}$. Dans la suite, quel que soit $n \in \{1, \dots, N\}$, l'espace $E(n)$ est dénommé « espace vectoriel n -modal ». Dans le cas d'un tenseur réel $A \in \mathbb{R}^{I_1 \times \dots \times I_N}$, $E^{(n)} = \mathbb{R}^{I_n}, \forall n \in \{1, \dots, N\}$.

Les espaces vectoriels n -modaux ainsi définis, sont largement utilisés dans la suite de notre étude. Ils constituent la base conceptuelle sur laquelle reposent tous les traitements proposés pour les données multidimensionnelles.

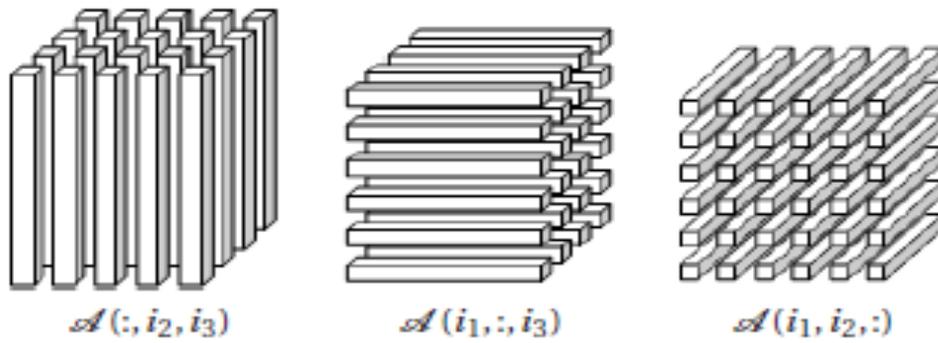


Figure 2.6 Fibres d'un tenseur d'ordre 3 [B.W.B&T.G.K06].

Dans la section suivante, nous présentons les principaux opérateurs tensoriels que nous sommes amenés à utiliser tout le long de ce manuscrit.

2.3 Décomposition de TUCKER3

La décomposition de TUCKER3 a été pour la première fois introduite par TUCKER en 1963 [L.T63]. Un affinement est apporté dans [L.T64, L.T66]. Plusieurs noms ont été attribués dans la littérature pour désigner cette décomposition : (i) *Three-mode factor analysis*—3MFA/Tucker3 [L.T66], (ii) *Three-mode principal components analysis*—3MPCA [P.K&J.L80], (iii) *N-mode principal components analysis* [A.K&H.N86], ou encore (iv) *Higher-order SVD*—HOSVD [L.L&BM00].

Le modèle de TUCKER3 a été initialement élaboré pour effectuer une décomposition en valeurs singulières sur tous les modes du tenseur indépendamment les uns des autres. Cette décomposition est désignée dans la suite de ce manuscrit, HOSVD ⁹(*9-HigherOrder Singular Value Decomposition*) [L.L&BM00]. Mais, ce modèle est aussi à l'origine de la solution d'approximation en rang- (K_1, \dots, K_N) inférieur d'un tenseur d'ordre N ¹⁰(*10-Lowerrank- (K_1, \dots, K_N) tensor approximation*), notée LR $TA-(K_1, \dots, K_N)$ [L.L00]. KROONENBERG et DE LEEUW ont proposé, en 1980, un algorithme des moindres carrés alternés ¹¹(*11-Alternating Least Square*) (ALS) pour résoudre la minimisation de l'erreur d'approximation.

2.3.1 Le modèle de TUCKER3

La décomposition de TUCKER3 est une forme d'analyse en composantes principales. Elle factorise un tenseur A d'ordre N en un tenseur noyau ¹²(*12-core tensor*) transformé (ou

multiplié) par une matrice sur chacun de ses N modes. Cette décomposition d'un tenseur A d'ordre N s'exprime alors :

$$A = \sum_{\times 1} U^{(1)} \sum_{\times 2} U^{(2)} \sum_{\times 3} \dots \sum_{\times N} U^{(N)}, \quad (2.6)$$

Où $U^{(n)} \in \mathbb{R}^{I_n \times K_n}$ est une matrice orthogonale telle que :

$$U^{(n)T} U^{(n)} = I_{K_n \times K_n}$$

Les matrices $U^{(n)} \forall n = 1, \dots, N$ sont aussi appelées les matrices facteurs et peuvent être assimilées aux composantes principales de chaque n -mode. Et, $\Sigma \in \mathbb{R}^{K_1 \times \dots \times K_N}$ est le tenseur noyau dont chaque indice indique le niveau d'interaction entre les différents composants.

La valeur de $K_n = 1, \dots, I_n$, désigne le nombre de composantes (soit le nombre de colonnes) des matrices orthogonales $U^{(n)}$ qui sont conservées, $\forall n = 1, \dots, N$. Autrement écrit, le modèle s'exprime par

$$\sum_{i=1}^{K_1} \sum_{i=2}^{K_2} \dots \sum_{i=N}^{K_N} \theta_{i1i2\dots iN} u_{i1}^{(1)} u_{i2}^{(2)} \dots u_{iN}^{(N)}$$

2.3.2 HOSVD

La HOSVD permet de déterminer les matrices orthogonales $U^{(n)}$ en effectuant la SVD sur chaque mode du tenseur indépendamment les uns des autres. Les matrices $U^{(n)}$ sont donc les vecteurs propres de la matrice de covariance de la dépliante dans le n -mode de A que nous notons $E[A_n A_n^T]$. Les deux étapes nécessaires à la détermination de la décomposition

de TUCKER3 sont donc :

- 1- $\forall n = 1, \dots, N$,
 - a- À Déploiement de $A \rightarrow A_n$
 - b- À Calcul de la SVD de $A_n = U^{(n)} \Sigma V^{(n)} \rightarrow U^{(n)}$
- 2- Calcul du tenseur noyau $\rightarrow \Sigma = A_{\times 1} U^{(1)T} \times_2 \dots \times_N U^{(N)T}$

La troncature de la HOSVD revient à ne conserver que les K_n premiers vecteurs propres associés aux K_n valeurs propres de la matrice de covariance n -modale (i.e. les K_n premières colonnes de la matrice $U^{(n)}$).

3. Comparaison entre la SVD et la HOSVD

Dans le cas d'une matrice $A \in \mathbb{R}^{I_1 \times I_2}$, les vecteurs singuliers gauches et droites sont assimilés aux vecteurs singuliers 1 et 2-modaux respectivement. Par conséquent, la SVD d'une matrice A peut s'écrire en écriture tensorielle telle que $A = U \Sigma V^T = \sum_{\times 1} U_{\times 2} V$.

Où Σ est la matrice diagonale, nommée matrice noyau, contenant les valeurs singulières et, où $U \in \mathbb{R}^{I_1 \times I_1}$ et $V \in \mathbb{R}^{I_2 \times I_2}$ sont les matrices unitaires contenant les vecteurs singuliers gauches et droites respectivement de la matrice A .

Ainsi, il y a une correspondance entre la matrice A et les dépliante dans le 1 et 2-mode du tenseur A d'ordre 2 $A_1 = A$ et $A_2 = A^T$.

Même si la HOSVD semble très similaire avec la SVD, il existe des distinctions. Ces différences concernent essentiellement la matrice noyau Σ impliqué dans la SVD et le tenseur noyau Σ impliqué dans la HOSVD. Deux conditions (C_1 et C_2) sont mises en cause [L.L&BM00] :

	SVD	HOSVD
C_1	Pseudo-diagonalité	Toute orthogonalité
C_2	Ordonnancement des valeurs singulières de la matrice noyau Σ	Ordonnancement des sous-tenseurs d'ordre $N-1$ du tenseur noyau Σ

C_1 - Toute orthogonalité : $\forall n = 1, \dots, N, \forall \alpha \neq \beta \in \{1, \dots, I_n\}$ tels que $\alpha \neq \beta$, les sous-tenseurs $\Sigma_{in = \alpha}$ et $\Sigma_{in = \beta}$ d'ordre $N-1$ sont orthogonaux : $\langle \Sigma_{in = \alpha} | \Sigma_{in = \beta} \rangle = 0$.

C_2 - Ordonnancement : $\forall n = 1, \dots, N, \|\Sigma_{in = 1}\| \geq \|\Sigma_{in = 2}\| \geq \dots \|\Sigma_{in = I_n}\|$.

Dans ces deux propriétés, le sous-tenseur $\Sigma_{in = \alpha}$ d'ordre $N-1$, issu du tenseur Σ , est obtenu en fixant le $n^{\text{ième}}$ indice à la valeur α . Pour tous les autres indices, chaque élément du tenseur $\Sigma_{in = \alpha}$ est défini de telle sorte que $(\Sigma_{in = \alpha})_{i_1 \dots i_{n-1} i_{n+1} \dots i_N} = S_{i_1 \dots i_{n-1} i_{n+1} \dots i_N} \alpha_{i_n}$, $S_{i_1 \dots i_N}$ étant l'élément d'indice (i_1, \dots, i_N) du tenseur Σ .

2.3.3 LRTA- (K_1, \dots, K_N)

Étant donné un tenseur $A \in \mathbb{R}^{I_1 \times \dots \times I_N}$ d'ordre N , le problème d'approximation en rang- K_n inférieur ($\forall n = 1, \dots, N, K_n < I_n$) de A consiste à trouver le tenseur B de rang- (K_1, \dots, K_N) telle que la norme de Frobenius (relation 2.7), entre ces deux tenseurs, soit minimale. Nous devons donc résoudre le problème :

$$\|A\|_F^2 = \sum_{i_1, \dots, i_N} \alpha_{i_1, \dots, i_N}^2 = \text{trace} [A_n A_n^T], \forall n = 1, \dots, N \quad (2.7)$$

$$\min_{\text{rang}-(k_1, \dots, k_N)} \|A - B\|_F^2 \quad (2.8)$$

Ce problème est traité dans [L.L00]. Dans le cas matriciel, ce problème est simplement résolu en utilisant la troncature de la SVD¹³ (13-Propriété de ECKART-YOUNG [Y.E&G.Y36]). La HOSVD ordonnant les « masses » de la même façon que la SVD, on

pourrait penser que cette décomposition résout la relation 2.8. Mais la généralisation de ce théorème aux tenseurs n'est plus vérifiée [L.L00].

Plusieurs interrogations théoriques, concernant l'approximation de rang- (K_1, \dots, K_N) inférieur d'un tenseur, ont été analysées dans [V.S&L.L08]. En particulier, les démonstrations de l'existence d'une approximation B de A optimale au sens des moindres carrés pour toutes valeurs des rangs (K_1, \dots, K_N) . D'après la relation 2.6, la décomposition tensorielle de TUCKER3 de B s'exprime par

$$B = D \times_1 U^{(1)} \times_2 U^{(2)} \dots \times_N U^{(N)} \quad (2.9)$$

Où $U^{(n)}$ est une matrice orthonormale de dimension $(I_n \times K_n), \forall n=1, \dots, N$, et $D \in \mathbb{R}^{K_1 \times \dots \times K_N}$ est le tenseur noyau. La solution des moindres carrés [109] implique que :

$$D = A \times_1 U^{(1)T} \times_2 U^{(2)T} \dots \times_N U^{(N)T} \quad (2.10)$$

$$(A \times_n U) \times_n V = A \times_n (VU). \quad (2.11)$$

$$(A \times_n U) \times_m V = (A \times_m V) \times_n U. \quad (2.12)$$

Par conséquent, en introduisant l'expression 2.10 dans 2.9, et en utilisant les propriétés 2.11 et 2.12 du n -mode produit, l'approximation de rang- (K_1, \dots, K_N) inférieur optimale du tenseur A s'exprime par :

$$B = A \times_1 P^{(1)} \dots \times_N P^{(N)}, \quad (2.13)$$

dans laquelle, $\forall n=1, \dots, N$, $P^{(n)}$ est le projecteur orthonormal sur le sous-espace $E^{(n)}_1$ de l'espace vectoriel n -modal $E^{(n)}$. $E^{(n)}_1$ est de dimension K_n et est engendré par les vecteurs colonnes de la matrice $U^{(n)}$. Le projecteur $P^{(n)}$ qui satisfait à la minimisation de l'expression 2.8 est donné par :

$$P^{(n)} = U^{(n)} U^{(n)T} \quad (2.14)$$

Rechercher l'approximation de rang- (K_1, \dots, K_N) inférieur revient donc à déterminer les matrices $U^{(n)}$ qui définissent les projecteurs $P^{(n)}$. Dans ce but, il est montré que minimiser (2.8) par rapport à B équivaut à maximiser par rapport aux matrices $U^{(1)}, \dots, U^{(N)}$ la fonction quadratique non linéaire :

$$g = (U^{(1)}, \dots, U^{(N)}) = \|D\|^2 \quad (2.15)$$

Il s'agit d'un problème d'optimisation des moindres carrés qui peut être résolu à l'aide de l'algorithme des moindres carrés alternés [L.L&BM00]. En effet, les matrices $U^{(n)}$, qui maximisent la relation 2.15, doivent être déterminées conjointement. Ce processus est décrit à l'annexe A, où les calculs relatifs à l'estimation conjointe des matrices orthogonales $U^{(n)}$ sont présentés, ainsi que les détails de l'algorithme ALS. De plus, l'annexe A illustre l'évolution de certains paramètres itérativement jusqu'à observer la convergence de l'algorithme.

En 1980, KROONENBERG et DE LEEUW [P.K&J.L80] nomment cet algorithme TUCKALS3, et démontrent l'existence et la nature de la convergence de ce dernier vers le maximum (global) de 2.16. DE LATHAUWER, DE MOOR et VANDEWALLE [L.L00] le désignent par l'algorithme *Higher-Order Orthogonal Iteration*—HOOI. ANDERSSON et BRO

[BRA00] proposent d'initialiser l'algorithme en utilisant la HOSVD, ils montrent que la convergence de ce dernier est plus rapide. Les simulations, présentées à l'annexe A, confirment cette dernière observation. $g(\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}) = \|\mathbf{D}\|^2$ (2.16)

2.3.4 Applications du modèle de TUCKER3

La décomposition de TUCKER3 trouve des applications dans des domaines très variés. Nous pouvons citer de manière non exhaustive les applications suivantes.

Historiquement, la décomposition de TUCKER3 trouve sa première utilité en analyse chimique dont HERION est l'initiateur. De nombreux exemples d'applications en psychométrie ont été fournis par KIERS et VAN MECHELEN.

DE LATHAUWER *et al.* ont utilisé la décomposition de TUCKER3 en traitement du signal. MUTI et BOURENNANE l'ont appliqué pour étendre le filtre de Wiener pour le débruitage d'images couleurs et des signaux sismiques.

VASILESCU et TERZOPOULOS sont les premiers à l'utiliser en vision par ordinateur pour la reconnaissance de visage. Ils considèrent plusieurs utilisateurs avec différentes postures et selon plusieurs illuminations. Cette technique est utilisée pour modéliser les expressions du visage. Enfin, la décomposition de TUCKER3 trouve un intérêt pour l'extraction de connaissance, où SAVAS et ELDÉN l'exploitent pour la reconnaissance de l'écriture manuscrite.

2.4 Autres décompositions tensorielles

De nombreuses décompositions tensorielles sont présentes dans la littérature. Elles divergent en fonction des données pour lesquelles elles sont prédestinées.

2.4.1 PARAFAC/CANDECOMP, ParaCand

Comme pour le modèle de TUCKER3, la décomposition de PARAFAC/CANDECOMP est désignée par plusieurs noms dans la littérature. On la retrouve en 1927 sous le nom de *Polyadic Form of a Tensor*, instauré par HITCHCOCK. Puis c'est en 1944 que CATTELL décrit le principe de *Parallel proportional analysis* et le concept d'axes multiples pour l'analyse, autrement dit PARAFAC qui est l'abréviation de *PARAllelFACTOR analysis*. Cette idée est devenue populaire en 1970, en psychométrie grâce à HARSHMAN et à CARROLL ET CHANG. Cette décomposition est alors répandue sous le nom de CANDECOMP. Dans la suite, nous référençons cette décomposition

PARAFAC/CANDECOMP. De nombreux tutoriaux sur ledéveloppement du modèle de PARAFAC/CANDECOMP (ParaCand) sont consultables.[NR08]

Alors que le modèle de TUCKER3 généralise la décomposition matricielle ¹⁴ (14-définie à la relation 2.13) de la SVD, ladécomposition de ParaCand généralise cette représentation sous sa forme canonique ¹⁵(15-définie à la relation 2.14).

2.4.2 Les décompositions dérivées de PARAFAC/CANDECOMP

Nous introduisons ici une liste non-exhaustive des évolutions de la décomposition deParaCand, d'autres compléments pourront être trouvés dans [T.G.K07].

- **INDISCAL** :L'abréviation de INDISCAL signifie IndividualDifferences inScaling et, est un cas particulier de la décomposition de ParaCand d'un tenseur d'ordretrois dont deux de ses modes sont symétriques. Dans ce cas, les deux premiers facteurs($U^{(1)}$ et $U^{(2)}$) sont identiques. L'algorithme ParaCand est utilisé, les matrices $U^{(1)}$ et $U^{(2)}$ sonttraitées comme des facteurs distincts jusqu'à la convergence où $U^{(1)} = U^{(2)}$.
- **INDORT** : ORThogonallyconstrained INDSCAL est une variante de la méthodeINDISCAL qui contraint l'orthogonalisation des facteurs.

Tableau 2.1 Comparaison entre différents les décompositions tensorielles[NR08]

	Définition de rang	Particularités	Décomposition
TUCKER3	rang multilinéaire, K_n	$U^{(n)}$ orthogonale, Σ non-diagonale	$\sum_{i_1=i_2=i_3=1}^{K_1, K_2, K_3} u_{i_1}^{(1)} \circ u_{i_2}^{(2)} \circ u_{i_3}^{(3)}$
ParaCand	rang tensoriel, K	Σ diagonale, Unicité, $U^{(n)}$ non-orthogonale	$\sum_{k=1}^K \sigma_k u_k^{(1)} \circ u_k^{(2)} \circ u_k^{(3)}$
PARAFAC2	rang tensoriel	Adaptée à des données dont une dimension est variable	$\mathcal{A}^{(1)}(:, :, i_3) \Sigma(:, :, i_3) U^{(2)}$
PARATUCK2	rang 1, 2-mode, P et R	Réduction sur 2 modes, unicité, flexibilité	$U^{(1)} D^{(1)}(:, :, i_3) H D^{(2)}(:, :, i_3) U^{(2)}$

PARAFAC2 :PARAFAC2 est une variante de la décomposition de ParaCand. Cettedécomposition s'applique sur une collection de i_3 matrices $A(:, :, i_3)$. Ces i_3 matrices ont

l'aparticularité d'avoir des tailles différentes dans une dimension. Ce modèle est construit à partir des matrices de variances-covariances des tranches frontales ¹⁹. (19-Voir figure 2.7)

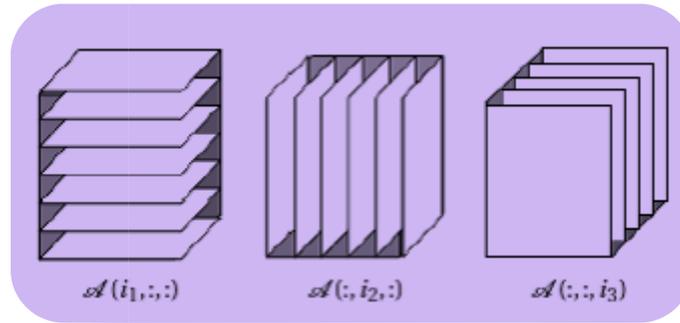


Figure 2.7 Tranches d'un tenseur d'ordre 3 [B.W.B&T.G.K06].

PARATUCK2 : Le modèle PARATUCK2 est la généralisation du modèle de PARAFAC et de TUCKER2. Ce modèle garde à la fois les propriétés d'unicité de ParaCand et, la flexibilité de TUCKER2. La réduction se fait sur deux modes (comme PARAFAC2 et TUCKER2).

2.5 Etat de l'art d'application de tenseurs et réduction de Tucker

2.5.1 Calculs de MATLAB efficaces avec des tenseurs rares et pondérés

Brett W. Bader en 2006 ont appliques la Calculs de MATLAB efficaces avec des tenseurs clairsemés et pondérées cette étude se base sur le principe suivant :

2.5.1.1 Tenseurs de Tucker [BW.B06]

Considérons un tenseur $\mathbf{x} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ telle que :

$$\mathbf{x} = \llbracket g; U(1), U(2); \dots; U(N) \rrbracket = g \times_1 U^{(1)} \times_2 U^{(2)} \dots \times_N U^{(N)} \quad (2.17)$$

Où $g \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_N}$ est le tenseur de base et $U^{(n)} \in \mathbb{R}^{I_n \times J_n}$ pour $n = 1, \dots, N$. C'est le format qui résulte d'une décomposition de Tucker et s'appelle donc un tenseur de Tucker. Nous utilisons la notation $\llbracket g; U(1), U(2); \dots; U(N) \rrbracket$ de [SNL06], mais une autre notation peut être utilisée. Par exemple, Lim propose que le covariant aspect de la multiplication peut être explicite en exprimant (2.17) :

$$(U^{(1)}, U^{(2)}, \dots, U^{(N)}) \cdot g.$$

Comme autre exemple, Grigorascu et Regalia soulignent le rôle du tenseur de base dans la

multiplication par l'expression (2.17) en tant que : $(U^{(1)} \overset{g}{*} U^{(2)} \overset{g}{*} \dots \overset{g}{*} U^{(N)})$.

Qui est appelé le produit Tucker pondérée; la version non pondérée a $g = J$, le tenseur d'identité. Indépendamment de la notation, les propriétés d'un tenseur Tucker sont le même.

2.5.1.2 Propriétés du tenseur Tucker

Il est de notoriété publique (datant de [L.R.T 66]) que les versions matricielles du tenseur de Tucker (2.17) ont une forme particulière, plus précisément,

$$\mathbf{X}_{(R^{\times e}; J_N)} = (\mathbf{U}^{(r_L)} \otimes \dots \otimes \mathbf{U}^{(r_1)}) \mathbf{G}_{(R^{\times e}; J_N)} (\mathbf{U}^{(C_M)} \otimes \dots \otimes \mathbf{U}^{(C_1)}) \quad (2.18)$$

Où $R \in \{r_1, \dots, r_L\}$ et $e = \{c_1, \dots, c_M\}$. On notera que l'ordre des indices de \mathbf{X} et e est importante, et d'inverser l'ordre des indices est une source fréquente de erreurs de codage. Pour le cas particulier de mode n matricization (1), nous avons

$$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} (\mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \dots \otimes \mathbf{U}^{(1)})^T \quad (2.19)$$

De même, pour la version vectorielle (2), nous avons :

$$\text{vec}(\mathbf{x}) = (\mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(1)}) \text{vec}(\mathbf{g}) \quad (2.20)$$

2.5.1.3 Norme d'un tenseur de Tucker

Pour la discussion précédente, il est clair que la norme peut également être calculé de manière efficace si le tenseur de base est faible par rapport au tenseur global, par exemple, $J_n < I_n$ pour tout n . Soit \mathbf{x} un tenseur Tucker comme en (2.17). De §4.2.3, nous avons :

$$\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \langle \mathbf{g}, \mathcal{F} \rangle \text{ Avec } \mathcal{F} = [\mathbf{g}; \mathbf{W}^{(1)}, \dots, \mathbf{W}^{(N)}] \text{ et } \mathbf{W}^{(n)} = \mathbf{U}^{(n)T} \mathbf{U}^{(n)} \text{ pour tout } n. \quad (2.21)$$

2.5.1.4 Calcul $\mathbf{X}_{(n)} \mathbf{X}_{(n)}^T$ pour un tenseur Tucker

Pour calculer le $\text{rang}_n(\mathbf{x})$, nous avons besoin de $\mathbf{Z} = \mathbf{X}_{(n)} \mathbf{X}_{(n)}^T$. Soit \mathbf{X} un tenseur Tucker comme dans (2.17);

$$\begin{aligned} \text{puis : } \mathbf{Z} &= \mathbf{U}^{(n)} \mathbf{G}_{(n)} (\mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \dots \otimes \mathbf{U}^{(1)})^T \\ &\quad (\mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \dots \otimes \mathbf{U}^{(1)}) \mathbf{G}_{(n)}^T \mathbf{U}^{(n)}. \end{aligned}$$

En utilisant les propriétés du produit de Kronecker, cela réduit à :

$$\begin{aligned} \mathbf{Z} &= \mathbf{U}^{(n)} \mathbf{G}_{(n)} (\mathbf{V}^{(N)} \otimes \dots \otimes \mathbf{V}^{(n+1)} \otimes \mathbf{V}^{(n-1)} \otimes \dots \otimes \mathbf{V}^{(1)})^T \\ &\quad (\mathbf{U}^{(N)} \otimes \dots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \dots \otimes \mathbf{U}^{(1)}) \mathbf{G}_{(n)}^T \mathbf{U}^{(n)}. \end{aligned}$$

Où $\mathbf{V}^{(m)} = \mathbf{U}^{(m)T} \mathbf{U}^{(m)} \in \mathbf{R}^{J_n \times J_m}$ pour tous $m \neq n$ pour un coût de $O(J_m J_m^2)$.

