

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique Et Populaire

Ministère de l'Enseignement Supérieur et de
la Recherche Scientifique

Université 20 août 1955 –Skikda-



وزارة التعليم العالي و البحث العالي

جامعة 20 أوت 1955 – سكيكدة-

Ref : D012120009D

FACULTE DE TECHNOLOGIE

THESE

Pour l'obtention du Diplôme de

DOCTORAT

Filière : Electronique

Option : Communication et Traitement de l'information

Présentée Par :

CHERGUI Abdelhakim

THEME

Vérification de la parenté par des images faciales

Soutenue le 09/07/2020 devant le Jury:

| | | | |
|----------------------|------------|---------------------|---------------|
| BELMEGUENAI Aissa | Professeur | Univ. Skikda | Président |
| OUCHTATI Salim | MCA | Univ. Skikda | Rapporteur |
| SEQUEIRA Jean | Professeur | Univ. AIX Marseille | Co-rapporteur |
| DEROUICHE Nadir | MCA | Univ. Skikda | Examineur |
| SAMAI Djamel | MCA | Univ. Ouargla | Examineur |
| MAVROMATIS Sébastien | MCF HDR | Univ. AIX Marseille | Invité |

Année universitaire : 2019-2020

العنوان: التحقق من القرابة من خلال صور الوجه

ملخص

يعتبر التحقق من القرابة من خلال صور الوجه أحد أكثر مجالات البحث نشاطاً في القياسات الحيوية ورؤية الكمبيوتر ، ولقد كان محل اهتمام متزايد في السنوات الأخيرة. إنها تقنية تهدف إلى استغلال خصائص الوجه للتعرف على درجة القرابة بين شخصين من خلال صور وجوههم.

أدت البحوث في مجال التحقق من القرابة إلى ظهور تطبيقات عديدة و متعددة ، فهو يشكل جوهر العديد من الأنظمة التشغيلية بالفعل ، هذه الأخيرة، يمكن أن تتراوح من أنظمة بسيطة مثل أنظمة تنظيم ألبوم الصور والبحوث التاريخية والأنساب ، وصولاً إلى أنظمة أكثر أهمية و أكثر تعقيداً ، مثل أنظمة البحث عن أفراد الأسرة المفقودين ، وتحديد الأشخاص المطلوبين وأنظمة تهريب الأطفال.

يدخل عملنا هذا في إطار التحقق من القرابة من خلال صور الوجه، حيث نقترح نظاماً للتحقق من القرابة الذي يتلقى عند مدخله زوجاً من صور الوجه (آباء، أبناء) لتحديد ما إذا كان شخصان لهما علاقة قرابة أم لا. تتضمن المقاربة المقترحة ست مراحل:

1. المعالجة المسبقة لصورة الوجه من أجل الحصول على صور وجه مصطفة و مؤطرة.
2. استخراج الميزات استناداً إلى واصفات النسيج ونماذج التعلم العميق
3. تمثيل الوجه باستخدام الهرم المتعدد المستويات وهذا لزيادة عدد الميزات
4. وظيفة تمثيل الميزات جنباً إلى جنب مع تطبيع الميزات
5. تقليل الميزات (الإسقاط أو التحديد) للاحتفاظ بأفضل الميزات
6. القرار بوجود صلة قرابة أم لا ، وذلك باستخدام ناقل مصنف لدعم الآلة (SVM).

تم اختبار المقاربة المقترحة على خمس قواعد بيانات مرجعية (Cornell KinFace, UB KinFace, Family101, KinFace W-I et KinFace W-II). في كل مرحلة، قمنا بإجراء العديد من التجارب، من أجل تحديد أفضل و أنسب المعايير. مقارنة الطريقة المقترحة مع الطرق المتقدمة الأخرى تبين بشكل جلي بأن النتائج المتحصل عليها هي أفضل وأحسن

الكلمات المفتاحية: التحقق من القرابة، تمثيل الوجه؛ تمثيل الخصائص؛ تخفيض الخصائص؛

التصنيف

Abstract

The kinship verification through facial images is one of the most active areas of research in biometrics and computer vision, and has received increasing attention in recent years. It is a technique which aims to exploit the characteristics of the face to recognize the degree of kinship of two individuals from their facial images.

The Kinship verification has given rise to many and multiple applications, it constitutes the core of several already operational systems, these can range from simple systems such as photo album organization systems and historical and genealogical research, up to " to larger and more complicated systems, such as missing family member systems, identification of wanted individuals and child smuggling systems.

The researches in the field of kinship verification has given rise to many and multiple applications, it constitutes the core of several already operational systems, these can range from simple systems such as photo album organization systems and historical and genealogical research, up to more important and complicated systems, such as missing family member systems, identification of wanted persons and child smuggling systems.

Our work is part of the kinship verification through facial images where we propose a kinship verification system which receives at its entry a pair of facial images (parents, children) to determine at its exit if two persons have a kin relation or not. The proposed approach involves six steps:

1. The preprocessing of the facial image in order to obtain aligned and cropped facial images.
2. Feature extraction based on the texture descriptors and deep learning models.
3. The face representation using the multi-level pyramid (PML) to increase the features number.
4. The feature representation function alongside with a features normalization.
5. The features reduction (projection or selection) to retain the best discriminative features.
6. The decision whether there is a kinship or not, and this, using a classifier (SVM) vector of machine support.

The proposed approach has been tested on five reference databases (Cornell KinFace, UB KinFace, Family101, KinFace W-I and KinFace W-II). For each step, we carried out several experiments, in order to determine the best and