2.6.2 Tucker Décomposition non négatif

Yong-Deok Kim, Seungjin Choi en 2007 ont appliques la Tucker Décomposition non négatif cette étude se base sur le principe suivant :

Non négatif tenseur factorisation (NTF) est une multivoie récente (multilinéaire) extension de la matrice non négative factorisation (NMF), où les contraintes de non-négativité sont imposées sur le modèle CANDECOMP/PARAFAC. Dans ce travail on considère le modèle Tucker avec les contraintes de non-négativité et on développe une nouvelle méthode de factorisation tenseur, dénommé non négatif Tucker décomposition (NTD). Les principales

contributions de cet article sont: (1) la mise à jour des algorithmes multiplicatifs pour NTD, (2) une méthode d'initialisation pour accélérer la convergence, (3) une méthode de contrôle clairsemé en tenseur factorisation. Plusieurs exemples de vision par ordinateur, nous montrent le comportement utile de la NTD, sur les méthodes du NTF et NMF existants. [Y-DK07]. L'Analyse des sous-espace ou analyse en composantes principales (PCA) est un modèle linéaire de données largement utilisé, dont la tâche est d'apprendre la matrice de base $A = [a_1 \dots a_R] \in \mathbb{R}^{m \times R}$ et la matrice variable de codage $S \in \mathbb{R}^{R \times l}$ qui minimise $\|X - AS\|^2$ ($\|\cdot\|$ désigne la norme euclidienne qui est également connue comme la norme de Frobenius lorsque l'argument est une matrice), compte tenu d'une matrice de données $X = [x_1 x_2 \dots x_l] \in \mathbb{R}^{m \times l}$. Un succès en PCA est l'analyse en composantes indépendantes (ICA) où le modèle linéaire de données $X = AS$ est appris que de tels vecteurs lignes de la matrice de codage variable sont statistiquement indépendants que possible, tout sous-espace analysé produit des composants non corrélés. Dans le cas où la matrice de données ne contient que des éléments non négatifs $X \in \mathbb{R}^{m \times l}$, la factorisation de matrice positive (PMF) ou la factorisation de matrices non négatives (NMF) est connue comme un outil utilisé dans l'apprentissage de la représentation sur la base des pièces - ainsi que dans l'extraction de caractéristiques. La vision par ordinateur implique un ensemble d'images ou de données vidéo qui peuvent être bien représentés par un tableau de données multivoies qui est connu tant que tenseur. Par exemple, un vecteur est une voie du tenseur, une matrice est un tenseur d'ordre 2, un cube est un tenseur d'ordre 3, et ainsi de suite. La structure à plusieurs voies reflète les coordonnées lignes, colonnes, RGB (ou HSV) couleurs, le temps, et ainsi de suite. Dans l'analyse de sous-espace classique ou méthodes de factorisation de matrice, les données d'image en 2D sont converties en images 1D ou vecteurs en négligeant la structure spatiale des données de l'image d'origine 2D. Récemment, les procédés d'extension 2D de l'analyse de sous-espace ont été proposés, y compris 2D-PCA et 2D-NMF. Un cadre général qui prend la structure à plusieurs voies en compte, est l'algèbre multilinéaire impliquant la décomposition de tenseur. Les méthodes de décomposition de tenseur exemplaires comprennent: (1) le modèle Tucker (également connu sous le nom multilinéaire SVD ou N-Mode SVD) [L.R.T66], (2) le CANDECOMP / modèle PARAFAC, (3) la factorisation de tenseur non négative (FSN) où la positivité des contraintes est incorporée dans le CANDECOMP/PARAFAC modèle ou le modèle de PARAFAC2 dans les applications de vision par ordinateur, la N-Mode SVD a été appliquée pour faire face à la représentation de l'image, montrant que, compte tenu de plusieurs modes tels que des personnes différentes, expressions, poses de tête

et des conditions d'éclairage améliore les performances la reconnaissance de visage. Multilinear ICA a également été appliquée à la représentation de face et la reconnaissance d'image. La NTF préserve la structure 2D de données d'image dans la factorisation conduisant à une décomposition supérieure dans le sens d'un codage clairsemé, comparé à NMF. Les méthodes existantes de NTF considèrent le modèle CANDECOMP/PARAFAC où un tenseur non négatif est approximé par une somme linéaire de produits de vecteurs non négatifs de sortie. Dans ce travail le modèle de Tucker [L.R.T66] est considéré et les contraintes de non-négativité sur le tenseur de base ainsi que le mode matrices. Ensuite, on développe des algorithmes de mise à jour multiplicatif pour l'apprentissage de la décomposition d'un tenseur Tucker non négatif avec restriction sur le tenseur base et mode matrices doit être négatifs. La mise à jour d'algorithmes multiplicatif matricize tenseur de manière itérative dans chaque mode puis résout le problème NMF. La méthode est appelée non négatif Tucker décomposition (NTD), afin de distinguer à partir des travaux antérieurs le NTF. En fixant le tenseur comme un tenseur structuré, NMF, le NTF, et nsNMF peuvent être représentés comme des cas particuliers de l'NTD. De plus les algorithmes de mise à jour aussi multiplicatifs peuvent être re-tirés de ceux de l'NTD. On présente une utilité pratique de méthode d'initialisation pour NMF et NTD, qui de façon spectaculaire accélère la convergence. On souligne également la rareté de contrôler les NTD à travers un noyau tenseur structuré, tout comme nsNMF. Récemment, Mørup et al. indépendamment ont mis au point une méthode de HONMF similaire à la NTD et l'accent sur l'analyse de données biomédicales. HONMF intègre l'additif L_1 norm pénalité pour clairsemé mais souffre de la convergence lente avec initialisation aléatoire. [Y-DK07]

2.6.2.1 Contexte analyse multivoie

Un bref aperçu de CANDECOMP/PARAFAC et Tucker modèle est présenté, qui est nécessaire pour de plus amples comprendre les NTD. Notations importantes sont dans le tableau 2.2 [Y-DK07]

Tableau 2.2 Notations simplifiant l'algèbre tensorielle. [Y-DK07]

Notation	Description
X	N-chemin tenseur
X	Matrice
$X_{(n)}$	Moden-matricization de tenseur X
$A^{(n)}$	Moden-matrice dans le modèle Tucker
\odot	Produit de Hadamard
\otimes	Produit de Kronecker
\circ	Produit Khatri-Rao
\circ	produit externe

Un tenseur N-chemin $X \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ à N indices (i_1, i_2, \dots, i_N) et ses éléments sont désignés par x_{i_1, i_2, \dots, i_N} où $1 \leq i_n \leq I_n$. Le moden-matricization de $X \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ réorganise les éléments de X pour former la matrice $X_{(n)} \in \mathbb{R}^{I_n \times I_1 I_2 \dots I_{n-1} I_{n+1} I_{n+2} \dots I_N}$ où $I_{n+1} I_{n+2} \dots I_N I_1 I_2 \dots I_{n-1}$ est en ordre cyclique suivant Kiers, l'indice le plus rapide évolution est i_{n+1} dans le cas de Moden-matricization un tenseur est analogue à vectoriser une matrice. Le produit scalaire de deux tenseurs x, y est défini comme étant $\langle x, y \rangle = \sum_{i_1, i_2, \dots, i_N} x_{i_1 i_2 \dots i_N} y_{i_1 i_2 \dots i_N}$ le Frobenius norme d'un tenseur X est donnée par $\|X\| = \sqrt{\langle X | X \rangle}$

2.6.2.2 La différence entre deux modèles [Y-DK07]

CANDECOMP/PARAFAC et modèles Tucker ont été indépendamment développés, offrant un terrain pour l'analyse multilinéaire. La principale différence entre ces deux modèles est en présence du tenseur de base dans le modèle de Tucker. Le tenseur de base permet aux vecteurs colonnes de matrices de mode à interagir entre eux afin de reconstruire le tenseur d'origine. Par exemple, un élément individuel du tenseur du noyau, $s_{j_1 j_2 \dots j_N}$, sert le poids de la combinaison associée $A_{:, j_1}^{(1)}, A_{:, j_2}^{(2)}, \dots, A_{:, j_N}^{(N)}$. En revanche, dans le CANDECOMP

/Modèle PARAFAC ces interactions se produisent entre $A_{:,j}^{(1)}, A_{:,j}^{(2)}, \dots, A_{:,j}^{(N)}$. À première vue, le CANDECOMP /Modèle PARAFAC peut être considérée comme un cas particulier de modèle Tucker lorsque le tenseur de base est égal à un super-diagonal unité tenseur, c'est-à-dire $\mathcal{S} = \mathbf{I}$ où $i_{1i_2 \dots i_N} = \delta_{i_1 i_2 \dots i_N}$. Cependant en général, le tenseur de base dans le modèle de Tucker ne peut pas être tourné vers le tenseur super-diagonal unité comme dans le modèle CANDECOMP/PARAFAC. Strictement parlant, seul le modèle CANDECOMP/PARAFAC a un caractère unique propriété jusqu'à l'échelle et la permutation des matrices. Même si le modèle Tucker n'a pas la propriété d'unicité, il est plus utile pour la compression de données. Tucker le modèle nécessite un petit nombre de vecteurs de composant pour le mode respectif que le CANDECOMP /Modèle PARAFAC, car il utilise toutes les combinaisons des vecteurs de mode (vecteurs colonnes de matrices de mode).

2.6.2.3 Non négative décomposition Tucker

Étant donné un non négatif N-fois tenseur $X \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, décomposition de Tucker non négative (NTD) cherche une factorisation de X , qui est de la forme :

$$X \approx \hat{X} = \mathcal{S} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \dots \times_N \mathbf{A}^{(N)}, \quad (2.22)$$

Où le cœur tenseur $\mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ et mode $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_n}$ pour $n = 1, \dots, N$ se limitent à avoir seuls les éléments non négatifs dans la factorisation. Comme écart mesures entre les données X et le modèle \hat{X} , nous examinerons la fonction d'erreur J_{LS} et J_I -divergence qui sont donnés par : $J_{LS} = \|X - \hat{X}\|_F^2$, (2.23)

$$J_I = \sum_{i_1, i_2, \dots, i_N} \left\{ x_{i_1 i_2 \dots i_N} \log \frac{x_{i_1 i_2 \dots i_N}}{\hat{x}_{i_1 i_2 \dots i_N}} - x_{i_1 i_2 \dots i_N} + \hat{x}_{i_1 i_2 \dots i_N} \right\} \quad (2.24)$$

NTD fournit une forme générale du tenseur non négative factorisation, y compris NMF, nsNMF et NTF à extraordinaire cas. Tableau. Modèles 2.3 résumés pour NMF, nsNMF, et NTF dans le cadre de décomposition du tenseur. Tout comme les montré en 2.2.4 sec., le tenseur de base et le mode des matrices sont apprises par itération, au moyen d'algorithmes multiplicatifs mise à jour. Cependant, au lieu d'apprendre le tenseur de noyau dans NTD, nous pouvons placer une structure spécifique prédéterminée sur le tenseur de noyau. De cette façon, les nsNMF et NTF émergent de la NTD.

2.6.2.4 Les algorithmes de mise à jour multiplicatifs

Nous établissons des algorithmes multiplicatifs de mise à jour pour les matrices $\mathbf{A}^{(n)}$ ainsi que le tenseur de base \mathcal{S} pour l'NTD. Mise à jour des algorithmes sont assez semblables à ceux de NMF. Il est possible d'obtenir directement des algorithmes multiplicatifs de mise à jour d'une manière contre-vérification pour minimiser l'erreur fonction (2.23) ou

(2.24), tel qu'il est fait en NTF .Toutefois,mise à jour des algorithmes sous forme de contre-verification semblent être très compliqué, qui pourraient déranger un efficace et compact mise en œuvre. Ainsi, l'approche que nous adoptons ici est pour utiliser pleinement les représentations de la matrice du modèle Tucker, via les propriétés du produit de Kronecker et $vec(\cdot)$ opération. Afin d'en tirer une règle de mise à jour, nous utilisons ce qui suit propriétés :

$$vec(UBV^T)=(V \otimes U)vec(B), \quad (2.25)$$

$$[U \otimes V]^T=U^T \otimes V^T,$$

$$(U \otimes V)(B \otimes C)=UB \otimes VC.$$

Nous définissons $A^{(n)}=[A^{(n-1)} \otimes \dots \otimes A^{(1)} \otimes \dots \otimes A^{(N)} \otimes \dots \otimes A^{(n+1)}]$, (2.26)

Qui recueille les produits Kronecker des matrices de mode sauf pour $A^{(n)}$ d'une manière cyclique vers l'arrière.

2.6.2.5 Mise à jour des matrices de mode

La matricization de la mode-n de l'NTD conduit à :

$$X_{(n)}=A^{(n)}S_{(n)}A^{(n)T}=A^{(n)}S_A^{(n)} \quad (2.27)$$

Tableau 2.3 Modèles pour NMF, nsNMF et NTF sont résumées dans le cadre de la décomposition de tenseur.[Y-DK07]

Model	matrix representation	NTD representation
NMF	$X \approx AS$	$X \approx I \times_1 A \times_2 S^T$
nsNMF	$X \approx AMS$	$X \approx M \times_1 A \times_2 S^T$
NTF	$X \approx \sum_{r=1}^R A_{(n)}^{(1)} \circ A_{(n)}^{(2)} \circ \dots \circ A_{(n)}^{(N)}$	$X \approx I \times_1 A^{(1)} \times_2 A^{(2)} \dots \times_N A^{(N)}$
nsNTF	N/A	$X \approx M \times_1 A^{(1)} \times_2 A^{(2)} \dots \times_N A^{(N)}$

Qui est de la même forme que le modèle NMF. Ainsi, la mise à jour algorithmes pour $A^{(n)}$ suit le NMF mise à jour des algorithmes pour ALS ;

$$A^{(n)} \leftarrow A^{(n)} \odot ([X_{(n)} S_A^{(n)T}] / [A^{(n)} S_A^{(n)} S_A^{(n)T}]). \quad (2.28)$$

2.6.2.6 Résultats

L'implémentation MATLAB de l'NTD utilise en partie la boîte à outils tenseur [BW.B06]. En utilisant AT&T et Université de Cambridge, ORL dataset qui se compose de 400 images de visage dans les expériences. On fixe l'emplacement des deux yeux, rogner le visage et le redimensionner à 48×48 pixels. Donc $48 \times 48 \times 400$ tenseur et $48 \cdot 48 \times 400$ matrice servent à NTD et NMF respectivement. Nous avons mis $J_1 = J_2 = 24$, $J_3 = 30$ et $R = 30$ pour MTN et NMF respectivement dans toute expérience. Tout d'abord, on compare cette méthode

d'initialisation avec aléatoireinitialisation. Les résultats, Fig2.8, montrent que nos méthodes considérablement accélérer la convergence. On montre que l'NTD contrôle en effet la faible densité avec lissage de matrices. On a mis le paramètre θ lissage en 0, 0,5 et 0,8. La faible densité, mesurée par Hoyer, de $S, A^{(1)}, A^{(2)}, A^{(3)}$ augmente de [0,56 0,50 0,44 0,53] à [0,59 0,67 0,56 0,56] et [0,82 0,59 0,58 0,57] respectivement. Fig2.9 montre quelques images de la base de chaque résultat. Tranches de la $S \times_1 A^{(1)} \times_2 A^{(2)}$ sont des images de base dans ce cadre, où l'image est matrice 2D plutôt que 1D vecteur. Comme le θ augmentations, des images de base deviennent également plus clairsemées. On synthétise les données pour tester les nsNTF.

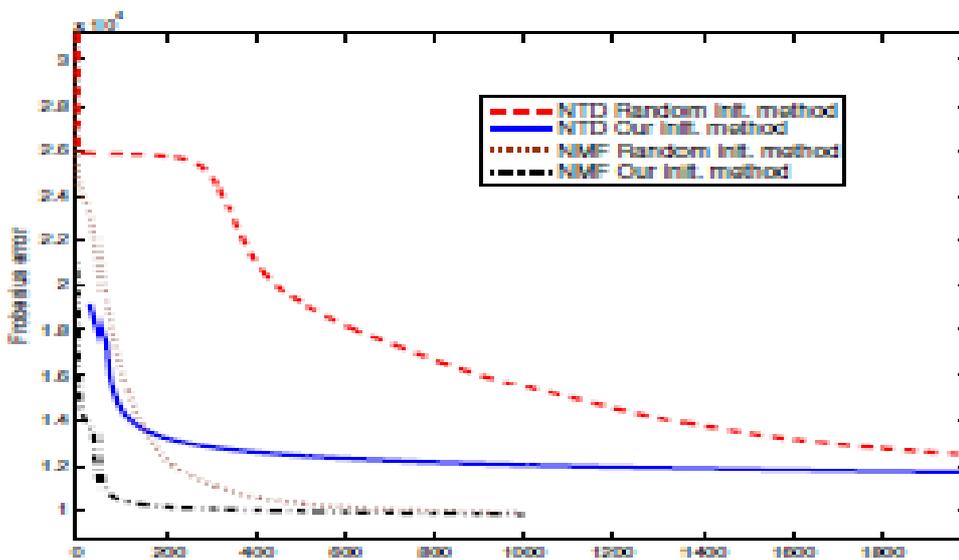


Figure 2.8 Trace de l'erreur de Frobenius : NMF et NTD avec différentes méthodes d'initialisation. [Y-DK07]



Figure 2.9 Exemples d'images base. De haut en bas: (1) $\theta = 0$; (2) $\theta = 0,5$; (3) $\theta = 0,8$. Comme le θ augmente, les images de base deviennent plus clairsemées. [Y-DK07]

(Pour les résultats d'expériences illustrés dans la figure 2.8 On propose des méthodes vitesses radicalement convergentes. Les itérations dans l'étape d'initialisation sont également ajoutées en cette trace de l'erreur pour les comparaisons justes).

Le tenseur de données est généré par $X = \sum_{r=1}^3 A_{:,r}^{(1)} \circ A_{:,r}^{(2)} \circ A_{:,r}^{(3)} + |v|$, (2.29)

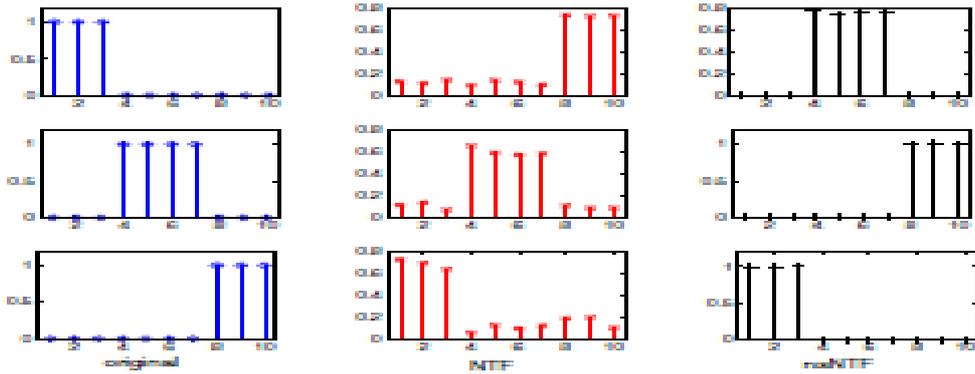


Figure 2.10 De gauche à droite: (1) signal original ; (2) signal récupéré de la NTF ; (3) signal récupéré par nsNTF $\theta = 0,8$;[Y-DK07]

Le nsNTF permet de trouver original signal avec succès.

Où $A^{(1)} = A^{(2)} = A^{(3)}$ sont 10×3 matrices et. Enfin, on montre le comportement de NMF et NTD, compte tenu des images contaminées par le bruit. Chaque image de visage a été superposée par le bruit « poivre & sel ». Sur la Fig 2.12 on peut voir que les images apprises par NTD ne sont pas sensibles au bruit « poivre & sel », alors que celles de NMF sont détériorées. Images reconstruites par NMF et NTD figurent également sur la Fig 2.11, où la supériorité de l'NTD au sujet de la robustesse au bruit est observée. Rapport signal-bruit de moyennes pics (PNSR) pour NMF et NTD sont 22.16dB et 24.12dB, respectivement. La plupart des pixels dans une région risquent davantage d'être similaires les uns des autres, alors que ce n'est pas vrai dans le cas du bruit 'poivre & sel'. NMF rompt la structure de région et elle doit remodeler des images 2D en vecteurs 1D. En revanche l'NTD, structure voisine 2D est conservée dans le cadre du tenseur 3 voies, conduisant à mieux reconstruire les images dans le cas du bruit 'poivre & sel'. Et c'est la raison pour laquelle NTD est robuste au bruit.[Y-DK07]

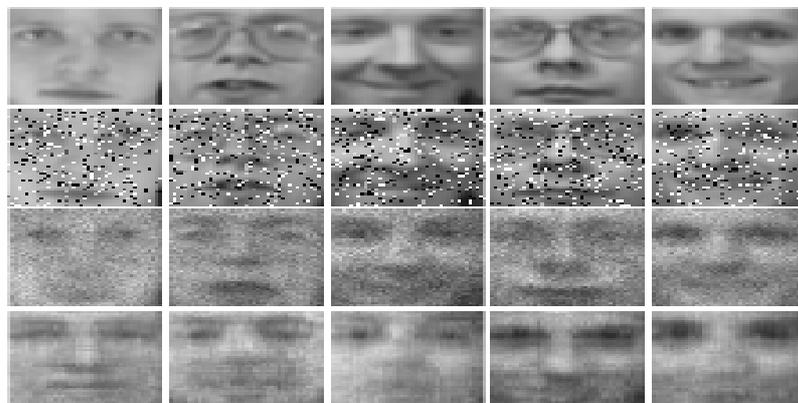


Figure 2.11 De haut en bas : des images originales (1) de visage ; (2) images contaminées par le bruit 'poivre & sel' ; (3) reconstruit des images par NMF ; (4) reconstruit des images par NTD.[Y-DK07]

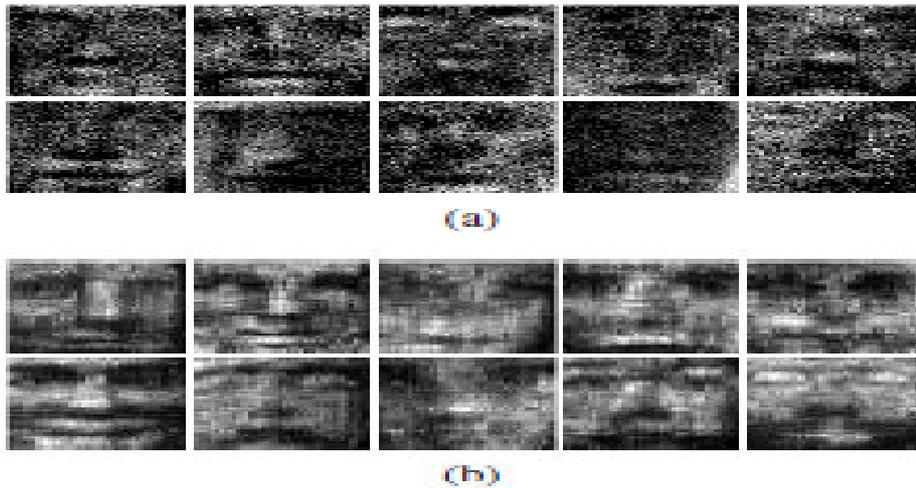


Figure 2.12 Images de base apprises par NMF (a) et (b) NTD, donné une ensemble de visage photographié contaminé par poivre & bruit de sel. [Y-DK07]

2.6.3 Contexte de l'imagerie hyperspectrale et problématiques abordées

Nadine RENARD en 2008 [NR08] a étudié le contexte de l'imagerie hyperspectrale et les problématiques abordées et pondérées cette étude se base sur le principe suivant :

- *La présence du bruit d'acquisition dans les images hyperspectrales*
- *et la dimension importante des spectres,*

Ceux-ci font que l'efficacité des algorithmes de détection de cibles et de classification est dégradée. Dans cette section, nous présentons certaines méthodes, existant dans la littérature, appliquées en prétraitement pour améliorer la performance des algorithmes d'analyse des images hyperspectrales. De ce fait deux parties se distinguent :

- 1) un état de l'art relatif aux *méthodes de débruitage*
- 2) et un autre relatif aux *méthodes de réduction de dimension.*

Les méthodes présentées ici sont utilisées pour comparer les méthodes de filtrages et de réduction multimodales respectivement proposées.

2.6.3.1 Etat de l'art des méthodes de débruitage

Une opération de débruitage des données hyperspectrales peut être pertinente pour améliorer l'efficacité des algorithmes d'analyse. Nous avons vu précédemment que les applications de l'imagerie hyperspectrale tirent parti de la richesse de l'information spectrale. Cependant, la variabilité spectrale d'une même cible (ou intra-classe) est certes porteuse d'une information riche pour la télédétection, mais pas toujours désirable. En effet, la variabilité spectrale est aussi due à des variations au sein même de l'espèce qui rend délicat le processus de classification des différentes espèces, alors qu'elle est très intéressante si le

but est d'identifier ces variations elles même. La distinction entre l'information utile (le signal) et l'information non-désirable (le bruit) est alors dépendante de l'application visée.

Plusieurs techniques de débruitage existent. La première solution consiste à importer les techniques de filtrage de l'imagerie scalaire. Cependant au vue de la spécificité des données hyperspectrales, la deuxième solution suggère une extension de ces méthodes au cas vectoriel. Nous introduisons ici, quelques unes de ces méthodes.

1 Filtres de l'imagerie scalaire

Considérant que chaque bande spectrale est une image bidimensionnelle (2D), les filtres classiques, utilisés en imagerie scalaire, peuvent être appliqués à chaque bande spectrale indépendamment les unes des autres. Dans ce cadre, les filtres de lissage reconnus en imagerie scalaire peuvent être considérés : l'efficacité des filtres morphologiques [J.S89] et des filtres par diffusion anisotrope [J.W96] a été démontrée en imagerie scalaire.

- **Filtre de Wiener** [NR08] : Dans ce contexte, le filtre de Wiener 2D minimise l'erreur quadratique moyenne entre le signal estimé et le signal original. Il est considéré comme "*mathématiquement optimal*" même s'il n'est pas toujours évident à mettre en œuvre. Ce filtre caractérise le signal et le bruit par leurs propriétés statistiques. Il utilise la statistique d'ordre deux pour "*optimalement*" décorrélérer les données en les transformant dans le domaine de Karhunen-Loève (KL). Dans ce domaine transformé, le signal et le bruit sont facilement discernables. Le principe du filtre de Wiener est d'estimer le signal en décorrélant les données et en pondérant les coefficients de corrélation par décroissance (c'est-à-dire de privilégier les directions de grande variance) et finalement de récorrélérer le signal. Ce processus aboutit au filtre de Wiener de la i_3 ^{ième} bande spectrale, défini tel que :

$$H_{i_3} = \frac{S_x}{S_x + S_b} = \frac{E[(x - E[x])^2]}{E[(x - E[x])^2] + \theta b^2}, \text{ avec } i_3 = 1 \dots I_3$$

Dont l'expansion en valeurs propres de la matrice de covariance de la bande spectrale i_3 ^{ième} est telle que : $H_{i_3} = U \Lambda U^T$

Où, U est la bande dans le domaine KL contenant les vecteurs propres et Λ est la matrice de pondération contenant les valeurs propres. L'estimation du signal de la bande spectrale i_3 basée sur un filtrage de Wiener est donnée par :

$$\hat{X}(:, :, i_3) = U \Lambda U^T R(:, :, i_3), \text{ avec } i_3 = 1 \dots I_3 \quad (2.30)$$

- Débruitage par ondelettes

Ces dernières années, plusieurs méthodes ont bénéficié de la concentration de la transformée en ondelettes, permettant ainsi d'approximativement décorréler à la fois un signal stationnaire et non-stationnaire. Une technique utilisée en hyperspectrale est le seuillage d'ondelettes (*wavelet shrinkage*), dont les pionniers sont Donoho et Jonhstone [D.D95]. Le principe de cette méthode est simple : on effectue une transformation 2D en ondelettes discrètes sur l' i_3 bande spectrale, on applique ensuite un seuillage sur les coefficients pour ne garder que les coefficients les plus significatifs dans les sous-bandes de détails. En effet, comme le bruit tend à se répartir sur tous les coefficients d'ondelettes, on sait que le SNR sera plus grand sur les coefficients les plus importants en amplitude. En ne gardant que ceux-là, on augmente généralement le SNR du signal.

Si \mathbf{W} est une matrice de la transformée en ondelettes orthogonale, la transformée en ondelettes du signal utile s'écrit : $\mathbf{W}(\mathbf{r}) = \mathbf{W}(\mathbf{x}) + \mathbf{W}(\mathbf{b})$

$\mathbf{W}(\mathbf{r})$ est composé de coefficients de grande amplitude (signal utile + bruit) et de coefficients de faible amplitude (bruit). La transformation en ondelettes permettant de décorréler les données, les coefficients peuvent être traités indépendamment. Cette transformation approche la transformée du domaine KL. Les coefficients sont donc seuillés pour ne conserver que les coefficients liés au signal utile. $\hat{\mathbf{x}} = \mathbf{W}(\mathbf{r})^{-1}(\eta(\mathbf{r}))$

Où η est un seuillage doux ou fort. Cette opération de conservation/rejet peut être considérée comme une opération de filtrage diagonal dans le domaine des ondelettes pour la bande spectrale i_3 , telle que : $\mathbf{H}_{i_3} = \text{diag} [h_1, h_2, \dots, h_{i_3}]$.

Par conséquent, l'estimation du signal basée sur une technique de seuillage d'ondelettes est donnée par : $\hat{\mathbf{X}}(:, :, i_3) = \mathbf{W}(\mathbf{r})^{-1} \mathbf{H}_{i_3} \mathbf{W}(\mathbf{r}) \mathbf{R}(:, :, i_3)$ (2.31)

Le filtrage par seuillage d'ondelettes nécessite de faire un choix sur le type de famille d'ondelettes à utiliser (*Daubechies, Symlets, Coifflet, Haar, etc*), sur le niveau d'approximation et sur le type de seuillage à effectuer (*Heursure, Sqtwlog, Minimaxi, etc*) [D.D95]. En comparant les équations 2.30 et 2.31, ce filtrage peut être vu comme une approximation du filtre *optimal* linéaire de Wiener. Il est basé sur la transformée en ondelettes (au lieu de la transformée KL pour Wiener) et utilise un seuillage pour ne conserver que le signal utile (au lieu d'une pondération *optimale* dépendante de la statistique du signal pour Wiener). Importer les techniques de filtrage, développées pour l'imagerie scalaire, à l'imagerie multi-bandes a été une stratégie intuitivement adoptée pour débruiter les images couleur et les données multispectrales. Cependant, en imagerie hyperspectrale,

analyser les bandes spectrales séparément en les considérant comme des signaux indépendants, se révèle être une stratégie simple mais possédant des inconvénients majeurs. En effet, cela implique autant de traitements qu'il y a de bandes spectrales et donc un temps d'analyse non négligeable. De plus, cette stratégie ignore les fortes relations inter-bandes existantes, pourtant porteuse d'une riche information. D'autres stratégies, plus adaptées aux spécificités des données hyperspectrales, ont donc dû être mises en place.

2 Filtres hybrides

Une autre approche consiste à étendre au cas vectoriel les traitements d'images scalaires existants. L'extension d'une méthode scalaire vers une méthode vectorielle nécessite la définition d'une norme et/ou d'un critère de distance et/ou d'une relation d'ordre, suivant les besoins de l'algorithme de traitement. Par exemple, un traitement scalaire est appliqué sur chaque bande spectrale, et le calcul d'une norme de chaque vecteur pixel est utilisé pour projeter le résultat final du filtre dans un espace de dimension 1. Une réduction de dimension des données d'observation est possible pour alléger le temps de calcul. De cette façon, l'extension vectorielle du filtrage morphologique et du filtre par diffusion anisotrope ont trouvé un intérêt en imagerie hyperspectrale. Dans une autre approche, Hunt et Kubler ont mis en évidence l'existence d'un domaine de transformation KL qui rend les bandes spectrales statistiquement décorrelées. En effet, en diagonalisant la matrice de corrélation spectrale, il est alors possible dans ce domaine transformé d'utiliser un traitement bidimensionnel sur chacune des bandes indépendamment sans perte d'information. Ces filtres dits *hybrides*, sont basés sur l'hypothèse que l'autocorrélation du signal, qui décrit les relations entre les bandes spectrales et dans les bandes (relation spatiale), est séparable. [NR08] Cette observation tend à montrer qu'il existe un sous-espace de projection où une information sur le bruit est rendue accessible et permet ainsi d'adapter le filtrage dans cet espace de projection. Suivant cette approche, la technique de réduction de bruit la plus répandue en hyperspectrale et multispectrale est le Maximum Noise Fraction (MNF), l'équivalent de Noise-Adjusted Principal Component Analysis (NAPCA). Cette méthode consiste à projeter l'image originale dans un espace où les composantes sont classées par SNR croissant. Pour se faire, elle considère le problème en deux temps : (i) une première étape estime la puissance du bruit présent sur l'image originale en considérant la corrélation spatiale entre les pixels d'une région homogène (où l'autocorrélation spatiale du signal est plus forte que l'autocorrélation du bruit). Utilisant la matrice de covariance du bruit ainsi estimée, une première PCA est appliquée sur les données d'observation, permettant de blanchir le bruit spectralement décorrélié des données

; (ii) Enfin, une seconde PCA est appliquée sur les projetés, maximisant le SNR des nouvelles composantes

Finalement, en considérant cette hypothèse de séparabilité, tous les filtres issus de l'imagerie scalaire peuvent être appliqués dans ce domaine transformé. Dans ce cadre, montre que le filtre de Wiener qui prend en compte la corrélation spectrale par la transformée KL est plus efficace qu'un filtre de Wiener appliqué indépendamment sur chaque bande spectrale. De même, utilise la transformée en ondelettes tandis que applique un filtre non linéaire par diffusion anisotrope sur chaque bande alors orthogonalisée par la transformation KL.

L'estimation d'une nouvelle base spectrale permettant une décorrélation spectrale est aussi possible en utilisant l'analyse en composantes indépendantes (ICA) [A.H&E.O00] ou la transformée de Fourier discrète. En utilisant cette dernière transformée, ont mis en évidence l'efficacité du filtrage par seuillage d'ondelettes sur cette base. Récemment, considérant un bruit dépendant du signal (bruit de photon) a proposé un algorithme de restauration basé sur le seuillage des coefficients de la transformée en ondelettes opérant dans le domaine des dérivées (où le bruit est amplifié). Ces méthodes bénéficient des caractéristiques spatiales et spectrales cependant ces dernières sont analysées indépendamment et de façon séquentielle. Finalement, le choix de la méthode de débruitage dépend drastiquement de l'application visée. Récemment dans un contexte de détection de cibles peu présentes dans la scène, a montré l'intérêt d'un filtre basé sur le filtre local *Lee's adaptive Wiener* qui privilégie l'information spectrale au détriment d'une restauration spatiale.

2.6.3.2 État de l'art des méthodes de réduction de dimension

Une des solutions à mettre en œuvre pour limiter les effets du fléau de la dimension est de réduire cette dimension avant d'analyser les données. La réduction de la dimension présente un grand intérêt et elle se justifie de plusieurs manières. D'une part, parce qu'il existe beaucoup de redondance spectrale. Cette forte corrélation s'explique par les propriétés physiques des matériaux qui ne changent pas radicalement d'une longueur à l'autre. De plus, une caractéristique intéressante est que les données de grande dimension et de distribution quelconque, voient leur projection dans un sous-espace réduit tendre vers une distribution gaussienne. Ce dernier point est intéressant pour l'utilisation d'algorithmes paramétriques de classification.

Les techniques de réduction de dimension (RD) sont traditionnellement divisées en deux catégories: les méthodes d'*extraction d'information* et les méthodes de *sélection*

d'information. Les méthodes d'extraction construisent à partir des variables originales, de nouvelles variables réduites ; tandis que les méthodes de sélection cherchent un sous-ensemble de variables parmi les variables originales selon un critère. Dans la suite, nous allons nous focaliser sur les méthodes d'extraction de caractéristiques qui sont certainement les plus utilisées. Toutes ces méthodes de réduction considèrent chaque profil spectral comme un vecteur aléatoire de dimension I_3 et visent à le transformer en un vecteur aléatoire de dimension p ($< I_3$) dans l'espace réduit.

1 Méthodes linéaires

Parmi les techniques de réduction, l'analyse en composantes principales (PCA) est certainement la méthode la plus connue. Basée sur un critère énergétique, elle cherche une combinaison linéaire des variables initiales de variances maximales. La PCA est basée sur l'existence de forte corrélation entre les bandes spectrales. Elle est la projection linéaire orthogonale qui vise à décorrélérer les composantes spectrales dans l'espace transformé. Une variante de cette méthode de réduction ajustée au bruit est connue sous le nom de *Maximum*

Noise Fraction (MNF) ou l'analyse en composantes principales ajustées au bruit (NAPCA). Cette méthode de réduction revient à chercher les composantes principales qui maximisent le rapport signal à bruit (SNR). Elle est l'équivalente de deux PCA en cascade, la première diagonalise l'estimation de la matrice de covariance pour "sphériser" le bruit, la deuxième estime les directions de projections de variances maximales.

Dans la même lignée, l'analyse en composantes indépendantes (ICA) [A.H&E.O00] cherche les combinaisons linéaires les plus indépendantes statistiquement possibles. Utilisant les statistiques d'ordre quatre, cette méthode de réduction est plus à même de caractériser les objets peu présents dans une scène. Une autre méthode de réduction par transformation linéaire est la poursuite de projection (PP) [J.F&J.T74]. Initialement utilisée pour visualiser les données de grande dimension, la PP cherche les combinaisons linéaires les moins gaussiennes possibles. Dans ce sens, la PP se trouve très proche de l'ICA. Plus tard, l'intérêt de la PP est la généralisation de toutes les méthodes de réduction, par transformation linéaire, qu'elle permet en définissant un *indice de projection*. La PP consiste alors à rechercher la direction de projection « la plus intéressante » possible, définie par un critère qualitatif qui est l'indice de projection.

2 Méthodes non-linéaires

Plus récemment des méthodes de réduction de dimension non-linéaires ont été proposées. Ces méthodes mettent en évidence les dépendances non-linéaires des données

hyperspectrales. L'extension de la PCA au cas non-linéaire est le *Kernel-PCA*, qui utilise les fonctions noyaux SVM en amont de la PCA. L'analyse en composantes curvilignes (CCA¹⁴ (*14-Curvilinear component analysis*)) cherche les projections non-linéaires qui préservent la topologie locale initiale des données. Le concept n'est alors plus statistique mais géométrique. De façon similaire, nous pouvons citer les extensions que sont l'analyse en distances curvilinéaires (CDA), la *locally Linear Embedding* (LLE) et la méthode *Isomap*.

3 Analyse discriminante

Pour la classification des données, les méthodes d'analyses discriminantes apportent l'intérêt de réduire la dimension des données spectrales en se fondant sur la distinction entre les classes. Ces méthodes nécessitent une définition au préalable des statistiques de chaque classe. Ce sont donc des méthodes supervisées qui sont autant perturbées par l'effet de HUGHES. La méthode d'analyse discriminante DAFE¹⁵ (*15. Discriminant Analysis Feature Extraction*) consiste à maximiser le rapport entre la variance inter-classe et la variance moyenne intra-classe. De ce fait, la dispersion entre les centres des classes doit être maximisée alors que la dispersion d'une même classe autour de son centre doit être minimisée. Ce concept simple est aussi rapide en calcul. Il a cependant deux limitations. La première rappelle que si deux classes ont des moyennes proches, même si elles se différencient de part leur covariance, DAFE ne générera pas un sous-espace efficace pour la classification. La deuxième limitation est que la dimension de l'espace de projection réduit est limitée par le nombre de classe moins un.

Une extension de DAFE permet de s'affranchir de ces limitations. Il s'agit de DBFE¹⁶ (*16. Decision Boundary Feature Extraction*) qui au lieu d'estimer les statistiques de chaque classe, utilise les échantillons d'entraînement directement. La méthode localise la frontière de décision entre deux classes et définit les paramètres discriminants informatifs (lien entre les échantillons des deux classes) et les paramètres discriminants redondants (lien entre les échantillons d'une même classe). Parce qu'elle utilise les échantillons d'entraînement au lieu de leur statistique, cette méthode est très dépendante du nombre d'échantillons d'entraînement utilisé.

Dans l'étude précédente nous avons présenté le principe d'acquisition des données hyperspectrales. Du fait de l'importance du nombre de bandes acquises (en général plus de 200) et du caractère contigu des bandes spectrales (résolution spectrale de 10 nm), les applications de détection de cibles, classification en zones géographiques et démixage révèlent le potentiel discriminant de telles données.

Mais la nature même des données hyperspectrales entraîne des contraintes qui perturbent l'efficacité des algorithmes. Nous avons introduit dans ce chapitre, deux problématiques directement liées aux données hyperspectrales, qui sont :

1. Le bruit dû aux processus d'acquisition.
2. La grande dimension des données.

L'état de l'art des méthodes qui répondent à ces deux problématiques révèlent que l'analyse des données hyperspectrales est fortement dépendante de la façon dont elles sont considérées.

Est-ce une concaténation d'images ou une concaténation de spectres ?

Ce choix implique l'analyse des données dans un espace donné. Lorsque les relations spatiales sont prises en compte, le traitement d'image est valorisé ; lorsque l'information spectrale est privilégiée, le traitement du signal est considéré. nous proposons des méthodes originales qui répondent aux deux problématiques respectives. Notre choix s'oriente vers une considération des données hyperspectrales dans leur ensemble. Par conséquent, les données seront représentées par un tenseur d'ordre 3, avec la définition d'un mode spectral et de deux modes spatiaux. Cette nouvelle représentation va impliquer le développement d'outils basés sur l'algèbre multilinéaire. L'étude suivante introduit les définitions et les propriétés du traitement du signal tensoriel.

Dans l'étude précédente, nous avons donné la définition d'un *tenseur* qui est utilisée dans la suite pour proposer une nouvelle représentation des données hyperspectrales. Dans un deuxième temps, nous avons présenté les propriétés des principaux opérateurs tensoriels. Nous avons mis en évidence le fait que la généralisation des outils bilinéaires aux outils multilinéaires n'est pas triviale. L'une des principales distinctions est le non extension du théorème d'ECKART-YOUNG [Y.E&G.Y36] au cas tensoriel (tenseur d'ordre > 2). Il en résulte deux définitions de rangs distincts pour un tenseur :

1. $D1$: Le rang n -modal K_n est le rang multilinéaire du tenseur A d'ordre N noté (K_1, \dots, K_N) . Ce rang généralise le rang colonnes et lignes d'une matrice pour chaque dépliante n -modale.
2. $D2$: Le rang tensoriel K . Ce rang généralise le rang associé au produit extérieur d'une matrice.

Ces deux définitions de rangs sont à la base de deux décompositions tensorielles qui leur sont associées et qui généralisent la décomposition en valeurs singulières d'une matrice. Nous avons ainsi introduit deux décompositions tensorielles :

1. $D1$: Le modèle de TUCKER3 qui généralise la représentation matricielle de la SVD. Ce modèle permet l'extension de la décomposition en valeurs singulières à l'ordre supérieur, nommée HOSVD. Le modèle de TUCKER3 est aussi à l'origine de l'approximation d'un tenseur en rangs (K_1, \dots, K_N) inférieure, notée la LR-TA- (K_1, \dots, K_N) .
2. $D2$: La décomposition tensorielle PARAFAC/CANDECOMP qui généralise la représentation canonique de la SVD. Ce modèle décompose un tenseur en une somme de tenseur de rang un ; il permet l'analyse en composantes principales multimodale.

Plusieurs autres décompositions tensorielles ont été proposées comme étant des variantes ou des extensions de ces deux modèles. Ces propositions croissantes montrent l'intérêt qui est porté au traitement tensoriel et qui trouve sa place dans des domaines d'application très variés. [NR08]

Dans la suite, nous présentons la représentation des données hyperspectrales par un tenseur d'ordre 3. L'ensemble de méthodes tensorielles développées sont basées sur le modèle de TUCKER3. En effet, ce travail s'inscrit dans la continuité des travaux de D. MUTI qui a proposé le filtre de Wiener multimodal basé sur TUCKER3 pour le débruitage des signaux sismiques et des images couleurs. De plus le modèle de TUCKER3 trouve une sémantique plus intuitive à l'application des images hyperspectrales que le modèle de PARAFAC/CANDECOMP. En effet, dans l'étude suivante on montrera l'importance de la flexibilité des valeurs des rangs (K_1, K_2, K_3) qui sont dissociées dans chaque n -mode alors que PARAFAC/CANDECOMP est associé à une valeur unique du rang tensoriel. De plus, l'orthogonalité des matrices n -modales rend le modèle de TUCKER3 plus facilement interprétable. Cependant, les autres décompositions tensorielles, en les adaptant aux données hyperspectrales, proposent une perspective intéressante pour l'extension des méthodes.

4 Réduction de dimension-Application pour l'amélioration de la classification

Des méthodes multimodales pour l'extraction d'information dans le but d'améliorer les performances des classifieurs. Cette problématique a été abordée à l'étude précédente, nous nous focalisons ici sur les méthodes de réduction de dimension par transformation linéaire, parmi lesquelles sont considérées : l'analyse en composantes principales (PCA) et indépendantes (ICA), ainsi que la poursuite par projection (PP). Après avoir formulé mathématiquement le problème de réduction linéaire de dimension, nous rappelons le principe de ces méthodes matricielles.

L'objectif est d'étendre ces méthodes d'algèbre linéaire au tenseur d'ordre 3. Illustre l'originalité des méthodes proposées, basées sur l'algèbre multilinéaire. Deux analyses conjointes sont effectuées :

1. Extraire P composantes spectrales (3-mode) en tenant compte de l'information spatiale.
2. Projeter simultanément les données dans un sous-espace spatial (1,2-mode) de dimension- $(K_{1,2})$ réduite.

Fixant cet objectif, la présentent deux méthodes multimodales. Elles se distinguent par la méthode de réduction spectrale utilisée :

1. la LRTAdr- (K_1, K_2, P) utilise la PCA;
2. l'hybride TUCKER2/PPdr- (K_1, K_2, P) utilise la PP .

Si l'influence du nombre, P , de composantes spectrales retenues est mis en avant par l'effet de HUGHES, les méthodes multimodales proposées révèlent l'influence des valeurs des rangs spatiaux $K_{1,2}$ sur les résultats de classification. Un critère est alors proposé pour estimer les valeurs de $K_{1,2}$ qui maximisent le taux de bonne classification.

L'efficacité des méthodes de réduction de dimension (RD) est évaluée en fonction de l'amélioration des résultats de classification des données réelles HYDICE 1. N.RENARD [NR08] considère trois algorithmes de classification qui ont été présentés à l'étude précédente : *spectral angle mapper* (SAM), le maximum de vraisemblance (ML), et la distance de Mahalanobis (MD).

La méthode de réduction de données [D.H&W.X03] cherche à favoriser les statistiques d'ordre supérieur pour l'extraction de données. En effet, les cibles de petite taille ou qui sont peu présentes dans la scène ne sont pas correctement caractérisées par les statistiques du second ordre. Ceci est dû au fait que les observations de ces cibles ne sont pas en assez grand nombre pour constituer une statistique fiable. Dans ce cas de figure, la PCA n'est pas en mesure de capturer leur information, contrairement à ICA qui utilise les statistiques d'ordre supérieur.

2.6.3.3 Application à l'imagerie hyperspectrale

À l'étude précédente, nous avons vu que les données hyperspectrales peuvent être considérées comme une concaténation de spectres. On suppose qu'un spectre est un vecteur aléatoire $\mathbf{r} = (r_1, r_2, \dots, r_{I_3})^T$ avec I_3 le nombre de bandes spectrales. Dans ce contexte, la dimension notée L précédemment, correspond à la dimension spectrale I_3 .

La représentation matricielle des données hyperspectrales est donnée par

$\mathbf{R} = [\mathbf{r}_1 \mathbf{r}_2 \dots \mathbf{r}_{M_3}]$ où M_3 est le nombre d'échantillons contenu dans l'image hyperspectrale

(i.e. le nombre total de vecteurs pixels) et est égale au produit $I_1 \cdot I_2$.

Ainsi, pour une matrice d'observation normalisée \mathbf{R} de dimension $M_3 \times I_3$, la relation (2.32) devient : $\mathbf{Y} = \mathbf{TR}$, (2.32)

Où la matrice \mathbf{Y} est la matrice réduite de taille $P \times M_3$; chaque ligne de \mathbf{Y} étant une composante spectrale dans l'espace transformé.

Les techniques de réduction de la dimension impliquent une concentration de l'information sur un nombre de composantes spectrales réduit et par conséquent une perte d'information. La nature de l'information valorisée et perdue dépend de la méthode de RD utilisée. La section suivante présente les méthodes par transformation linéaire considérées.

2.6.3.4 Méthodes de réduction basées sur l'algèbre linéaire

A partir de la représentation matricielle des données hyperspectrales, la PCA est simple à mettre en œuvre. Chaque ligne y_j de la matrice réduite \mathbf{YPC} contient l'information relative à la $j^{\text{ième}}$ composante principale et peut être re-transformée au format de l'image de taille $I_1 \times I_2$. La figure 2.13 illustre les 15 premières composantes principales des données HSI02_scene3. [NR08] Même si la PCA est *optimale*, au sens des moindres carrés, en compression de données hyperspectrales, dans un contexte d'extraction de composantes, la PCA a ses limites. En effet, les directions de projection maximisées par la PCA ne sont pas forcément les plus appropriées pour discriminer les différentes classes. En valorisant la variance, la PCA caractérise mal les classes peu représentées dans la scène. La section suivante introduit l'ICA qui, grâce à un autre critère de projection, tente de limiter le défaut de la PCA. ICA permet de trouver des directions où les composantes sont « les plus indépendantes possibles ». Ces directions caractérisent mieux les relations entre les composantes qui sont en réalité rarement orthogonales. Les directions estimées sont les moins gaussiennes possibles donc les plus asymétriques, ce qui est une caractéristique intéressante pour la classification. Mais l'application d'ICA à l'imagerie hyperspectrale implique quelques précautions. PP poursuite de projection : Cette méthode par déflation est avantageuse pour l'analyse des données hyperspectrales à grande dimension, car elle permet de réduire l'effet de HUGHES 5. Les estimations statistiques effectuées dans le sous-espace de projection sont de ce fait plus robustes. Ainsi, l'utilisation de PP pour l'analyse des données hyperspectrales ont été introduites par JIMENEZ et LANDGREBE. Ils ont proposé une méthode supervisée de RD basée sur la PP pour améliorer le taux de bonne classification. L'indice de projection est alors la distance de Bhattacharyya, calculée entre les classes définies par apprentissage au préalable. Tandis que ont mis en avant l'intérêt de l'information de divergence comme

indicede projection pour dévier les composantes ayant une distribution gaussienne. Son utilisationen tant que méthode non-supervisée de détection d'anomalies a également été proposée avec la mesure des cumulants *skurtosis* et *skewness*. Ces dernières mesures sont proches de la méthode ICA. HYVÄRINEN a proposé dans l'algorithmeFastICA, deux stratégies pour contraindre l'orthogonalisation à chaque itération, ceci évitant que les colonnes de W ne convergent vers la même source. La première, par approche « symétrique », orthogonalisetoutes les colonnes simultanément et identifie une méthodeglobale. La deuxième estime chaque colonne itérativement en ajoutant simplementune orthogonalisation de Gram-Schmidt avec les autres colonnes avant de normaliser. Cettéméthode est l'approche « par déflation » d'ICA. Ainsi, les mesures de non-gaussianité proposéespar HYVÄRINEN dans FastICA [A.H&E.O00] peuvent être utilisées en poursuite par projection.La figure 2.13 illustre les 15 premières composantes indépendantes estimées par PP (avec RD au préalable égale à 30).

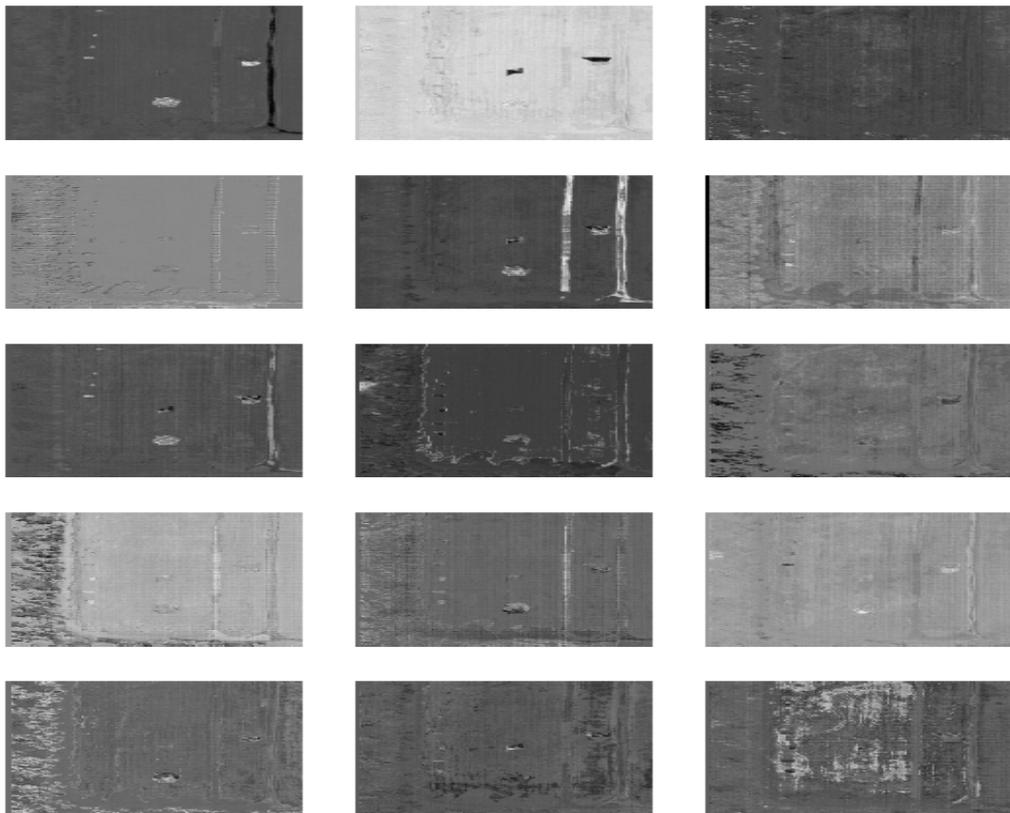


Figure 2.13 Illustration des 15 premières composantes indépendantes obtenues en utilisant la méthode de PP, avec réduction de dimension au préalable ($\mathcal{A}E$ 30). Données utilisées HSI02_scene3[NR08]

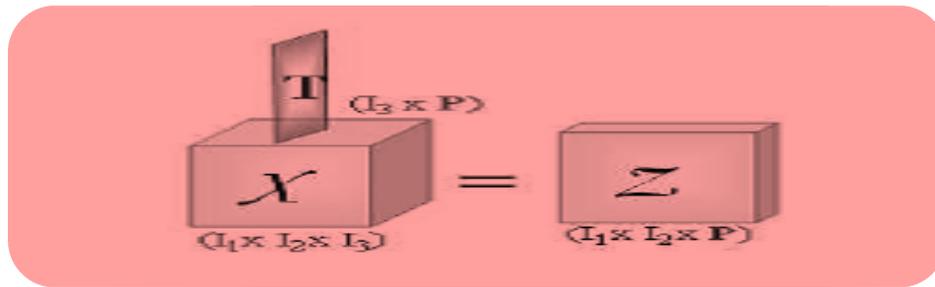


Figure 2.14 Schématisation de la réduction de dimension 3-modale d'un tenseur d'ordre 3 par transformation linéaire.

1 Formulation tensorielle des transformations spectrales linéaires

L'étude précédente a mis en évidence l'équivalence entre la représentation matricielle 6 ducube hyperspectral, et la dépliante 3-mode du tenseur hyperspectral. Dans cette section, nous récrivons, en formulation tensorielle [L.L00], le principe de la RD par transformation linéaire, définie à la relation 2.32 Cette nouvelle écriture inclut la re-transformation de chacun des P vecteurs lignes, qui compose la matrice réduite Y , au format de l'image spectrale de taille $I_1 \times I_2$. Le tenseur réduit Y d'ordre 3 $\in \mathbb{R}^{I_1 \times I_2 \times P}$ contenant les P composantes transformées, s'écrit : $Y = R \times_3 T$, (2.33)

De la même façon, on peut réécrire la PCA telle que : $Y_{PC} = R \times_3 U_P^{(3)T}$ (2.34)

Où Y_{PC} est le tenseur d'ordre 3 contenant les P composantes principales, et où la matrice $U_P^{(3)}$ est constituée des P premiers vecteurs propres de $\hat{f}^{(3)}$, la matrice de covariance dans le 3-mode. Enfin, l'équivalent de la PP selon une telle écriture est définie de sorte que :

$$Y_P = R \times_3 W, (2.35)$$

Où Y_P est le tenseur d'ordre 3 contenant les P composantes indépendantes estimées selon la méthode par déflation décrite précédemment. L'étape de blanchiment, nécessaire en prétraitement de la méthode PP, s'écrit dans le 3-mode :

$$R^{cr} = R \times_3 A^{-1/2} U_d^{(3)} (2.36)$$

La figure 2.14 schématise la transformation linéaire appliquée dans l'espace 3-modal qui réduit le tenseur d'ordre 3 R en un tenseur réduit Y . Cette illustration et les relations 2.33 à

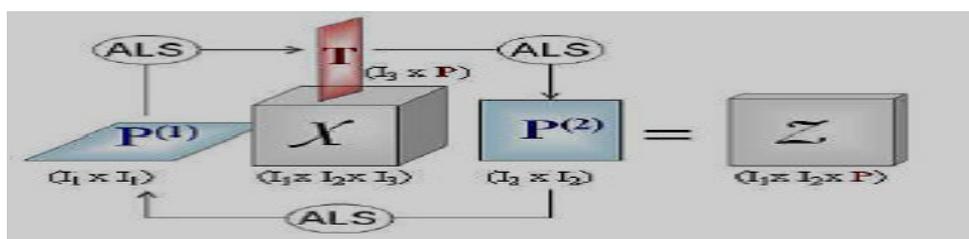


Figure 2.15 Schématisation de la méthode de réduction multimodale proposée d'un tenseur d'ordre 3

2 Objectif : une analyse multimodale

Les méthodes de réduction par transformation linéaire, PCA, ICA et PP, sont des méthodes basées sur l'algèbre linéaire qui requièrent un réarrangement des données d'observations comme illustré à la figure 2.5. Dans ce chapitre, on adopte la même approche qu'à l'étude précédente pour répondre au problème de la grande dimension des données hyperspectrales. En effet, la représentation tensorielle des données est adoptée dans le but de proposer des outils d'algèbre multilinéaire qui aboutissent à une analyse conjointe et simultanée dans les espaces spatiaux et dans l'espace spectral. Les méthodes multimodales proposées étendent les méthodes de réduction 3-modales présentées précédemment. Nous considérons, comme à l'étude précédente, une approche par décomposition tensorielle, et la décomposition de TUCKER est à la base de nos méthodes.

L'objectif de nos méthodes multimodales est d'approximer spatialement le tenseur d'ordre 3 en rangs (K_1, K_2) inférieurs (avec $K_{1,2} < I_{1,2}$), tout en extrayant un nombre réduit de composantes spectrales ($P < I_3$). L'approximation en rangs inférieurs étant effectuée sur les 1,2-mode, on peut parler d'une approximation de TUCKER2 (2 modes sur 3 sont approchés). Ce traitement est assimilé à une PCA 1,2-modale. Joint à une réduction par transformation linéaire dans le 3-mode, le tenseur d'ordre 3 résultant est alors compressé sur tous les n modes.

Pour cela, sont estimés conjointement :

1. Deux projecteurs spatiaux, notés $\mathbf{P}_K^{(1)}$ et $\mathbf{P}_K^{(2)}$, qui projettent les données dans un sous-espace orthogonal de dimension réduite égale à $K_{1,2}$.
2. Une matrice de transformation \mathbf{T} qui extrait P composantes spectrales.

La figure 2.15 schématise le principe de cette réduction multimodale. La section suivante décrit la méthode.

2.6.3.5 Méthodes de réduction basées sur l'algèbre multilinéaire

Dans cette section, nous présentons les méthodes multimodales proposées, basées sur le modèle de TUCKER qui a été présenté à dont l'algorithme ALS, TUCKALS3. Les méthodes proposées se distinguent par la transformation linéaire appliquée dans le 3-mode :

1. Pour la première, il s'agit de l'extension de la PCA. Nous avons observé à l'étude précédente, que la LRTA- (K_1, \dots, K_N) est assimilée à une PCA multimodale. Par

extension, nous nommerons la méthode de réduction, basée sur l'approximation tensorielle, la $LRTA_{dr}(K_1, K_2, P)$.

2. Pour la seconde, il s'agit de considérer la méthode de RD spectrale PP. Cette extension est appelée l'hybride TUCKER2/PP $_{dr}(K_1, K_2, P)$. Elle permet de généraliser l'approche en considérant que l'utilisateur aura le choix de l'indice de projection pour la transformation dans le 3-mode.

1 Solution par approximation tensorielle, $LRTA_{dr}(K_1, K_2, P)$

La méthode de RD multimodale a donc deux objectifs :

1. Estimer la matrice de projection \mathbf{T} en tenant compte de l'information spatiale ;
2. effectuer un traitement spatial/spectral simultanément dans le but de décorréler les données dans tous les espaces n -modaux.

À l'étude précédente, nous avons introduit la décomposition de TUCKER d'un tenseur d'ordre N . Le tenseur d'observation d'ordre 3 se décompose alors en tenseur « noyau » d'ordre 3 $\zeta \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, et en 3 matrices n -modales orthogonales $\mathbf{U}^{(n)} \in \mathbb{R}^{I_1 \times I_2}$, telle que :

$$\mathbf{R} = \zeta \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}.$$

Nous avons établi que l'approximation tensorielle $LRTA(K_1, K_2, K_3)$ estime les éléments de cette décomposition en minimisant l'erreur quadratique entre le tenseur initial et le tenseur approché. [NR08] La présentation de la $LRTA(K_1, \dots, K_N)$ a montré que minimiser l'erreur quadratique revient à maximiser la norme de Frobenius du tenseur « noyau » du tenseur approché. L'expression de ce tenseur « noyau », noté D , en fonction du tenseur d'observation et des matrices orthogonales à estimer est telle que la construction du tenseur approché, Y par la $LRTA(K_1, K_2, K_3)$ est réalisée en projetant, selon tous les modes du tenseur d'observation, une matrice de projection $\mathbf{P}^{(n)}$ définie à partir des matrices orthogonales $\mathbf{U}^{(n)}$ estimées. Le tenseur approximé et réduit spectralement par la $LRTA_{dr}(K_1, K_2, P)$ est similaire, sauf que P composantes principales sont extraites dans le 3-mode. A la différence du tenseur approché $Y \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, la construction du tenseur $Y_{PC}(K_1, K_2, P) \in \mathbb{R}^{I_1 \times I_2 \times P}$ est alors définie telle que :

$$Y_{PC}(K_1, K_2, P) = \mathbf{R} \times_1 \mathbf{P}^{(1)}_{K_1} \times_2 \mathbf{P}^{(2)}_{K_2} \times_3 \mathbf{U}^{(3)T}_P,$$

Où la matrice $\mathbf{P}^{(n)}_{K_n} = \mathbf{U}^{(n)}_{K_n} \mathbf{U}^{(n)T}_{K_n}$ définit le projecteur dans le n -mode, $n = 1, 2$; avec $\mathbf{U}^{(n)}_{K_n}$ contenant les K_n vecteurs propres associés aux K_n plus grandes valeurs propres de $\hat{\mathbf{f}}^{(n)}$, la matrice de covariance n -modale. L'estimation conjointe des matrices orthogonales $\mathbf{U}^{(n)}$ minimisant l'erreur quadratique est un problème non linéaire résolu par un algorithme des

moindres carrés alternés, ALS. Cette méthode multimodale de RD est décrite par l'algorithme 1 [NR08].

Algorithme 1: $LRTA_{dr} (K_1, K_2, P)$

entrée : le tenseur d'observation $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$; les valeurs de K_1, K_2, p
 sortie : le tenseur réduit $\mathcal{Y}_{PC-(K_1, K_2, P)} \in \mathbb{R}^{I_1 \times I_2 \times P}$
 $j = 0,$
 ① INITIALISATION DE $\mathbf{U}^{(n)}$ PAR LA HOSVD :
 pour tous $n = 1$ à 2 faire
 $\mathbf{R}_n \leftarrow$ dépliage de \mathcal{R} selon le mode n
 EVD de la matrice de covariance n -modale, $\tilde{\Gamma}^{(n)}$
 $\mathbf{U}^{(n),0} \leftarrow K_n$ vecteurs propres associés aux K_n ou P plus grandes valeurs propres
 fin
 ② ALS :
 répéter
 pour tous $n = 1$ à 3 faire
 $\mathcal{Z}^{(n),j} \leftarrow \mathcal{R} \times_q \mathbf{U}^{(q),j+1^T} \times_r \mathbf{U}^{(r),j^T}$, avec $q < n < r$
 EVD de $\mathbb{E}[\mathbf{Z}_n^{(n),j} \mathbf{Z}_n^{(n),j^T}]$
 $\mathbf{U}^{(n),j+1} \leftarrow K_n$ ou P vecteurs propres associés aux K_n ou P plus grandes valeurs propres
 fin
 $j \leftarrow j + 1$
 jusqu'à convergence de $\|\mathcal{Z}^j\|_{max}^2$
 ③ CONSTRUCTION DU TENSEUR APPROCHÉ ET RÉDUIT :
 pour tous $n = 1, 2$ faire
 $\mathbf{P}^{(n)} \leftarrow \mathbf{U}^{(n)} \mathbf{U}^{(n)^T}$
 fin
 $\mathcal{Y}_{PC-(K_1, K_2, P)} \leftarrow \mathcal{R} \times_1 \mathbf{P}^{(1)} \times_2 \mathbf{P}^{(2)} \times_3 \mathbf{U}^{(3)^T}$

2Etude d'un cas simple sur des données synthétiques

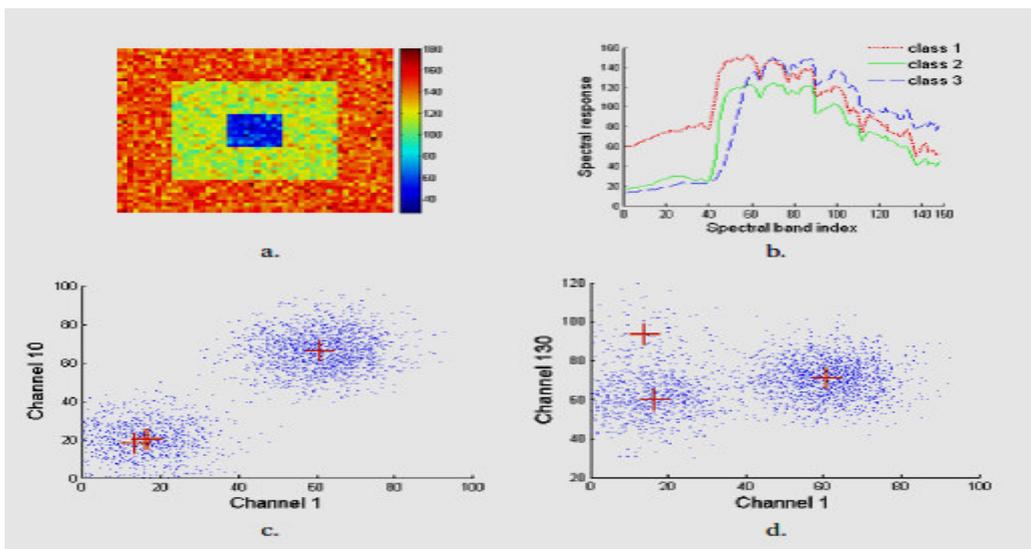


Figure 2.16 a. Tenseur d'ordre 3 simulé $2 R^{50 \times 50 \times 150}$, b. diagramme de dispersion entre les bandes spectrales c. rapprochées et d. éloignées. [NR08]

3 Evaluation sur les données réelles

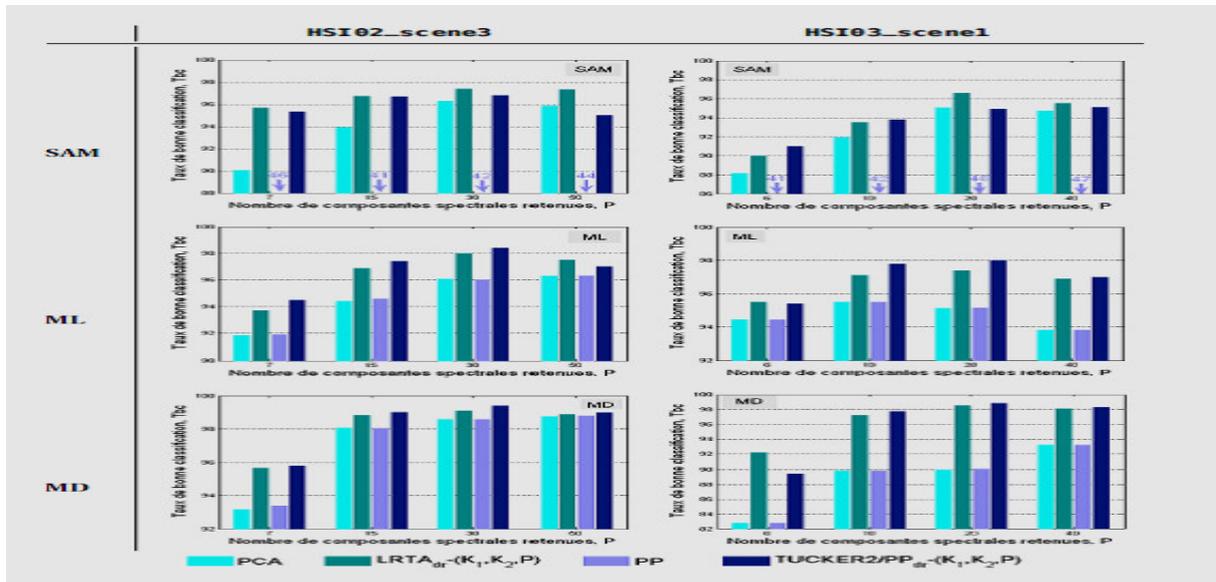


Figure 2.17 Résultats de classification en fonction de la méthode de réduction de dimension spectrale utilisée pour plusieurs valeurs de P. Données utilisées : HSI02_scene3 (gauche) et HSI03_scene1 (droite). [NR08]

2.7 Débruitage et réduction dimensionnelle à l'aide d'outils multilinéaires pour images hyper-spectrales

Nadine Renard, Salah Bourennane et Jacques Blanc Talon [NR&JB-T08]. en Avril 2008 ont réalisés le débruitage et réduction dimensionnelle à l'aide d'outils multilinéaires pour images hyper-spectrales et pondérées cette étude se base sur le principe suivant :

En se basant sur la décomposition de Tucker3 on montre la capacité de rang inférieur (K_1, K_2, D_3) On propose une nouvelle méthode, appelée $LRTA_{dr}(K_1, K_2, D_3)$ qui effectue à la fois une approximation de rang spatiale inférieure et la réduction de dimension spectrale. L'objectif principal est de présenter ces deux méthodes et de montrer comment elles fonctionnent dans des expériences de classification. On considère un tableau à trois dimensions comme un tenseur du troisième ordre, les entrées qui sont accessibles via trois indices. Il est désigné par $X \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, avec des éléments disposés de manière $x_{i_1 i_2 i_3}$, $i_1 = 1, \dots, I_1$; $i_2 = 1, \dots, I_2$; $i_3 = 1, \dots, I_3$ et R est le vrai collecteur. Chaque index est appelée voie ou le mode et le nombre de niveaux sur un seul mode est appelé dimension de ce mode. Les vecteurs n-mode sont les vecteurs en dimensions obtenues à partir d'un tenseur par indice variant en conservant les autres indices fixes. Le soi-disant n-Mode aplatie X_n de matrice X est telle que ses colonnes sont les vecteurs n-mode. L'analyse d'IHS qui est un ensemble d'images I_3 avec la taille $I_1 \times I_2$

résultant de la confluence de multiples modes est un problème d'algèbre multilinéaire. Dans ce cadre mathématique, les données IHS peuvent être représentées comme un tableau à trois voies ou comme un tenseur du troisième ordre : deux voies lignes et colonnes et un moyen pour la bande spectrale. X est appelé un tenseur de rang (K_1, K_2, K_3) si $\text{Rang}_n(X) = K_n$ pour tout $n = 1, 2, 3$. Cette représentation à plusieurs voies considère les données IHS comme un ensemble de données qui implique un traitement commun le long de chaque mode sans hypothèse de séparabilité, plutôt que de diviser les données ou seulement le traitement des images vectorielles. Cette représentation suppose intuitivement des techniques de traitement sur la base de l'algèbre multilinéaire. La décomposition Tucker3 permet d'atteindre LRTA (K_1, K_2, K_3) .

2.7.1 Outil de réduction de dimensionnalité LRTA_{dr} - (K_1, K_2, D_3)

Dans le contexte IHS, la réduction du nombre de bandes spectrales par sélection des caractéristiques spectrales les plus importantes afin d'améliorer le classement.

Les principes de PCA_{dr}(D_3) sont les suivantes : les images I_3 taille $I_1 \cdot I_2$ sont considérés. Chaque image est transformée en un vecteur ligne par concaténation. Tenseur de données $R \in R^{I_1 \times I_2 \times I_3}$ composé de toutes les images I_3 devient une matrice $R \in R^{I_3 \times p}$ où $p = I_1 \times I_2$. L'objectif de la RD est d'extraire un petit nombre $D_3 < I_3$ de fonctionnalités, appelé composantes principales (PC). Par conséquent, les ordinateurs D3PCs génèrent une matrice réduite $Z \in R^{D_3 \times p}$ $Z = R \times A^{-1/2} U^T$. (2.37)

Les données peuvent être remodelés en trois voies série $Z \in R^{I_1 \times I_2 \times D_3}$. Dans la formulation du tenseur, la matrice obtenue précédemment R est équivalente matrice mode 3 aplatie de R notée R_3 . Ensuite, l'équation (2.37) peut être écrite : $Z = R \times_3 A^{-1/2} U^{(3)T}$. (2.38)

De la même manière, nous pouvons tourner la LRTA (K_1, K_2, K_3) dans un outil spectrale de réduction de la dimensionnalité. Cet outil est appelé LRTA_{dr}(K_1, K_2, D_3) dans le présent document, où D_3 représente le nombre de composantes principales spectrales. LRTA_{dr}(K_1, K_2, D_3) extraits D_3 PC spectrales pour obtenir trois voies série $Z \in R^{I_1 \times I_2 \times D_3}$. Le défi est réalisée grâce aux LRTA_{dr} - (K_1, K_2, D_3) est de réduire conjointement la dimension du mode spectral et de projeter les informations sur les modes spatiaux sur inférieur (K_1, K_2) sous-espaces de dimension. Les derniers traitements permettent de compresser et de débruiter spatialement les données. Des valeurs de paramètre différentes - (K_1, K_2, D_3) - peuvent être conservées pour chaque mode. Le modèle LRTA_{dr}(K_1, K_2, D_3) :

$$Z = R \times_1 P^{(1)} \times_2 P^{(2)} \times_3 A^{-1/2} U^{(3)T}, \quad (2.39)$$

Où $U^{(3)}$ est une matrice tenant D_3 vecteurs propres sélectionnés, Λ est la matrice diagonale des valeurs propres en maintenant les plus grandes valeurs propres D_3 et $P^{(n)}$ est le projecteur pour le mode n , définie dans la section III ci-dessus.

Le principal $LRTA_{dr}(K_1, K_2, D_3)$ attribue par rapport à la $PCA_{dr}(D_3)$ est double. Tout d'abord, $LRTA_{dr}(K_1, K_2, D_3)$ commune utilise des informations spatiales et spectrales pour extraire les composantes spectrales principales. Deuxièmement, $LRTA_{dr}(K_1, K_2, D_3)$ débruite les principales composantes spectrales extraites grâce aux projections spatiales estimées, $P^{(n)}_{n=1,2}$. Les itératives SLA algorithmes permettent de prendre en compte la crossdependence entre **NR** et **DR**. Une itération SLA nécessite approximativement 6 s. sur un 3.2 GHz PC fonctionnant sous Windows et utilisant Matlab[®]7. Cinq itérations donnent des résultats satisfaisants. Les données utilisées dans les expériences suivantes sont des données réelles recueillies par imagerie HYDICE, avec un 1,5 m spatiale et une résolution spectrale de 10 nm et dont 148 bandes spectrales (435-2326 nm), 310 lignes et 220 colonnes.

Dans cette expérience, on teste la robustesse au bruit de la procédure $LRTA_{dr}(K_1, K_2, D_3)$. Pour cette question, un bruit gaussien stationnaire et à zéro moyen est ajouté aux données IHV du monde réel avec un écart type variant de 5 à 25. Pour plus de commodité, les résultats OA ne sont obtenus à partir de $D_3 = 30$ composantes spectrales conservées grâce aux méthodes de **DR**. Le tableau II montre que les valeurs obtenues à partir de l'arthrose PCADR résultent de diminuer lorsque l'écart-type augmente. Le 2D - filtrage (Wiener ou SWT) appliqués sur les composants spectraux permet d'améliorer l'OA-valeur. La $LRTA_{dr}(40,40,30)$ atteint le même effet en dégageant des résultats de classification qui sont robustes à l'affaiblissement du bruit et sensiblement mieux que le PCA_{dr} -Wiener et PCA_{dr} -

Tableau 2.4 Précision globale du SAM en environnement bruyant. Résultats obtenus à partir de 30 des composantes spectrales conservées. [NR&JB-T08]

Standard deviation	PCA_{dr}	PCA_{dr} - Wiener	PCA_{dr} - SWT	$LRTA_{dr}(40,40,30)$
5	94,9	97,6	97,6	97,55
15	90,7	96,78	97,14	97,58
20	80	96,9	97,12	97,5
25	75.5	96,6	97,09	97.5

2.8 Conclusion

Dans ce chapitre, nous avons présenté le principe d'acquisition des données hyperspectrales. Du fait de l'importance du nombre de bandes acquises (en général plus de 200) et du caractère contigu des bandes spectrales (résolution spectrale de 10 nm), les applications de détection de cibles, classification en zones géographiques et démixage révèlent le potentiel discriminant de telles données. Mais la nature même des données hyperspectrales entraîne des contraintes qui perturbent l'efficacité des algorithmes. Nous avons introduit dans ce chapitre, deux problématiques directement liées aux données hyperspectrales, qui sont :

1. Le bruit dû aux processus d'acquisition ;
2. La grande dimension des données.

L'état de l'art des méthodes qui répondent à ces deux problématiques révèle que l'analyse des données hyperspectrales est fortement dépendante de la façon dont elles sont considérées.

Est-ce une concaténation d'images ou une concaténation de spectres ?

Ce choix implique l'analyse des données dans un espace donné. Lorsque les relations spatiales sont prises en compte, le traitement d'image est valorisé ; lorsque l'information spectrale est privilégiée, le traitement du signal est considéré.

D'après l'étude bibliographique, des méthodes originales qui répondent aux deux problématiques respectivement sont proposées. Notre choix s'oriente vers une considération des données hyperspectrales dans leur ensemble. Par conséquent, les données seront représentées par un tenseur d'ordre 3, avec la définition d'un mode spectral et de deux modes spatiaux. Cette nouvelle représentation va impliquer le développement d'outils basés sur l'algèbre multilinéaire. Une étude effectuée nous a permis d'introduire les définitions et les propriétés du traitement du signal tensoriel. Dans cette étude, nous avons donné la définition d'un *tenseur* qui est utilisée dans la suite pour proposer une nouvelle représentation des données hyperspectrales. Dans un deuxième temps, nous avons présenté les propriétés des principaux opérateurs tensoriels. Nous avons mis en évidence le fait que la généralisation des outils bilinéaires aux outils multilinéaires n'est pas triviale. L'une des principales distinctions est la non extension du théorème d'ECKART-YOUNG [Y.E&G.Y36] au cas tensoriel (tenseur d'ordre > 2). Il en résulte deux définitions de rangs distincts pour un tenseur :

1. *Def1* : Le rang n -modal K est le rang multilinéaire du tenseur A d'ordre N noté (K_1, \dots, K_N) . Ce rang généralise le rang colonnes et lignes d'une matrice pour chaque dépliante n -modale.
2. *Def2* : Le rang tensoriel K . Ce rang généralise le rang associé au produit extérieur d'une matrice.

Ces deux définitions de rangs sont à la base de deux décompositions tensorielles qui leur sont associées et qui généralisent la décomposition en valeurs singulières d'une matrice. Nous avons ainsi introduit deux décompositions tensorielles :

1. *Def1* : Le modèle de TUCKER3 qui généralise la représentation matricielle de la SVD. Ce modèle permet l'extension de la décomposition en valeurs singulières à l'ordre supérieur, nommée HOSVD. Le modèle de TUCKER3 est aussi à l'origine de l'approximation d'un tenseur en rangs (K_1, \dots, K_N) inférieure, notée la LR-TA- (K_1, \dots, K_N) .
2. *Def2* La décomposition tensorielle PARAFAC/CANDECOMP qui généralise la représentation canonique de la SVD. Ce modèle décompose un tenseur en une somme de tenseur de rang un ; il permet l'analyse en composantes principales multimodale.

Plusieurs autres décompositions tensorielles ont été proposées comme étant des variantes ou des extensions de ces deux modèles. Ces propositions croissantes montrent l'intérêt qui est porté au traitement tensoriel et qui trouve sa place dans des domaines d'application très variés.

Dans la suite du mémoire, nous proposons de représenter les données hyperspectrales par un tenseur d'ordre 3. L'ensemble des méthodes tensorielles développées sont basées sur le modèle de TUCKER3. En effet, ce travail s'inscrit dans la continuité des travaux de D. MUTI [D.M04] qui a proposé le filtre de Wiener multimodal basé sur TUCKER3 pour le débruitage des signaux sismiques et des images couleurs. De plus le modèle de TUCKER3 trouve une sémantique plus intuitive à l'application des images hyperspectrales que le modèle de

PARAFAC/CANDECOMP. En effet, les études précédentes montrent l'importance de la flexibilité des valeurs des rangs (K_1, K_2, K_3) qui sont dissociées dans chaque n -mode alors que PARAFAC/CANDECOMP est associé à une valeur unique du rang tensoriel. De plus, l'orthogonalité des matrices n -modales rend le modèle de TUCKER3 plus facilement interprétable. Cependant, les autres décompositions tensorielles, en les adaptant aux données hyperspectrales, proposent une perspective intéressante pour l'extension des méthodes présentées. Dans ce chapitre, nous avons présenté l'extension des techniques de réduction de dimension spectrale par transformation linéaire au cas multimodal. Avec la même stratégie que l'étude précédente, la représentation tensorielle des données hyperspectrales permet l'utilisation d'outils basés sur l'algèbre multilinéaire. Basées sur l'approximation tensorielle de TUCKER, les méthodes multimodales de réduction proposées analysent simultanément tous les modes [NR08]. L'objectif est double :

1. Compresser l'espace spatial en projetant les données dans un sous-espace orthogonal de rangs- $(K_{1,2})$ inférieurs. Ce traitement est assimilé à la décomposition de TUCKER2 dans les 1,2-modes, et nécessite l'estimation de matrices de projection spatiale, $\mathbf{P}_K^{(1,2)}$.
2. Réduire la dimension de l'espace spectral par une transformation linéaire des données dans le 3-mode. Ce traitement nécessite l'estimation d'une matrice de transformation, $\mathbf{T}^{(3)}$.

L'algorithme des moindres carrés alternés (ALS) estime conjointement les matrices $\mathbf{P}_K^{(1,2)}$ et $\mathbf{T}^{(3)}$. Ainsi, il garantit l'inter-dépendance de l'analyse des modes. Dans cette étude, deux méthodes multimodales sont développées qui se distinguent par la transformation linéaire utilisée pour réduire la dimension spectrale.

3. La première, utilisant la PCA, est alors assimilée à une PCA multimodale et fait référence à l'approximation en tenseur de rangs inférieurs. Par extension, cette méthode est notée $\text{LRTA}_{dr}-(K_1, K_2, P)$, l'indice « *dr* » indiquant que le 3-mode est de dimension réduite et égale à P .
4. La deuxième est basée sur une réduction par PP et est notée $\text{TUCKER2/PP}_{dr}-(K_1, K_2, P)$. L'indice de projection sélectionné est ici une mesure de non-gaussianité.

Pour valider les méthodes proposées, les résultats de classification ont été quantifiés sur des données synthétiques et des données hyperspectrales réelles HYDICE. Trois classifieurs ont été sélectionnés pour cette investigation : le classifieur SAM, ML et MD. On a évalué les méthodes de réduction spatiale/spectrale, dites multimodales, et les méthodes de réduction spectrale dites 3-modale. Pour cela, les résultats de classification issus de ces différentes méthodes de réduction sont comparés. Les méthodes comparatives sont les méthodes basées sur la même technique de réduction spectrale : PCA et la PP ; et une méthode adaptée au bruit : la MNF. Les résultats attestent de l'intérêt des méthodes de réduction multimodales, permettant de diminuer le nombre de pixels non-classés ainsi que les pixels isolés mal classés. Le résultat aboutit à des classes plus homogènes, du fait que la variabilité spectrale ait été réduite par la projection de l'espace spatial dans un sous-espace réduit. Ce constat est établi au même titre que la bonne classification des classes peu représentées dans la scène.

Le développement de la méthode multimodale de réduction, $\text{TUCKER2/PP}_{dr}-(K_1, K_2, P)$, basée sur la transformation par PP, permettra dans les travaux à venir de définir d'autres indices de projection et donc de faire une étude plus rigoureuse dans le but de sélectionner celui qui sera le plus adapté au contexte de classification.

Chapitre 2

Etude du SRV et réduction de ticket

Introduction

Nous nous pencherons sur les étapes pour la reconnaissance du visage. Il s'agit d'une amélioration de l'identification des personnes et peut être résumé selon les phases existantes dans tout SRV. Rappelons que notre travail dans ce présent projet de fin d'étude se fera surtout sur la partie de réduction de données. Nous nous limiterons à l'étude de la décomposition de TUCKER3. Cette dernière généralise l'analyse en composantes principales pour l'approximation d'un tenseur en rangs- (K_1, \dots, K_M) inférieurs.

Système de Reconnaissance de Visage

La figure 3.1 nous montre les différentes étapes du système SRV.

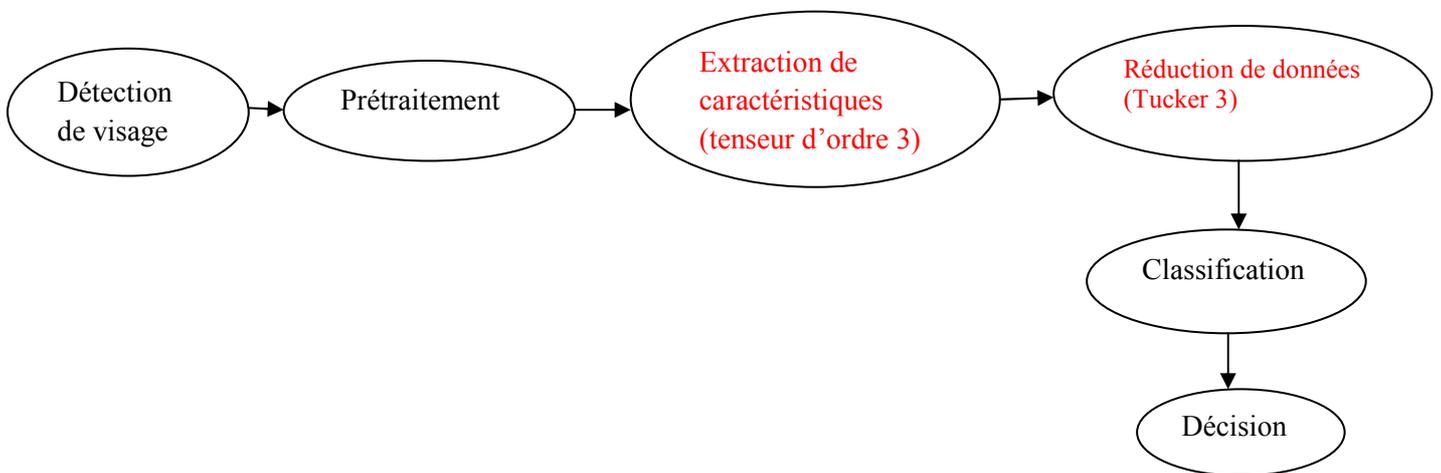


Figure 3.1 Système simplifié – reconnaissance de visages (technique de Tucker3)

3.1 Détection de visage

Comme nous l'avons précédemment précisé, la détection de visage est réalisée afin de nous assurer que les blobs que nous détectons dans l'image correspondent à des humains. Les systèmes de détection de visage actuels peuvent être classés selon qu'ils se basent sur le visage entier ou sur des traits caractéristiques du visage. Dans la première approche, on génère une base d'exemples à partir de laquelle un classifieur va apprendre ce qu'est un visage (réseaux de neurones, machines à vecteurs supports (SVM), analyse en composantes principales, eigenfaces, *etc.*). Ces systèmes peuvent obtenir des scores de détection élevés mais requièrent de grandes bases d'apprentissage. Dans la seconde approche, on peut distinguer trois niveaux d'analyse. Au niveau le plus rudimentaire, on ne prend en compte que la couleur ou les niveaux de gris pour détecter des régions ressemblant à un visage,

généralement de face. A un niveau d'analyse intermédiaire, on cherche à détecter des caractéristiques indépendantes des conditions lumineuses et de l'orientation des visages.

Enfin, à un haut niveau d'analyse, on recherche des traits caractéristiques du visage tels que les yeux, les contours extérieurs, le nez et la bouche que l'on associe à des configurations (« *templates* ») connues *a priori* ou apprises. Des modèles déformables, des snakes ou des modèles de points distribués (Point Distributed Models PDM) peuvent alors être utilisés. Ces derniers modèles requièrent notamment une bonne résolution de l'image. Des exemples de détections de visages dans des images sont illustrés par la **Figure 3.2**. Quelque soit l'approche utilisée, détecter les visages dans des images grand public est une tâche de haut niveau qui doit faire face à des difficultés liées à divers facteurs, que nous nous proposons d'expliquer.

[L.H11]



Figure 3.2 Exemples de détection de visages dans des images. [L.H11]

3.1.1 Les difficultés liées à la détection de visage

Les difficultés liées à la détection de visage peuvent être attribuées aux facteurs suivants :

- **Posture.** L'image d'un visage change en raison de la position relative caméra-visage (de face, de profil ou dans une position intermédiaire), et certains attributs faciaux tels que les yeux ou le nez peuvent être partiellement ou complètement occultés. [L.H11]
- **Présence ou absence de composantes structurelles.** Les attributs faciaux tels que la barbe, la moustache et ou des lunettes peuvent être présents ou pas et cela avec une forte variabilité. De plus, ces attributs peuvent eux-mêmes revêtir des formes très différentes d'un individu à l'autre : géométrie, couleur, taille, *etc.*
- **Expression faciale.** L'expression faciale d'une personne affecte directement l'aspect de son visage.
- **Occultation.** Des visages peuvent être partiellement cachés par d'autres objets. Dans une image avec un groupe de personnes, certains visages peuvent partiellement ou entièrement en cacher d'autres.
- **Orientation de l'image.** Les images de visage changent directement pour différentes rotations autour de l'axe optique de la caméra.

- **Conditions de prise de vue.** Des facteurs tels que l'éclairage (distribution, orientation et intensité de la source) et les caractéristiques de la caméra (capteur, optique) affectent l'aspect d'un visage dans l'image.

Les difficultés citées ici représentent les plus gros défis pour les algorithmes de détection de visage. Les méthodes développées tendent à augmenter les performances de détection et à être assez robustes à ces dernières, c'est-à-dire à augmenter les taux de bonne détection et à diminuer les fausses détections (faux positifs et faux négatifs). Selon l'application visée (suivi d'individus, reconnaissance biométrique, *etc.*) et l'environnement d'utilisation (extérieur ou intérieur, fond de scène maîtrisé ou non, éclairage variant ou invariant, *etc.*), certains écueils peuvent être directement évités. Il convient donc, avant de développer une méthode de détection de visage, de savoir pour quelle application et dans quel environnement celle-ci sera utilisée. De la sorte, son développement pourra être orienté pour pallier telle ou telle difficulté. Un autre facteur qui influence directement la manière dont un algorithme de détection de visage est développé et utilisé est lié aux contraintes associées au temps de réponse requis pour l'application (nécessité éventuelle d'une réaction en « temps réel »).

Certains algorithmes peuvent être assez robustes mais lents en exécution, et d'autres peuvent être très rapides mais moins robustes. Le choix de la méthode à utiliser doit donc se faire principalement en fonction de l'environnement et du cahier des charges de l'application visée. [L.H11]

3.1.2 Les méthodes de détection de visage

Les méthodes de détection de visage peuvent être classées en quatre catégories [YanKM02], à savoir :

- *les méthodes basées sur la connaissance* : Dans cette catégorie, les méthodes de détection de visage sont développées à partir de règles dérivées de la connaissance que l'on a des visages humains [L.H11].
- *les méthodes basées sur des caractéristiques invariantes* : En réponse aux « points faibles » de l'approche précédente, on essaye ici de trouver des caractéristiques invariantes qui existent quels que soient la posture, le point de vue, et les conditions d'éclairage. On emploie ensuite ces dernières pour localiser les visages [L.H11].
- *les méthodes d'appariement de modèles* : Dans cette catégorie de méthodes, plusieurs modèles standards de visage sont appris et stockés pour décrire ce dernier dans son ensemble ou certaines caractéristiques faciales séparément [CraTB92].

- *les méthodes basées sur l'apprentissage de modèles* : Dans cette catégorie, et contrairement aux méthodes d'appariement de modèles où ceux-ci sont prédéfinis par des experts, les modèles exploités ici sont appris à partir d'exemples d'images [SchK98] .

3.2 Prétraitement

3.2.1 Prétraitement basique

Dans l'étape du prétraitement, on regroupe tous les modules qui vont modifier la représentation sources. L'objectif est de trouver une représentation qui facilite le travail des modules suivants. Deux types de transformation sont utilisés. Elles permettent souvent une première réduction des données et elles atténuent les effets dus à une différence de conditions lors des prises de vues. En regardant les images nous constatons clairement l'apparition de particularités non souhaitées au niveau du cou comme les cols de chemise, les polos, etc. Par ailleurs, les cheveux sont également une caractéristique changeante au cours du temps (changement de coupe, teinte, calvitie,...). L'arrière plan apparaît sur les images ; il ne sert à rien, et gonfle inutilement la taille des données. Enfin les oreilles posent aussi un problème. En effet, si la personne se présente légèrement différemment devant la caméra (rotation), nous ne pouvons apercevoir qu'une seule oreille. C'est pourquoi on coupe généralement l'image verticalement et horizontalement et ne garde qu'une fenêtre de dimension 132x120 centrée sur le visage en utilisant la technique inspirée de [RB&TP 93]. Ensuite les images sont filtrées par un filtre passe bas pour pouvoir effectuer une décimation de facteur 2. Ce qui réduit par 4 la taille de l'image découpée. Les images passeront ainsi d'une dimension 65536 à une dimension $66 \times 60 = 3960$ (après découpage et décimation), voir figure 3.3.



Figure 3.3 L'image avant (a) et après (b) découpage et décimation

Après, on applique une photo normalisation. Cela veut tout simplement dire que pour chaque image, nous soustrayons à chaque pixel la valeur moyenne de ceux-ci sur l'image, et que nous divisons ceux-ci par leur déviation standard. Enfin nous faisons une normalisation. La photo normalisation agit sur une image alors que la normalisation agit sur un groupe d'images (pour

chaque composante, nous retirons la moyenne de cette composante pour toutes les images et nous divisons par la déviation standard).

3.2.2 Exemple de prétraitement des bases d'apprentissage

Afin de mieux comprendre ce type de procédure nous présentons l'exemple illustré sur la figure 3.4 appliqué sur la base d'images GENKI [hmue11]. Comme précisé dans cette figure, la première étape consiste à construire l'ensemble d'apprentissage normalisé.

Les étapes suivantes du prétraitement sont illustrées dans la Figure 3.5. Elles consistent dans la détection du visage (B), la rotation et au recadrage du visage (C) en considérant la distance inter-pupillaire (IPD), la mise à l'échelle de la taille (D) et l'étalement du contraste.

Par rapport à la taille des images contenues dans la base GENKI (approximativement 180x192), avant de réaliser la détection du visage, nous augmentons artificiellement la taille des images en réalisant une interpolation linéaire pour arriver à des résolutions de l'ordre de 450x480 en préservant l'aspect de l'image.

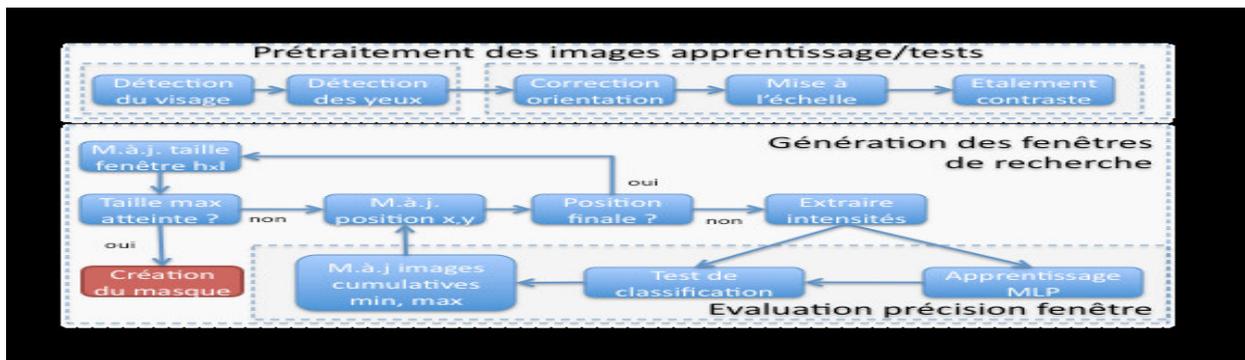


Figure 3.4 Aperçu de l'approche [hmue11]



Figure 3.5 Etapes du prétraitement pour les images. [hmue11]

La bibliothèque logicielle OpenCV est employée pour détecter et extraire le visage depuis les images initiales en utilisant un détecteur de visage en position frontale [PV,MJJ 04]. Le détecteur utilise les caractéristiques de type Haar qui considère les changements en termes de contraste observés entre groupes de pixels adjacents. Ensuite, la détection des pupilles est réalisée en utilisant un réseau de neurones implémenté dans la bibliothèque logicielle STASM. Par rapport, au visage tout entier, on définit de zones de recherche situées dans la

partie haute du visage. La position des pupilles sert à réaliser une éventuelle rotation du visage en position verticale. [hmue11]

Le roulis du visage est estimé en utilisant les positions relatives de pupilles. On calcule l'angle entre les centres des pupilles et on applique une rotation en sens inverse. La rotation s'effectue par rapport au centre du visage. Ainsi les orientations du visage jusqu'à ± 25 degrés par rapport à l'axe horizontal sont corrigées. Les visages ainsi obtenus sont utilisés dans les nombreux réseaux de neurones qui seront construits pour les fenêtres de recherches générées à l'étape suivante. [hmue11]

3.3 Extraction des caractéristiques

3.3.1 Extraction des caractéristiques par patches

Le détecteur de points caractéristiques du visage proposé utilise des patches de différentes tailles en entrée du classifieur. La taille de ces patches est un choix sensible. En effet, une taille trop petite ne prendra en compte que l'information locale (perdant donc des informations globales) et introduisant des détections fines mais ambiguës alors qu'une taille trop grande n'extraira que des informations grossières et peu précises. Pour pallier à ce problème, les auteurs dans [VR,TS et al 12] proposent une architecture pyramidale de patches multi-résolutions extrayant différents niveaux d'informations. Pour un pixel i , nous prenons un premier patch p^1_i de taille suffisamment grande pour capturer un maximum d'information globale. Les autres patches ($p^2_i, p^3_i, \dots, p^N_i$) sont quant à eux extraits en sélectionnant une zone de plus en plus petite, récupérant ainsi des informations de plus en plus locales et fines autour du point. Tous les patches sont ensuite redimensionnés à la taille du plus petit (fig. 3.6).

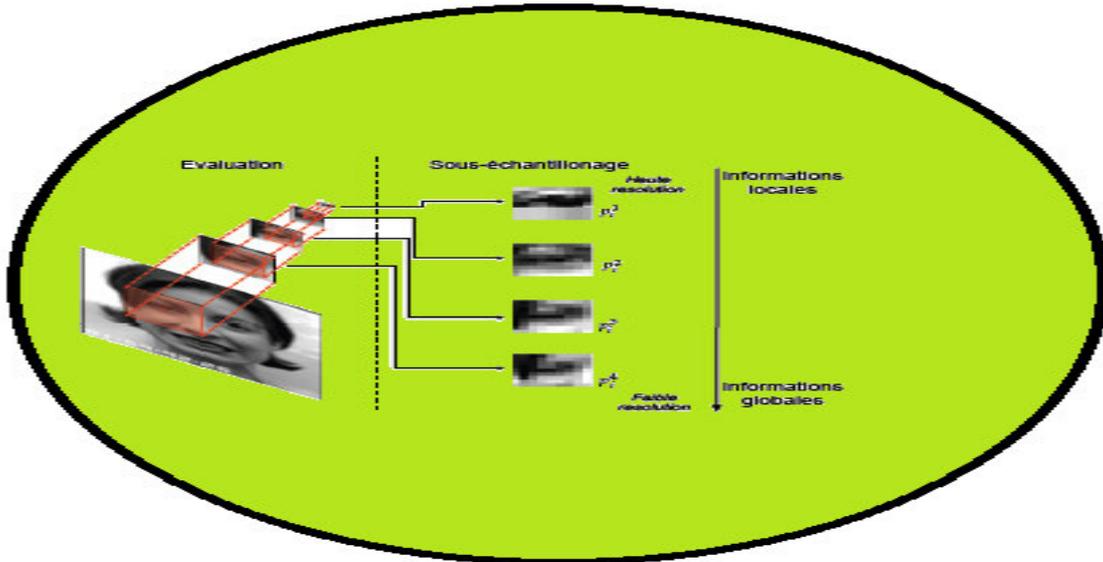


Figure 3.6 Pyramide de patches multi-résolutions pour un pixel i . A chaque étape, différents niveaux d'informations sont mis en valeur. [VR,TS et al 12]

Ainsi, ce système sera capable, grâce à ces patches de petites tailles, de récupérer les informations locales et les petits détails, comme le canthus ou la position de la pupille. Les patches de grandes tailles se focaliseront sur l'extraction d'informations globales permettant ainsi de différencier un œil droit d'un œil gauche en utilisant par exemple la forme du nez ou des oreilles. On propose ici d'utiliser un détecteur par point. Etant dans un problème de classification, l'apprentissage est réalisé, pour chacun des points, à l'aide d'exemples positifs et négatifs. Neuf patches multi-résolutions sont extraits comme exemples positifs (target) : le premier centré sur la vérité terrain et les huit autres sur les huit positions connexes. Pour les exemples négatifs (non-target) on utilise, au départ, seize patches multi-résolutions distribués uniformément autour de la vérité terrain (fig. 3.7).

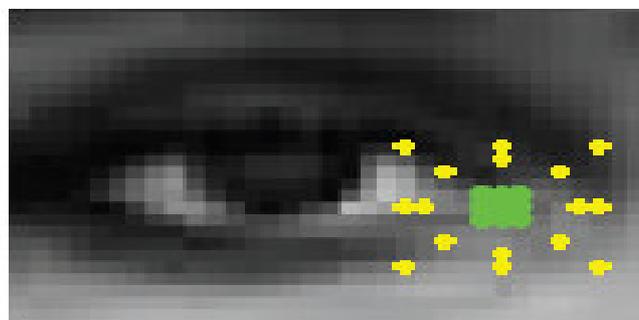


Figure 3.7 Les exemples positifs sont extraits sur les 9 points les plus proches de la vérité-terrain. Les exemples négatifs sont ici représentés par des croix. [VR,TS et al 12]

3.3.2 Extraction de la Forme du Visage

La *forme* du visage décrit l'emplacement des principales composantes du visage que sont le front, les sourcils, les yeux, le nez, la mâchoire et la bouche. Cette forme peut être donnée par la localisation de points caractéristiques (en 2 ou 3 dimensions), par des maillages ou encore par des formes géométriques simples. [CS13]

- *Points caractéristiques* : Un visage peut être décrit par un vecteur formé par les coordonnées 2D ou 3D d'un certain nombre de points caractéristiques (par exemple le coin gauche des lèvres). Dans la figure 3.8, le visage est annoté par 73 points caractéristiques en 2D. Le vecteur résultant possède donc 73×2 composantes. Des procédures mathématiques permettent d'extraire les principales déformations et de réduire la dimensionnalité de ces vecteurs. Pour ce faire, plusieurs visages sont annotés et les procédures sont appliquées sur l'ensemble des vecteurs associés. Parmi ces méthodes, nous pouvons citer *l'analyse en composantes principales (ACP)* qui permet de transformer des variables corrélées en nouvelles variables dé-corrélées les unes des autres. Ces nouvelles variables sont nommées *composantes principales*. Les axes correspondants sont orthogonaux et correspondent aux principales déformations allant de la plus grande à la plus petite. Cette technique a souvent été utilisée pour décrire des visages. Néanmoins, elle est très dépendante des données d'apprentissage et les déformations obtenues ne possèdent pas d'interprétation physique évidente. Une autre technique est *l'analyse en composantes indépendantes (ACI)* qui décompose les données en variables statistiquement indépendantes. [CS13]

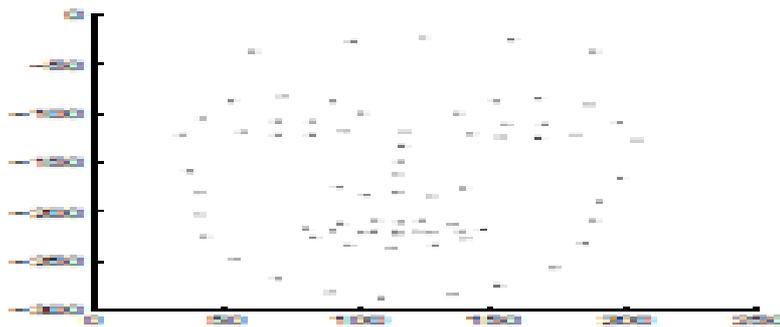


Figure 3.8 Description de la forme d'un visage par les coordonnées 2D de 73 points caractéristiques. [CS13]

Les *modèles de forme actifs (ASM)*, basés sur l'ACP, permettent de retrouver automatiquement les points caractéristiques du visage. Le modèle est appris sur un ensemble de visages annotés manuellement. Le principe de l'ASM est de faire correspondre le modèle de forme à une nouvelle image, en trouvant la transformation (recalage global puis local) permettant d'optimiser la correspondance entre le modèle et la nouvelle image. [CS13]

Les formes géométriques simples, il s'agit ici de définir la forme du visage par des équations simples, correspondant à des formes simples. Ces techniques sont appliquées pour la description et la détection de certaines parties du visage, notamment l'iris, modélisé par une ellipse ou un cercle. Pour détecter l'iris sur un nouveau visage, le modèle de l'iris créé est appliqué sur la nouvelle image et les paramètres du modèle sont modifiés de façon à correspondre au mieux au modèle. De telles méthodes sont efficaces lorsque les yeux sont ouverts mais la fermeture et le clignement des yeux posent souvent problème. De plus, ces techniques nécessitent une résolution de haut niveau pour fournir de bons résultats. Et à part quelques exceptions, elles sont rarement appliquées à des formes complexes, ne pouvant pas prendre en compte la complexité des déformations du visage. [CS13]

Les maillages et courbures, avec l'arrivée récente sur le marché des caméras de profondeur à bas coût et la définition 3D des visages, les techniques de description de surfaces 3D sont en plein essor.

Une première méthode consiste à définir la forme du visage par un **maillage**, c'est-à-dire que la surface du visage est approximée par des facettes, la plupart du temps des triangles. La figure 3.7 propose une représentation sous la forme de maillage à facettes triangulaires. Une représentation classique d'un maillage consiste à avoir deux vecteurs, le premier contenant les coordonnées 3D de très nombreux points du visage et le second contenant la liste des triangles (c'est-à-dire des points permettant de définir une facette). Contrairement aux *points caractéristiques*, les maillages utilisent un grand nombre de points n'ayant pas de localisation physique particulière. [CS13]

Une autre méthode permettant d'approximer la surface consiste à définir la **courbure** du visage en de nombreux points. La courbure en un point du visage peut être définie par le Laplacien des coordonnées des points qui sont dans son voisinage. Les coordonnées ainsi obtenues forment un vecteur 3D en chacun des points. La direction du vecteur donne l'inclinaison de la surface du visage et la norme du vecteur donne la force de la courbure. [CS13]

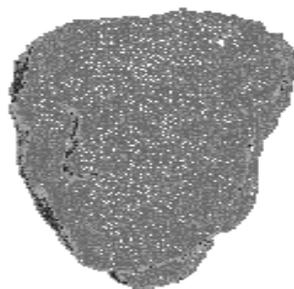


Figure 3.9 Représentation d'un visage sous la forme d'un maillage à facettes triangulaires. [CS13]

3.3.3 Extraction de la Texture du Visage

La *texture* peut tout d'abord être décrite par le niveau de gris de l'ensemble des pixels de l'image contenant le visage. Afin de limiter l'information relative à l'arrière plan, le plus souvent l'image est préalablement recentrée sur le visage et rognée de façon à ce que les bords du visage touchent (ou presque) les bords de l'image. Parfois, la forme du visage (contour) est utilisée et seules les informations de texture à l'intérieur de cette forme sont prises en compte. Ces informations sont

représentées dans un vecteur dit vecteur de texture. Comme dans le cas des points caractéristiques, des transformations (ACP, ICA) peuvent être appliquées pour extraire les principales déformations et réduire la dimensionnalité de ces vecteurs.

La texture peut aussi être codée par l'application de filtres de Gabor sur les valeurs des pixels.

C'est le cas du système proposé par Lyons et al. [ML,SA et al98] qui code les visages par l'utilisation d'un ensemble de filtres de Gabor multi-orientés et multi-résolutions. Cette représentation a pour avantage d'avoir des similitudes avec les annotations sémantiques d'observateurs humains. Une autre façon de décrire un visage consiste à définir un certain nombre de motifs et d'en extraire leur répartition dans l'image. C'est le cas de la méthode LBP [CS,SG et al05]. Cette méthode consiste à associer un motif à chaque pixel, selon la nature des pixels environnants. Un histogramme des motifs est alors créé pour une zone de l'image, indiquant la répartition de ces motifs dans cette zone. Ces informations permettent de caractériser les différentes zones du visage et de décrire ainsi les données visuelles de l'image. Des contributions plus récentes continuent d'améliorer cette technique largement employée. Ces méthodes ne permettent pas de reconstruire l'image du visage mais fournissent une signature intéressante permettant d'analyser des visages.

3.4 Réduction de données

3.4.1 Le format hiérarchique de Tucker

Nous donnons ici un bref aperçu du format de la hiérarchique Tucker (H-Tucker) et renvoyons le lecteur à [L.G10] pour une plus explication détaillée.

De plus, la multiplication par une matrice et le produit intérieur de deux tenseurs au format \mathcal{H} -Tucker peut être effectuée de manière efficace, nous renvoyons le lecteur à [D.K&C.T12] pour de détails sur la mise en œuvre. La troncature d'un tenseur x au format H-Tucker est l'approximation par un tenseur \tilde{x} avec rangs hiérarchiques inférieurs. [L.G10] Pour plus de détails nous nous référons à [L.G10,D.K&C.T12]. Les expérimentations numériques sont basés sur la disposition du public htucker boîte à outils [D.K&C.T12] dans Matlab qui fournit une représentation d'un tenseur Format H-Tucker, ainsi que les opérations utiles.

3.5 Classification

3.5.1 Classification à base de SVM

Afin de procéder à la reconnaissance des visages, un SVM est utilisé. Pour l'apprentissage d'une SVM, toutes les images de la BDD d'apprentissage désirée sont utilisées.

- *SVM multi-noyaux* : Etant donné un exemple d'apprentissage composé d'histogrammes LGBP et d'un vecteur de coefficients d'AAM, $x_i = (H_i; C_i)$, associé

aux étiquettes y_i (exemple positif ou négatif), la fonction de classification de la SVM associe un score s à un nouvel exemple $x = (H; C)$: $s = (\sum_{i=1}^m \mathbf{a}_i k(x_i, x) + \mathbf{b})$ (3.1)

avec a_i la représentation duale du vecteur orthogonal à l'hyperplan k étant la fonction noyau réalisant un produit scalaire dans un espace de plus grande dimension. Dans le cas d'un SVM multi-noyaux, le noyau k peut être toute combinaison convexe de fonctions semi-définies positives vérifiant la condition de Mercer. [V.R&T.S11]

$$k(x_i, x) = \sum_{j=1}^K \beta_j k_j \text{ avec } \beta_j \geq 0, \sum_{j=1}^K \beta_j = 1 \quad (3.2)$$

Ici, nous avons un noyau par type de descripteur :

$$k = \beta_1 k_{LGBP}(H_i; H) + \beta_2 k_{AAM}(C_i; C) \quad (3.3)$$

Les poids a_i et β_j sont optimisés pour avoir un hyperplan maximisant la marge (distance minimum d'un exemple à l'hyperplan). Ce problème d'optimisation est conjointement convexe en a_i et β_j . β_1 représente le poids donné aux descripteurs LGBP et β_2 celui donné pour le vecteur d'apparence des AAM. Ainsi, après apprentissage, le système est capable, de lui même, de trouver la meilleure combinaison de ces deux types de descripteurs. [V.R&T.S11]

- *Fonctions noyaux* : les résultats expérimentaux montrent que dans la reconnaissance basée sur des histogrammes utilisés avec un noyau d'intersection d'histogramme conduit aux meilleurs performances : $K_{LGBP}(H_i; H_j) = \sum_k \min(H_i(k); H_j(k))$ (3.4)

Pour le vecteur d'apparence des AAM, un noyau gaussien (RBF pour Radial Basis Function)

est utilisée : $KAAM(C_i, C_j) = e^{-\frac{\|x_i - x_j\|^2}{2\theta^2}}$ (3.5)

Avec θ un paramètre à optimiser sur une base de cross-validation durant l'apprentissage.

- *Filtrage Temporel* : Afin de prendre en compte l'information temporelle dans le système, on applique un filtre moyennneur sur chacune des sorties des SVM. La taille du filtre a été choisie de manière à maximiser la F1-mesure obtenue sur la base d'apprentissage. [V.R&T.S11]

3.5.2 Evaluation des descripteurs et des fonctions noyaux

Dans cette section, nous résumons des expérimentations précédemment effectuées [T.S,K.B et al 10] sur la base de données Cohn- Kanade. Ce sont ces résultats qui ont orienté vers le choix des histogrammes LGBP associé à un noyau d'intersection d'histogrammes.

- *Descripteurs*

Dans la figure 3.10.a, nous présentons la comparaison des performances de différents types d'histogramme en utilisant le noyau d'intersection d'histogramme. Les histogrammes

calculés à partir des images LBP et des images de Gabor donnent de bien meilleurs résultats que les histogrammes calculés directement sur les niveaux de gris. C'est avec la combinaison des deux approches, à travers les LGBP, qu'on obtient les meilleurs résultats. [V.R&T.S11]

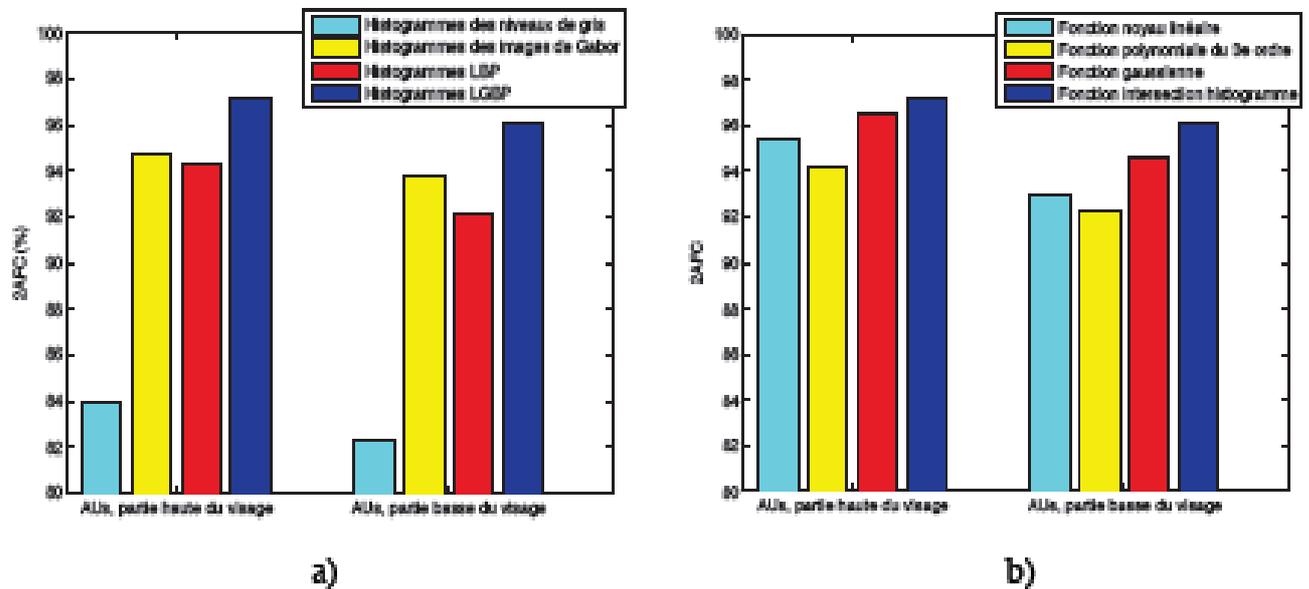


Figure 3.10 Scores 2AFC pour la détection des AU de la partie haute et basse du visage pour a) une fonction noyau intersection d'histogrammes et différents histogrammes et b) des histogrammes LGBP et différentes fonctions noyaux. [V.R&T.S11]

La figure 3.10.b montre les résultats obtenus avec les histogrammes LGBP associés à différentes fonctions noyaux. Pour le noyau polynomial et le noyau gaussien, différentes valeurs d'ordre du polynôme ou d'écart-type ont été testés. Les meilleures performances sont obtenues avec la fonction d'intersection d'histogramme. Comparée à un noyau linéaire, l'aire moyenne sous la courbe ROC passe de 95.5% à 97.2% pour les AU du haut du visage et de 93.0% à 96.1% pour les AU du bas. Le noyau gaussien donne quant à lui de bons résultats, mais les paramètres ont été directement optimisés sur la base d'apprentissage. [V.R&T.S11]

Conclusion

Nous pouvons dire que le processus de reconnaissance du visage et l'identité de la personne n'est pas un processus facile, car il doit passer par plusieurs étapes et met en scène chacun des avantages et des données spécifiques de son fonctionnement. Le résultat est soit le rejet ou l'acceptation d'une preuve d'identité de la personne. Nous avons mis l'accent sur une réduction fort prometteuse qui est la décomposition de Tucker3 qui est l'objectif de notre étude.

Capítulo

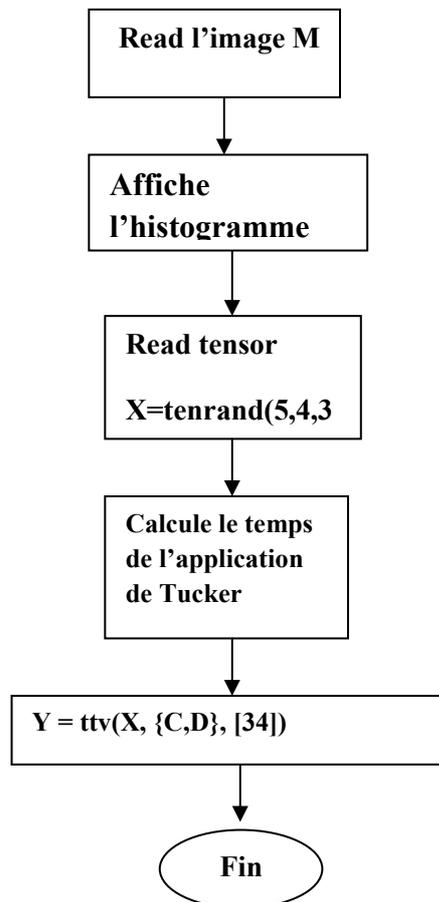
CONVENCIÓN DE LA REDUCCIÓN DE LOS EFECTOS AMBIENTALES

Introduction

Dans ce chapitre nous essayons d'implémenter plusieurs algorithmes de réductions linéaires et multilinéaires et calculer les performances de chacune. Ceci valide notre choix sur les méthodes de réduction multilinéaires. Nous commençons tout d'abord par la création d'un tenseur 3D sur lequel sera fondé notre algorithme. Puis nous procédons à l'application de notre méthode de réduction sur des images de visage en essayant de calculer le temps d'exécution de Tucker afin de valider son efficacité.

Après la création d'un tenseur de niveaux de gris d'ordre 3, son application sur des images de visage. Nous réalisons l'implémentation de la méthode de Tucker caractérisée par sa rapidité dans le système de la reconnaissance de visage et l'identité des personnes.

Et résumé les étapes de programme pour ce chapitre dans **l'organigramme** suivant :



X temps, C dans mode-3 et D dans mode-4

4.1 Environnement de la conception

4.1.1 Identification de l'ordinateur

- ✚ Processeur : Intel(R) Pentium(R) CPU P6200 @ 2.13Ghz 2.13Ghz
- ✚ Mémoire installée(RAM) : 2 Go
- ✚ Type de système : système exploitation 32 bits
- ✚ Logiciel : Matlab 7.5.0 (R2007b)
- ✚ Toolbox : v2.5

4.1.2 Matlab

Matlab est un environnement de calcul intégré basé sur un langage de programmation qui permet le développement rapide d'applications. Le langage permet la manipulation d'objets mathématiques à l'aide de fonctions. Matlab a été conçu à l'origine pour le calcul matriciel (*MATrix LABoratory*). Matlab dispose de fonctions de calcul de base ainsi que de bibliothèques plus spécialisées appelées boîtes à outils (*toolboxes*).

4.1.3 Quelles sont les particularités de MATLAB ?

MATLAB permet le travail interactif soit en mode commande, soit en mode programmation; tout en ayant toujours la possibilité de faire des visualisations graphiques. Considéré comme un des meilleurs langages de programmations (C ou Fortran), MATLAB possède les particularités suivantes par rapport à ces langages :

- la programmation facile,
- la continuité parmi les valeurs entières, réelles et complexes,
- la gamme étendue des nombres et leurs précisions,
- la bibliothèque mathématique très compréhensive,
- l'outil graphique qui inclut les fonctions d'interface graphique et les utilitaires,
- la possibilité de liaison avec les autres langages classiques de programmations (C ou Fortran).

Dans MATLAB, aucune déclaration n'est à effectuer sur les nombres. En effet, il n'existe pas de distinction entre les nombres entiers, les nombres réels, les nombres complexes et la simple ou double précision. Cette caractéristique rend le mode de programmation très facile et très rapide. En Fortran par exemple, une subroutine est presque nécessaire pour chaque variable simple ou double précision, entière, réelle ou complexe. Dans MATLAB, aucune nécessité n'est demandée pour la séparation de ces variables.

La bibliothèque des fonctions mathématiques dans MATLAB donne des analyses mathématiques très simples. En effet, l'utilisateur peut exécuter dans le mode commande n'importe quelle fonction mathématique se trouvant dans la bibliothèque sans avoir à recourir à la programmation. Pour l'interface graphique, des représentations scientifiques et même artistiques des objets peuvent être créées sur l'écran en utilisant les expressions mathématiques. Les graphiques sur MATLAB sont simples et attirent l'attention des utilisateurs, vu les possibilités importantes offertes par ce logiciel.

4.2 Conception des méthodes multilinéaires

4.2.1 Comment calculer un tenseur ?

Dans cette partie nous présentons les résultats obtenus par le programme d'application en Matlab avec la fonction 'tenrand' en précisant la taille (size) du tenseur. L'affichage des résultats est le suivant :

***A c'est un tenseur d'ordre 4 de size 5 x 3 x 3 x 3**

A(:, :, 1, 1) =

0.8930	0.0179	0.7975
0.3236	0.0197	0.3944
0.4595	0.1375	0.7227
0.3879	0.4177	0.4365
0.8872	0.5512	0.8261

A(:, :, 2, 1) =

0.9109	0.0417	0.0106
0.2516	0.7303	0.4532
0.0026	0.0279	0.9216
0.9214	0.2763	0.8273
0.2554	0.4644	0.5977

A(:, :, 3, 1) =

0.7102	0.0502	0.5007
0.9850	0.7635	0.6010
0.0084	0.2290	0.0159
0.5621	0.5776	0.2932
0.5132	0.1930	0.0056

A(:, :, 1, 2) =

0.7926	0.9896	0.7246
--------	--------	--------

0.4905 0.9564 0.4161
0.0095 0.5326 0.2894
0.9416 0.5279 0.8380
0.2380 0.2258 0.7197

$A(:, :, 2, 2) =$

0.8069 0.7928 0.0954
0.4450 0.0630 0.6331
0.8155 0.0558 0.5299
0.0078 0.4769 0.9942
0.2550 0.3150 0.6942

$A(:, :, 3, 2) =$

0.4351 0.8361 0.4166
0.2193 0.1351 0.9078
0.0342 0.3102 0.8442
0.2980 0.9234 0.5239
0.4077 0.2534 0.4436

$A(:, :, 1, 3) =$

0.4589 0.1287 0.5346
0.1010 0.0247 0.7153
0.1119 0.3197 0.2266
0.0131 0.8697 0.6163
0.4037 0.5055 0.0360

$A(:, :, 2, 3) =$

0.7808 0.9877 0.7390
0.7399 0.4741 0.6731
0.4590 0.8111 0.0562
0.6175 0.2028 0.9013
0.8369 0.9732 0.3213

$A(:, :, 3, 3) =$

0.8283 0.9899 0.6137
0.2718 0.5773 0.3921
0.5877 0.0339 0.4393
0.4590 0.5272 0.5843
0.1887 0.9263 0.4614

* **A** est un tenseur d'ordre 3 de size 5 x 3 x 3

$A(:,:,1) =$

0.2310	0.6172	0.7959
0.0128	0.9511	0.4224
0.9048	0.8920	0.2337
0.2487	0.1991	0.3345
0.2731	0.4611	0.0010

$A(:,:,2) =$

0.4213	0.0790	0.2278
0.6043	0.6510	0.9481
0.1673	0.6231	0.1811
0.9320	0.6997	0.9313
0.5230	0.4574	0.3545

$A(:,:,3) =$

0.4785	0.3391	0.0721
0.3585	0.0087	0.3752
0.3745	0.9778	0.1586
0.0128	0.5455	0.7784
0.2926	0.6924	0.1970

4.2.2 Conception de méthodes de réduction

- Algorithmes d'apprentissage

L'apprentissage se fait normalement comme pour tous les systems de reconnaissance en respectant le protocole de la base de données considérée. Nous obtenons ainsi la base de données modèle (indexée).

- Certaines fonctions

Les fonctions de réductions mises en oeuvre sont les suivantes :

PCA : Analyse en Composantes Principales

KPCA : Kernel-Analyse en Composantes Principales

LGE : Linear Graphic Embedding. Elle fournit un cadre général pour le graphique à base de sous-espace d'apprentissage. Cette fonction sera appelée LPP, NPE, IsoProjection, LSDA, AMF ...)

OLGE : Orthogonal Linear Graphic Embedding (Fournit un cadre général pour le graphique sous-espace sur la base d'apprentissage (vecteurs de base orthogonale). Cette fonction sera appelée OLPP. Il est également très facile de développer ONPE, OIsoProjection, OLSDA, OMFA ...

Tenseur LGE : Tensor Linear Graphique Embedding , fournit un cadre général pour le graphique basée un tenseur sous-espace d'apprentissage Cette fonction sera appelée par Tensor P. Il est également très facile de développer TensorNPE, TensorIso Projection, Tensor LSDA, Tensor MFA ...)

KGE : Kernel Graph Embedding , fournit un cadre général pour le graphique basé sur le noyau sous-espace d'apprentissage. Cette fonction sera appelée par KernelLPP Il est également très facile de développer KernelNPE, KernelIsoProjection, KernelLSDA, KernelMFA ...

LDA : Analyse Discriminante Linéaire. Généralement, LDA peut également utilisé LGE comme une sous-routine. Cependant, nous pouvons utilisé la structure de LDA graphique spéciale pour obtenir certains avantages de calcul.

KDA : Kernel Analyse Discriminante. Généralement, KDA peut également utilisé KEG comme une sous-routine. Cependant, nous pouvons utilisé la structure de KDA graphique spéciale pour obtenir certains avantages de calcul.

LPP : Locality Preserving Projection. Pour cela nous devons utilisé les fonctions LGE.m ainsi que constructW.m.

OLPP : Orthogonal Locality Preserving projections. Pour cela nous devons utilisé les fonctions LGE.m ainsi que constructW.m.

TensorLPP : Tensor Locality Preserving Projections.
Pour cela nous devons utilisé les fonctions LGE.m ainsi que constructW.m.

KernelLPP : Kernel Locality Preserving Projections.

Pour cela nous devons utiliser les fonctions LGE.m ainsi que constructW: Fonction utilisée pour construire la matrice d'affinité.

NPE : No Preserving Embedding. Pour cela nous devons utiliser les fonctions LGE.m.

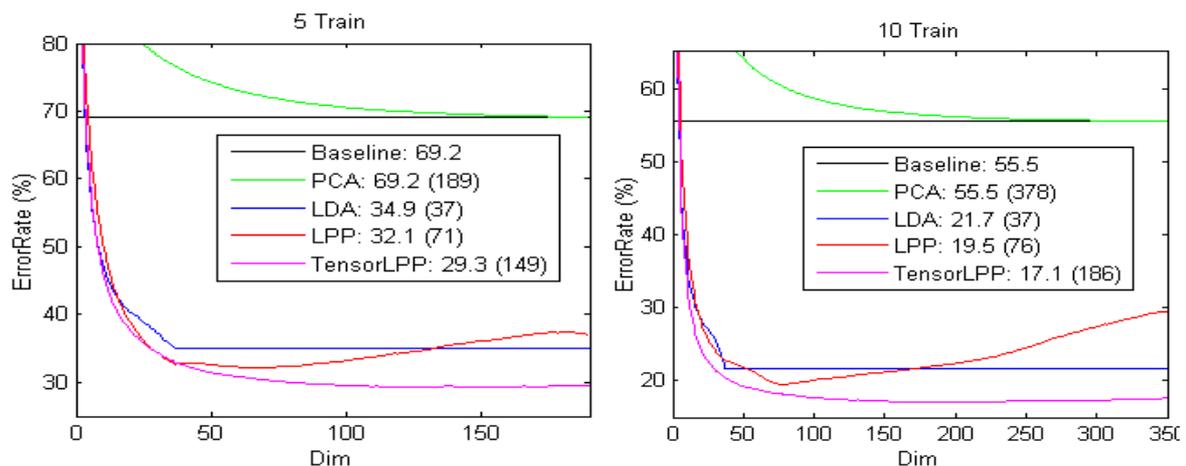
IsoProjection : Projection isométrique. Pour cela nous devons utiliser les fonctions LGE.m et le fichier dijkstra.dll (pour Windows) , dijkstra.mexglx (pour Linux): algorithme de Dijkstra (le code source).

LSDA : Local Sensible Analyse Discriminante. Pour cela nous devons utiliser LGE.m.

AMF : Analyse Marginale de Fisher. Pour cela nous devons utiliser LGE.m. Les ensembles de données sont adaptés au format Matlab. Les Bases de données pour le visage utilisées sont : Yale, ORL, PIE et YaleB.

Pour la comparaison la distance Euclidienne est utilisée pour la mesure de similitude pour l'identification de visage.

- 1) Calculer la matrice de distance Euclidienne des deux matrices de données.
- 2) Construire Kernel : Construire la matrice du noyau.



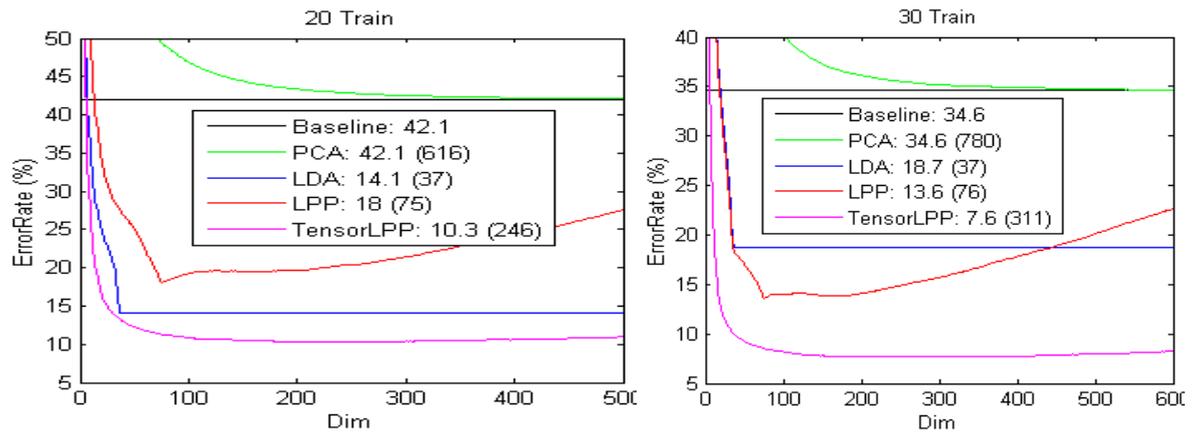


Figure 4.1 Taux d'erreur pour différents échantillons d'apprentissage

La figure 4.1 présente les courbes d'erreur pour les différentes méthodes de réduction linéaires et non linéaire utilisant le tenseur pour différentes tailles d'échantillon. Nous remarquons que l'erreur est meilleure quand la taille de la base d'apprentissage devient important ce qui est évident. Le résultat important dans notre travail est le taux d'erreur excellent obtenu dans le cas de l'association du tenseur aux méthodes de réduction linéaire. Ceci est montré sur la figure 4.1 et c'est valable quelque soit la taille de l'échantillon étudié. Nous remarquons la meilleur méthode de réduction Tenseur LPP (*Locality Preserving Projection*).

Ce résultat est encourageant et nous pousse à utiliser l'analyse tensorielle en lui associant une réduction adaptée qui n'est autre que la réduction de Tucker 3.

4.2.3 Comment créer et afficher un tenseur ?

Pour créer un tenseur nous utilisons une fonction que nous avons conçu `tensorgray()` ensuite pour valider le bon fonctionnement de notre algorithme nous affichons ce tenseur d'ordre 3.

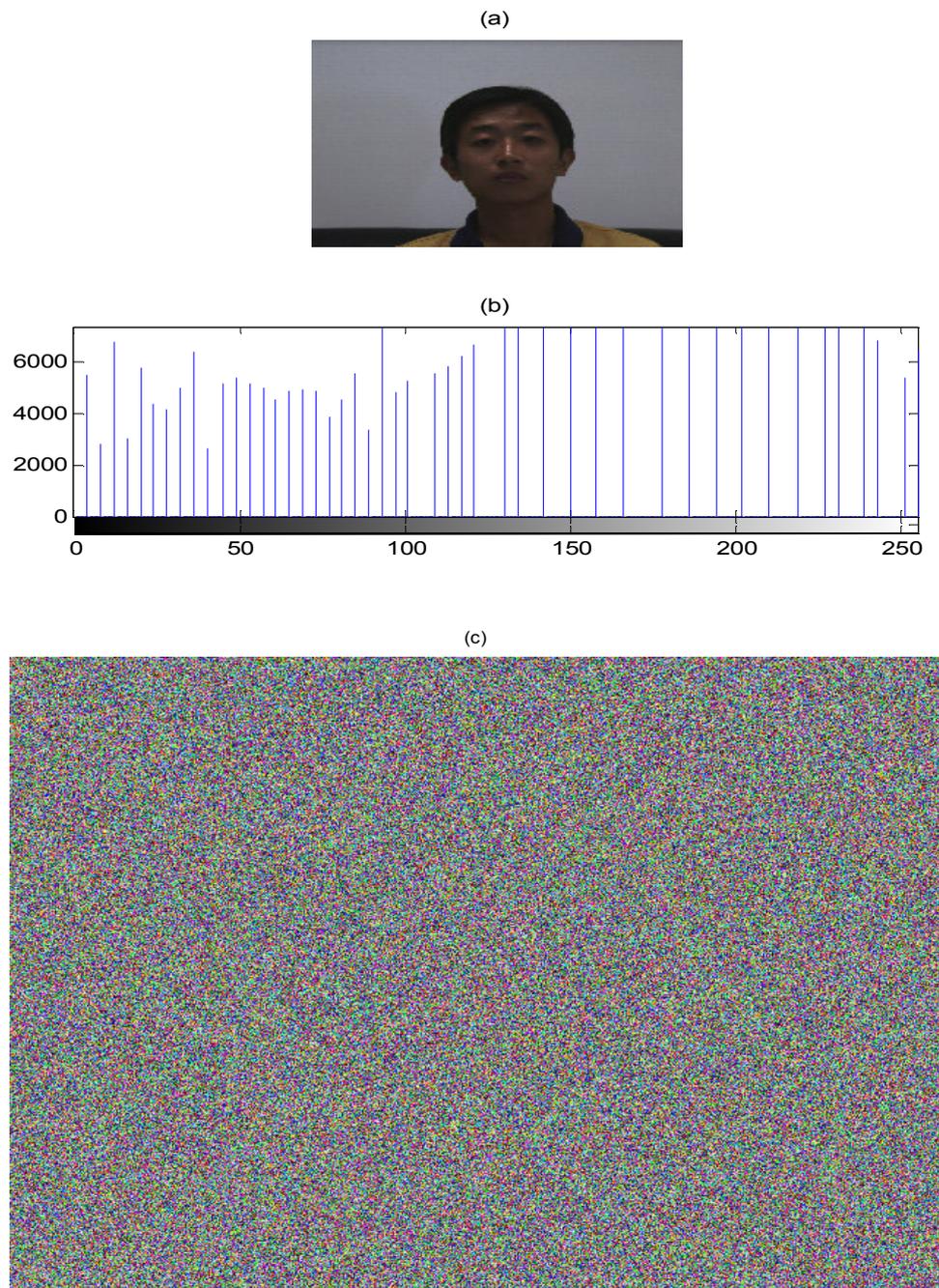


Figure 4.2 (a) image originale, (b) histogramme d'image originale, (c) tenseur (affichage 2D)

Une fois que notre tenseur est conçu on peut envisager notre application et tenter l'implémentation de la réduction de Tucker sur des images de visages 3D.

4.2.4 Comment calculer le temps d'exécution de la méthode de réduction de Tucker ?

Quelques résultats d'application :

Y c'est un tenseur Tucker de taille $3 \times 4 \times 2$

Y.core c'est un tenseur de size $2 \times 2 \times 2$

$Y.U\{2\}$: Y = pour obtenue utilise l'instruction `ttv(X, {C,D}, [3 4])` ; X temps C dans mode-3 & D dans mode-4.

ans :pour obtenue utilise l'instruction `norm(full(Y) - ttv(full(X),A, 1))` ; Same as dense case.

Y : pour obtenue utilise l'instruction `ttv(X, {A,B,C,D}, [1 2 3 4])` ; All-mode multiply -- scalar result.

Dans cette partie est applique les instructions précédant obtenue sur le résultat suivant :

$Y.core(:, :, 1) =$

1.3171 0.2658

1.0694 0.9612

$Y.core(:, :, 2) =$

1.3377 1.4308

0.3816 0.7186

$Y.U\{1\} =$

0.8729 0.9669

0.2379 0.6649

0.6458 0.8704

$Y.U\{2\} =$

0.0099 0.8903

0.1370 0.7349

0.8188 0.6873

0.4302 0.3461

$Y.U\{3\} =$

0.1660 0.1911

0.1556 0.4225

ans =

3.9154e-016

Y =

3.8758

Y c'est un Tucker tenseur de 5 x 3

Y.core c'est un tenseur de size 2 x 2

$Y.core(:, :) =$

0.6489 0.3358

0.5348 0.3779

$Y.U\{1\} =$

0.3651	0.4586
0.3932	0.8699
0.5915	0.9342
0.1197	0.2644
0.0381	0.1603

$Y.U\{2\} =$

0.8729	0.9669
0.2379	0.6649
0.6458	0.8704

Y c'est un Tucker tenseur de size 4

Y.core c'est un tenseur de size 2

$Y.core(:) =$

2.3205
2.3598

$Y.U\{1\} =$

0.0099	0.8903
0.1370	0.7349
0.8188	0.6873
0.4302	0.3461

Conclusion

Dans ce chapitre nous obtenu sur les résultats elles existent en Matlab donc, nous l'avons vu le tenseur applique sur le visage aide à les méthodes de réduction et calcul le temps de la méthode Tucker est caractérisé à rapidité pour reconnaissance de visage

Conclusion générale

Ce mémoire est consacré au développement de nouvelles méthodes algébriques dédiées à l'analyse des données multidimensionnelles telles les images de visage issues de base de données de tailles gigantesque et aussi à l'analyse des images hyper-spectrales. Notre objectif est d'analyser conjointement les dimensions spatiale et spectrale de telles données. De ce fait, nous tentons de rompre avec la dichotomie entre le domaine spatial et spectral qui continue, très majoritairement, à dominer dans les traitements d'imagerie hyper-spectrale. Dans ce contexte, nous considérons ces données dans leur ensemble, en les représentant par un tenseur d'ordre 3. Cette représentation implique l'utilisation d'outils d'algèbre multilinéaire, les méthodes qui en découlent sont alors dites *multidimensionnelles* ou *multimodales*.

Très récemment, dans la recherche on a constaté l'intérêt naissant de l'utilisation de méthodes basées sur la décomposition tensorielle pour la compression ou l'extraction d'information des données hyper-spectrales. Ce procédé ouvre des perspectives et une nouvelle conception dans l'analyse de ces données complexes. Dans ce travail, nous nous focalisons sur l'une (la DR) des deux problématiques liées à la nature même de l'imagerie multi dimensionnelle. Il s'agit de :

1. *Débruitage des données.*

2. *Réduction de la dimension spectrale (DR spectrale).* En effet, le grand nombre de bandes spectrales implique de manipuler des espaces vectoriels de grande dimension. En plus de représenter un défi algorithmique et de stockage non négligeable, la grande dimension des spectres implique une diminution de la précision de l'estimation statistique pour un nombre d'échantillons fixe. Par conséquent, cela conduit à une dégradation des résultats de classification. La méthode proposée est basée sur la décomposition tensorielle de TUCKER qui généralise la représentation matricielle. Cette décomposition permet en outre l'approximation d'un tenseur en rangs- (K_1, K_2, K_3) inférieurs. Projeter les données dans un sous-espace de dimension réduite, les méthodes multimodales proposées sont dérivées de cette approximation tensoriel. L'objectif est de projeter les données dans le sous-espace où le signal est prédominant par rapport au bruit, pour augmenter le rapport signal à bruit et aussi de projeter les données dans un sous-espace de dimension réduite, qui concentre l'information utile pour augmenter la séparabilité des classes. La nature de l'information à favoriser est dans ce cas à définir.

Conclusion générale

Nous avons mené une étude détaillée sur la décomposition de Tucker en donnant la définition mathématique d'un tenseur d'ordre supérieur. Les propriétés propres à leur utilisation ainsi que les opérateurs tensoriels manipulés dans ce travail, sont exposés. En particulier, certaines méthodes de décomposition tensorielle. L'accent est mis sur la décomposition de TUCKER 3 qui est à la base des méthodes multimodales développées dans ce mémoire. Cette décomposition généralise l'analyse en composantes principales pour l'approximation d'un tenseur en rangs- (K_1, \dots, K_N) inférieurs.

La méthode de base DR algèbre multilinéaire - a deux objectifs: (i) pour estimer la matrice $U^{(3)}$ de l'équation en utilisant des informations spatiales, (ii) effectuer une transformation spatial-spectral commun dans le but de blanchir et de compression des modes spatiaux et spectraux. En conséquence, le procédé proposé effectue simultanément une réduction des dimensions du mode spectral ($p < I_3$), et une projection sur un inférieur (K_1, K_2) de dimension sous-espace des deux modes spatiaux.

Le but de ce projet est l'étude de méthode de réduction multilinéaire en vue d'être implémenté sur un système de reconnaissance automatique de visage 3D et hyperspectrales IHS. Ce modèle contribuera sûrement à l'amélioration des performances du système. Nous avons présenté une méthode de décomposition de Tucker. Nous avons vérifié nos méthodes numériques présentées et expérimenté avec un ensemble de données image visage.

Pour les systèmes linéaires et problèmes aux valeurs propres avec une structure très particulière, il est parfois possible de concevoir des algorithmes spécialisés qui peuvent être plus efficaces et plus faciles à analyser. Seulement les contraintes sont énormes et difficiles à résoudre. Les algorithmes randomisés représentent une solution alternative pour extraire rapidement une approximation de rang bas d'informations partielles sur les entrées de la matrice. Ces idées ont été étendues à bas rang décompositions tensorielles, pour la décomposition Tucker. Pour cela, le long de notre étude nous avons essayé d'implémenter ces outils mathématiques sur des exemples simples avant d'étendre nos manipulations sur des applications sur l'image de visage.

D'après les résultats de toutes ces expériences nous pouvons affirmer que les outils d'algèbre multilinéaires et les réductions telle : TUCKER3 ont un avenir prometteur dans les applications biométriques.

Nous proposons dans le futur d'appliquer cette réduction sur des BDD de visages 3D sous diverses variantes et ajouter une phase de débruitage dans le cas d'utilisation d'images IHS.

Abréviations et acronymes

SVR3D : *Système de reconnaissance de visage 3D.*

ICP : *Iterative Closest Point.*

MMG : *Mélange de Modèles Gaussiens.*

PSD : *Point Set Distance.*

ELBP : *Extended Local Binary Patterns.*

LBP : *Gabor et local binary pattern.*

PCA : *Principal Component Analysis.*

LDA : *Linear Discriminant Analysis.*

GDA : *Generalized Discriminant Analysis.*

DLDA : *Direct Linear Discriminant Analysis.*

KFA : *Kernel Fisher Analysis.*

ICA : *L'analyse en composantes indépendantes.*

HMM : *Hidden Markov Models.*

SVM : *Machine à vecteurs de support.*

QP : *Programme Quadratique.*

HTD : *La décomposition hiérarchique de Tucker.*

IHS : *Images hyper-spectrales.*

DR : *Dimensionality reduction.*

3MFA : *Three-mode factor analysis.*

3MPCA : *Three-mode principal components analysis.*

HOSVD : *Higher Order Singular Value Decomposition.*

LRTA : *Lower rank- (K_1, \dots, K_N) tensor approximation.*

Abréviations et acronymes

ALS : *Alternating Least Square.*

HOOI : *Higher-Order Orthogonal Iteration.*

PARAFAC : *PARAllel FACtor analysis.*

INDISCAL : *Individual Differences in Scaling.*

INDORT : *ORTHogonally constrained INDSCAL.*

NTF : *Non négatif tenseur factorisation.*

NMF : *Maximum Noise Fraction.*

NTD : *Tucker Décomposition non négatif.*

PMF : *Positive matrix factorization.*

KL : *Karhunen-Loève.*

NAPCA : *Noise-Adjusted Principal Component Analysis.*

PP : *Poursuite de projection.*

DBFE : *Decision Boundary Feature Extraction.*

MD : *Distance de Mahalanobis.*

IPD : *Distance inter-pupillaire.*

CCA : *Curvilinear component analysis.*

CDA : *Analyse en distances curvilinéaires.*

LLE : *Locally Linear Embedding.*

SAM : *Spectral Angle Mapper.*

A-1 Les bases de visages utilisées

A.1.1 IV² [A.M09]

Le projet IV² est un projet d'évaluation biométrique français. Il a pour but de créer des ressources et des conditions d'une évaluation à l'échelle nationale et internationale de différents systèmes liés à l'information du visage, de l'iris et de la voix, dans des milieux semi-contraints.

Pour constituer la base de données IV², une cabine d'acquisition très semblable à une cabine de type photomaton a été utilisée et est représentée dans la figure A.1

Afin d'obtenir un éclairage optimal dans la cabine, nous disposons de plusieurs lampes halogènes qui seront pilotées depuis un ordinateur à l'aide de télé- variateurs de puissance.

La base IV² est une base multimodale comportant une séquence de 2 minutes de visage parlant qui est acquise avec des caméras différentes dont une "Webcam", des données du visage 3D avec différentes expressions et sous diverses conditions d'illumination, des données de visage acquises par stéréovision et des images d'iris acquises en proche infrarouge.

Cette base de données a été acquise sur trois sites différents : l'INT (EVRY), l'université d'Evry et l'école Centrale de Lyon (LYON).



Vue externe de la cabine IV²



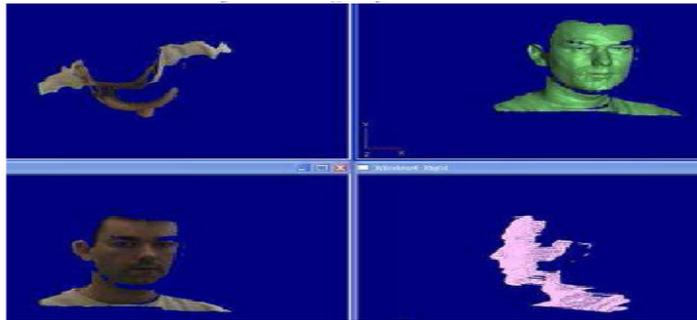
Vue interne de la cabine IV²



Exemple d'image d'iris



Exemple d'images par stéréovision



Exemple d'acquisition 3D

Figure A.1 Exemples de données de la base IV² [A.M09]

A.1.2 La base XM2VTS [Mes99]

La base XM2VTS (*The Extended M2VTS Database*) est une base de données multimodale qui renferme des images faciales 2D et 3D, des séquences vidéo avec des enregistrements de la voix. La base de données a été acquise dans le cadre du projet M2VTS (*Multi Modal Verification for Teleservices and Security applications*) à l'Université de Surrey (UK). La base de données contient 295 personnes enregistrées en 4 sessions durant une période de 4 mois. La base de données de visage 2D contient des prises de vue faciales et de profil. Deux prises de vue faciales ont été enregistrées, à chaque session, ce qui fait un total de 8 images pour chaque personne. La figure A.2 illustre un exemple d'images d'une personne. Les images en colonnes représentent les sessions auxquelles elles appartiennent.



Figure A.2 Exemples d'images de la base XM2VTS. [Mes99]

La base de données XM2VTS a été conçue initialement pour l'authentification de personnes. Ainsi un protocole d'évaluation a été élaboré spécifiant les ensembles

d'apprentissage, d'évaluation et de test [LM98]. Nous revenons sur ce protocole dans le chapitre suivant pour évaluer notre système d'identification.

A.1.3 La base FERET

La base de données FERET a été collectée dans le cadre du programme « Facial Recognition Technology » [PMRR00, Sit4] mené par le National *Institute of Standards and Technology* (NIST) Américain. Il s'agit de la plus grande base disponible pour les chercheurs qui a été acquise avec des poses différentes et durant 15 sessions entre 1993 et 1996. Les images, initialement collectées depuis un appareil photographique de 35mm ont ensuite été digitalisées. Une première version de cette base de données a été réalisée en 2001 et contient 14051 images faciales en niveaux de gris avec une résolution de 256 x 384 pixels. La version la plus récente, réalisée en 2003, renferme des images numériques couleurs de plus grande qualité avec une résolution de 512 x 768 pixels et une compression sans perte de données à la différence des premières images en niveaux de gris. En plus, de multiples erreurs de noms d'images, d'identifiants, et dates de capture, qui figurent sur la première base à niveau de gris ont été corrigées. Cette dernière base contient 11338 images représentant 994 personnes différentes. Pour chaque individu, on dispose d'une vue faciale régulière *fa* et une vue faciale alternative *fb* prise un peu après *fa*. D'autres poses ont été acquises pour la majorité de ces individus allant du profil gauche au profil droit avec des rotations de 15°, 22°, 45°, 67°, et 90° en profondeur de la tête. Pour quelques personnes de la base, on dispose d'autres vues « *duplicate* » collectées dans des conditions similaires à *fa* et *fb* mais dans des sessions ultérieures. Aucune contrainte n'est imposée sur la date de la prise de vue de l'image duplicate I. par contre, la vue duplicate II a été collectée au moins 540 jours après la première prise de vue.

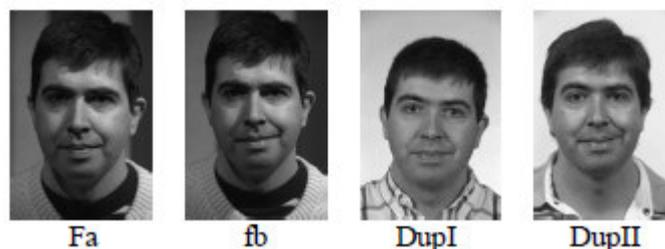


Figure A.3 Extrait de la base Color FERET Les images sont transformées en niveau de gris.

A.1.4 La base AR [MB98]

La base AR a été constituée en 1998 au sein du laboratoire Computer Vision Center (CVC) à Barcelone, en Espagne. 116 personnes (63 hommes et 53 femmes) sont enregistrées. Les images sont en couleur de taille 768 x 576 pixels. 13 vues de chaque sujet ont été collectées.

Pour la majorité de ces personnes, 13 autres vues ont été acquises lors d'une deuxième session à deux semaines d'intervalle. Un extrait des images collectées lors de la première session est donné en figure A.4. Ces vues renferment des changements d'expression faciale, d'éclairage, ainsi que des occultations partielles des yeux (lunettes de soleil) et de la partie basse du visage (cache-col). Lors de la seconde session, les 13 vues sont collectées dans les mêmes conditions que pour la première.

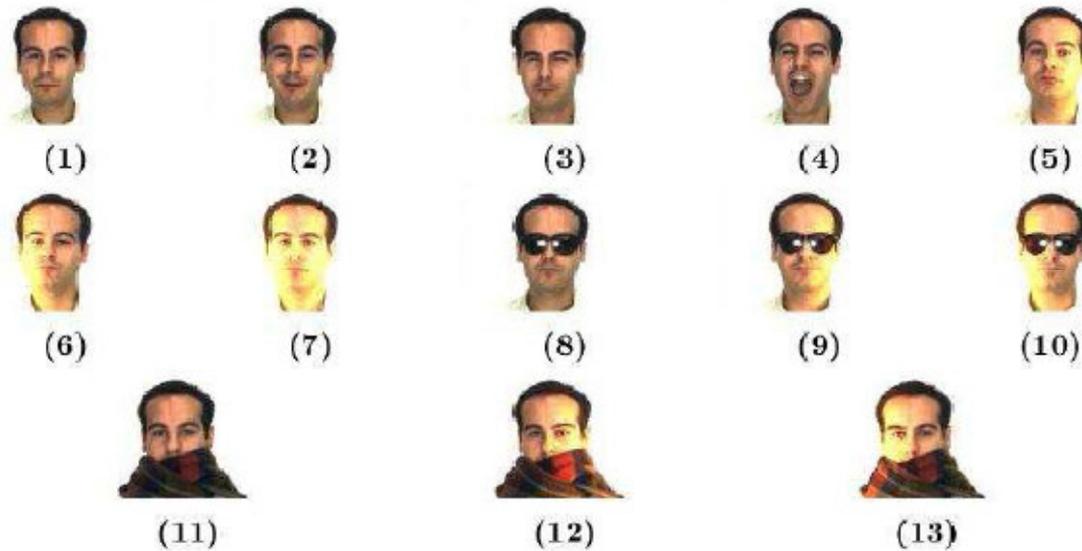


Figure A.4 Extrait de la base AR. Ensembles des vues collectées pour un individu de la base.

[MB98]

A.1.5 La base CVL [Sit5]

Cette base de données de visage a été créée au *Computer Vision Laboratory* (CVL) à l'université de Ljubljana [CVL.si]. 114 personnes y sont enregistrées. Les images sont en couleur de taille 640 x 480 pixels. 7 vues de chaque sujet ont été collectées. Ces vues renferment différentes poses et expressions prises dans des conditions de luminosité uniformes. Parmi ces images, trois seulement sont de face. La figure A.5 illustre un exemple d'images de face de la base CVL.



Figure A.5 Images faciales extraites de la base CVL. [Sit5]

A.1.6 La base ORL [A.C09]

La base ORL a été collectée entre avril 1992 et avril 1994 par un laboratoire d'AT&T, basé à Cambridge. La base contient 40 personnes, chacune étant enregistrée sous 10 vues différentes (cf. figure A.6). Les images sont de taille 112×92 pixels. Pour quelques sujets, les images ont été collectées à des dates différentes, avec des variations dans les conditions d'éclairage, les expressions faciales (expression neutre, sourire et yeux fermés) et des occultations partielles par les lunettes. Toutes les images ont été collectées sur un fond foncé. Les poses de la tête présentent quelques variations en profondeur par rapport à la pose frontale. Cependant, ces variations ne portent que sur certaines personnes et ne sont donc pas systématiques.



Figure A.6 Extrait de la base ORL. Pour chacune des 40 personnes enregistrées, on dispose de 10 vues avec des changements de pose, d'expression et d'éclairage. [A.C09]

A.1.7. Notre corpus d'évaluation [A.C09]

Dans un premier temps, n'ayant pas encore la totalité de ces bases, nous avons travaillé exclusivement sur la base XM2VTS. Cette première phase d'évaluation nous a permis d'étendre rapidement nos algorithmes sur le corpus constitué par un mélange de la plupart des bases présentées. Le tableau A.1 récapitule les différentes bases utilisées et le nombre de sessions et d'images disponibles pour chaque personne.

Concernant la base IV², nous avons récupéré les images « cam » du camescope utilisé lors de la campagne d'évaluation. 9 images par personne sont extraites lors de la première session et seulement cinq pour la deuxième session.

Tableau A.1 Bases de données de visage formant le corpus de notre évaluation. [A.C09]

Base	IV ²	XM2VTS	Color FERET	AR	CVL	ORL
# de personnes	302	295	994	136	113	40
# de sessions	1-3	4	1-15	1-2	1	1
# d'images faciales par personne	9-14	8	2-22	13-26	3	10

Ainsi nous avons regroupé une base de travail assez conséquente de 1880 personnes pour évaluer nos approches d'identification. Pour la majorité des images la position des centres des yeux était fournie avec la base. Nous avons annoté manuellement le reste des images pour pouvoir les normaliser. Ainsi, toutes les images de notre corpus sont normalisées selon la procédure décrite à l'étude précédente. Notre procédure consistait à une rotation du visage dans l'image de manière à ce que l'axe interoculaire soit horizontal, suivi d'un redimensionnement des visages pour avoir une même distance interoculaire. Ces deux premières étapes qui nécessitent les coordonnées des centres des yeux dans l'image brute sont suivies d'un découpage et d'une égalisation de l'histogramme du visage découpé.

A.1.8 AT&T

Cette base contient des images de visage de 40 personnes, avec 10 images pour chacune. Pour la plupart des sujets, les 10 images ont été tirées à différentes heures et avec différentes conditions d'éclairage, mais toujours sur un fond foncé. La taille d'image est 92 x 112. Les limitations de cette base de données sont : le nombre limité de personnes, la non-conformité des conditions d'éclairage d'une image à une image, et la non assignation de toutes les images par les différentes expressions faciales, rotation de la tête, ou conditions d'éclairage [Sit6].

A.1.9 FEI

La base de données de visage de FEI est une base de données brésilienne de visage qui contient un ensemble d'images de visage prises entre juin 2005 et mars 2006 au laboratoire d'intelligence artificielle de FEI dans São Bernardo font Campo, São Paulo, Brésil. Elle contient des images de visage de 200 personnes, avec 14 images pour chacune d'entre elles, un total de 2800 images. Toutes les images sont en couleur et prises sur un fond blanc homogène en position frontale avec divers angles de pose de chaque personne, s'étendant du profil aux vues frontales. La variation d'éclairage est de 10% et la résolution de l'image est de 640x480 Pixel [Sit6].

Le nombre de sujets masculins et féminin sont exactement identique et égale à 100. La figure A.7 montre quelques exemples des variations d'image de la base de données de visage de FEI.



Figure A.7 Exemples d'images de visages de la base FEI [Sit6].

A.1.10 MUCT

La base de données de MUCT se compose de 3755 en couleur [Sit6]. Elles ont été rassemblées en photographiant 345 sujets à de divers angles. La base de données a été créée pour fournir plus de diversité d'éclairage, d'âge, et d'appartenance ethnique, aussi pour développer, tester et évaluer des algorithmes de reconnaissance de visages. Les images couleurs ont une résolution 512 x 768, en format JPG. Quelques exemples sont montrés dans la figure A.8.



Figure A.8 Exemples d'images de visage de la base MUCT [Sit6].

Chapitre 1

[BM13] THÈSE Présentée par BELAHCENE Mébarka ; Authentification et Identification d'individus en biométrie, Soutenue le 14janvier 2014 à l'université Mohamed KHIDER Biskra.

[A.C09] THESE Présenté par Anis CHAARI Nouvelle approche d'identification dans les bases de données biométriques basée sur une classification non supervisée Soutenue le mardi 6 Octobre 2009 au laboratoire IBISC (mardi 29 avril 2014, 19:54:45).

[L.B12] Thèse Présentée par Lahoucine Ballihi Biométrie faciale 3D par apprentissage des caractéristiques géométriques : Application à la reconnaissance des visages et à la classification du genre Thèse soutenue le 12 Mai 2012 (mardi 29 avril 2014, 19:54:51).

[B.S12] Mémoire de Magister en Electronique Présenté par : BOUDJELLAL Sofiane Détection et identification de personne par méthode biométrique, Juin 2012(17 février 2014, 17:36:46).

[Sit1] editeur.over-blog.com jeudi 16 janvier 2014, 22:53:28.

[A.M09] THESE Présenté par Anouar Mellakh Reconnaissance des visages en conditions dégradées 07 Avril 2009)10 février 2014, 17:22:47).

[PHTN10] Stagiaire : PHAM Thanh Nam, RECONNAISSANCE DE VISAGES À PARTIR DES SÉQUENCES VIDÉO Evry, Avril – Septembre, 2010(mardi 29 avril 2014, 19:55:03)

[HM&PJP98] H. Moon and P.J. Phillips. Analysis of pca-based face recognition algorithms. Empirical Evaluation Techniques in Computer Vision, 1998.

[PHPJ,WH,HJ&RP98] Phillips P. J., Wechsler H., Huang J., and Rauss P. The feret database and evaluation procedure for face-recognition algorithms. Image and Vision Computing, 1998.

[BDWY&RB02] B. Draper W. Yambor and R. Beveridge. Analyzing pca-based face recognition algorithms : Eigenvector selection and distance measures. Empirical Evaluation Methods in Computer Vision, World Scienti_c Press, 2002.

[Sit2]: www.memoireonline.com/.../Reconnaissance-de-visages-par-Analyse-Dis

[L.F.C, H.Y.M. L&al00] L.F. Chen, H.Y.M. Liao, M.T. Ko, J.C. Lin, and G.J. Yu. A new lda-based face recognition system which can solve the small sample size problem. Pattern Recognition, 33(10) :1713_1726, October 2000.

[HY&JY01] Hua Yu and Jie Yang. A direct lda algorithm for high-dimensional data - with application to face recognition. Pattern Recognition, 34(10) :2067_ 2070, 2001.

[MB,HL&TS98] M. Bartlett and H. Lades and T. Sejnowski, "Independent component representations for face recognition", in T. Rogowitz, B. & Pappas, editor, Proceedings of the SPIE Symposium on Electronic Imaging: Science and Technology; Human Vision and Electronic Imaging III, volume 3299, San Jose, CA, January 1998. SPIE Press. 1998.

[Sit3] "Support vector machine". Document available at : http://en.wikipedia.org/wiki/Support_vector_machine.

[LG10] L. Grasedyck. Hierarchical low rank approximation of tensors and multivariate functions, 2010. Lecture notes of Zurich summer school on Sparse Tensor Discretizations of High- Dimensional Problems.

[DK&CT12] Daniel Kressner and Christine Tobler htucker – A Matlab toolbox for tensors in hierarchical Tucker format February 6, 2012.

[L.S&M.K87] L. Sirovich and M. Kirby. "Low-dimensional procedure for the characterization of human faces". Journal of Optical Society of America, Vol. 4, No. 3, pp. 519–524, 1987.

[MT&AP91] M. Turk and A. Pentland. "Eigenfaces for recognition". Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71–86, 1991.

[PB,JH&DK97] P. Belhumeur, J. Hespanha, and D. Kriegman. "Eigenfaces vs. Fisherfaces : Recognition Using Class Specific Linear Projection". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, pp. 711–720, 1997.

[MB,JM&TS02] M. Bartlett, J. Movellan, and T. Sejnowski. "Face recognition by independent component analysis". IEEE Transactions on Neural Networks, Vol. 13, No. 6, pp. 1450–1464, 2002.

[GG,SL&KC00] G. Guo, S. Li, and K. Chan. "Face Recognition by Support Vector Machines". In : Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, pp. 196–201, 2000.

[SA,PC&RL06] S. Arca, P. Campadelli, and R. Lanzarotti. "A Face Recognition System Based On Automatically Determined Facial Fiducial Points". Pattern Recognition, Vol. 39, No. 3, pp. 432–443, 2006.

Chapitre 2

[NEzbaed 08] Nadine Renard, Salah Bourenane and Jacques Blanc-Talon Denoising and Dimensionality Reduction Using Multilinear Tools for Hyperspectral Images IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, APRIL 2008.(mardi 29 avril 2014, 19:55:33).

[NR08] THÈSE Présentée et soutenue publiquement par : Nadine RENARD TRAITEMENT DU SIGNAL TENSORIEL APPLICATION À L'IMAGERIE HYPERSPECTRALE, ANNEE : 2008.)mardi 29 avril 2014, 19:55:33).

- [**M.L&L.H02**] M. LENNON, G. MERCIER et L. HUBERT-MOY, *Nonlinear filtering of hyperspectral images with anisotropic diffusion*, IEEE Trans. on Geosc. and Remote Sens. Symposium, 2002. IGARSS 02. 2002 IEEE Int., vol. 4, pp. 2477–2479, Juin 2002.
- [**V.S&L.L08**] V. DE SILVA et L. LIM, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. Appl., A paraitre. 2008.
- [**J.L.F98**] J. LELONG-FERRAND et J. ARNEAUDIES, *Cours de mathématiques. Tome 1 : Algèbre*, Dunod (3e ed.), 1998.
- [**P.L83**] P. LEROUX, *Algèbre linéaire : une approche matricielle*, Québec :Modulo, 1983.
- [**B.W.B&T.G.K06**] B. W. BADER et T. G. KOLDA, *Algorithm 862 : MATLAB tensor classes for fast algorithm prototyping*, ACM Trans. on Mathematical Software, vol. 32, no. 4, pp. 635–653, Dec. 2006.
- [**L.T63**] L. TUCKER, *Implications of factor analysis to three-dimensional matrices*, Problems in Measuring Change, C.W. Harris, ed., University of Wisconsin Press, pp. 122-137, 1963.
- [**L.T64**] L. TUCKER, *The extension of factor analysis to three-dimensional matrices*, (pp. 109– 127), Holt, Rinehart and Winston, NY, 1964.
- [**L.T66**] L. TUCKER, *Some mathematical notes on three-mode factor analysis*, Psychometrika, vol. 31, pp. 279–311, 1966.
- [**P.K&J.L80**] P. KROONENBERG et J. DE LEEUW, *Principal component analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika, vol. 45, no. 1, pp. 69–97, Mars 1980.
- [**A.K&H.N86**] A. KAPTEYN, H. NEUDECKER et T. WANSBEEK, *An approach to n-mode components analysis*, Psychometrika, vol. 51, pp. 269–275, 1986.
- [**L.L&BM00**] L. DE LATHAUWER, B. DE MOOR et J. VANDEWALLE, *A multilinear singular value decomposition*, SIAM Journal on Matrix Analysis and Applications, vol. 21, pp. 1253–1278, Avril 2000.
- [**L.L00**] L. DE LATHAUWER, B. DE MOOR et J. VANDEWALLE, *On the best rank- (r_1, \dots, r_N) approximation of higher-order tensors*, SIAM Journal on Matrix Analysis and Applications, vol. 21, pp. 1324–42, Avril 2000.
- [**BRA00**] B. R. ANDERSSON, C.A., *The n-way toolbox for MATLAB*, Chemometrics and Intelligent Laboratory Systems, vol. 52, pp. 1–4, 2000.
- [**Y.E&G.Y36**] Y. ECKART et G. YOUNG, *The approximation of a matrix by another of lower rank*, Psychometrika, vol. 1, pp. 211–218, 1936.
- [**L.L97**] L. DE LATHAUWER, *Signal Processing Based on Multilinear Algebra*, Thèse de doctorat, K.U. Leuven, E.E. Dept. (ESAT), Belgium, Sept. 1997.
- [**T.G.K07**] T. G. KOLDA et B. W. BADER, *Tensor decompositions and applications*, Rapport technique SAND2007-6702, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, Nov. 2007.
- [**BW.B06**] Brett W. Bader and Tamara G. Kolda ; Efficient MATLAB computations with sparse and factored tensors, 2006 (mardi 29 avril 2014, 19:55:31).
- [**SNL06**] *Multilinear operators for higher-order decompositions*, Tech. Report SAND2006-2081, Sandia National Laboratories, Albuquerque, New Mexico and Livermore, California, Apr. 2006.
- [**L. R. T 66**] L. R. TUCKER, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966) pp. 279-311.
- [**Y-DK07**] Yong-Deok Kim, Seungjin Choi, Nonnegative Tucker Decomposition 2007 (mardi 29 avril 2014, 19:55:36).
- [**J.S89**] J. SERRA, *Image analysis and mathematical morphology : theoretical advances*, vol. 2, Academic Press, London, 1989.
- [**J.W96**] J. WEICKERT, *Anisotropic diffusion in image processing*, Thèse de doctorat, University of Kaiserlautern, Germany, January 1996.
- [**D.D95**] D. DONOHO, *De-noising by soft-thresholding*, IEEE Trans. on Information Theory, vol. 41, no. 3, pp. 613–627, Mai 1995.
- [**A.H&E.O00**] A. HYVARINEN et E. OJA, *Independent component analysis : Algorithms and applications*, Neural Networks, vol. 13, pp. 411–430, 2000.
- [**J.F&J.T74**] J. FRIEDMAN et J. TUCKEY, *A projection pursuit algorithm for exploratory data analysis*, IEEE Trans. on Computers, vol. 23, no. 9, pp. 881–889, 1974.
- [**D.H&W.X03**] D. HONGTAO, Q. HAIRONG, W. XIAOLING, R. RAMANATH et W. SNYDER, *Band selection using independent component analysis for hyperspectral image processing*, dans *Proceedings Applied Imagery Pattern Recognition Workshop 32nd*, (pp. 93 – 98), 15-17 Oct 2003.
- [**NR&JB-T08**] Nadine Renard, Salah Bourennane and Jacques Blanc-Talon, Denoising and Dimensionality Reduction Using Multilinear Tools for Hyperspectral Images, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, APRIL 2008 (vendredi 8 février 2013, 17:06:32).

[D.M04] D. MUTI, *Traitement du signal tensoriel. Application aux images en couleurs et aux signaux sismiques*, Thèse de doctorat, Institut Fresnel, Université Paul Cézanne, Aix- Marseille III, Marseille, France, 2 december 2004.

Chapitre 3

[L.H11] THESE *Présentée Par* Lyes HAMOUDI, Application de techniques d'apprentissage pour la détection et la reconnaissance d'individus Soutenue le 7 juin 2011(samedi 15 février 2014, 14:29:52).

[YanKM02] M. Yang, D. Kriegman, N. Ahuja, Detecting faces in images: a survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence In Pattern Analysis and Machine Intelligence*, vol. 24 (1), pages 34-58, 2002.

[CraTB92] I. Craw, D. Tock, and A. Bennett, Finding face features, *Proceedings of the 2nd European Conference Computer Vision (ECCV)*, Santa Margherita Ligure, Italy, pages 92-96, 1992.

[SchK98] H. Schneiderman and T. Kanade, Probabilistic modeling of local appearance and spatial relationships for object recognition, *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition*, pages 45-51, Santa Barbara, CA, USA, 1998.

[RB&TP 93] R. Brunelli and T. Poggio. "Face recognition: features versus templates". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.15,no.10,pp 1042-1052, October 1993.

[DS&NB05] D.SAIGAA,N. BENOUDJIT, K. BENMAHAMED ,S.LELANDAIS AUTHENTIFICATION D'INDIVIDUS PAR RECONNAISSANCE DE VISAGES, *Courrier du Savoir – N°06*, Juin 2005, pp.61-66 (jeudi 20 mars 2014, 10:52:30).

[hmue11] <http://mplab.ucsd.edu>, The MPLab GENKI Database, GENKI-4K Subset, 2011.

[PV,MJJ 04] P. Viola, M. J. Jones, Robust real-time face détection, *International Journal of Computer Vision* 57 (2004) 137–154.

[VR,TS et al 12] Vincent Rapp, Thibaud Senechal, Kevin Bailly, Lionel Prevost Machine à Vecteurs Supports Multi-Noyau pour la détection de points caractéristiques du visage.

[CS13] THESE *Présentée par* : Catherine SOLADIE, Représentation Invariante des Expressions Faciales Application en Analyse Multimodale des E motions Soutenue le 13 décembre 2013 (dimanche 9 février 2014, 16:51:24).

[ML,SA et al98] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pp. 200–205, 1998.

[CS,SG et al05] C. Shan, S. Gong, and P. W. McOwan, "Robust facial expression recognition using local binary patterns," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2, pp. II–370, 2005.

[L.G10] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029{2054, 2010.

[D.K&C.T12] D. Kressner and C. Tobler. htucker { a matlab toolbox for tensors in hierarchical Tucker format. Technical Report 2012-10, Seminar for Applied Mathematics, ETH Zurich, 2012. Available at <http://anchp.epfl.ch>.

[V.R&T.S11] Vincent Rapp, Thibaud Senechal, Hanan Salam, Lionel Prevost, Renaud Seguier, Kevin Bailly, Combinaison de Descripteurs Hétérogènes pour la Reconnaissance de Micro-Mouvements Faciaux.2011.

[T.S,K.B et al 10] T. Senechal, K. Bailly, and L. Prevost, "Automatic facial action detection using histogram variation between emotional states," in *Proc. Int'l Conf. Pattern Recognition (ICPR'10)*, 2010.

Annexe

[A.M09] THESE *Présenté par* Anouar Mellakh Reconnaissance des visages en conditions dégradées 07 Avril 2009)10 février 2014, 17:22:47)

[Mes99] K. Messer, J. Matas, J. Kittler, J. Luetttin, G. Maitre, ' "XM2VTSDB: The Extended M2VTS Database", *Proc. AVBPA'99 Washington DC*, 1999.

[LM98] J. Luetttin, G. Maître, "Evaluation protocol for the extended M2VTS database (XM2VTSDB) ", in *IDIAP Communication 98-05*, IDIAP, Switzerland, 1998.

[PMRR00] P. J. Philips, H. Moon, S. A. Rizvi, P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.22, no.10, pp.1090-1100, 2000.

[Sit4] Le site web de la base de données Color FERET, <http://www.nist.gov/humanid/colorferet>

[MB98] A.M. Martinez, R. Benavente, "The AR face database", Technical Report 24, Computer Vision Center, Espagne, 1998.

[Sit5] CVL Face Database, <http://www.lrv.fri.uni-lj.si/facedb.html>.

[A.C09] THESE *Présenté par* Anis CHAARI Nouvelle approche d'identification dans les bases de données biométriques basée sur une classification non supervisée 6 Octobre 2009 (14 février 2014, 11:02:06)

Bibliographie

2014

[Sit6] Téléchargement de la base de données de visage
http://web.mit.edu/emeyers/www/face_databases.html#orl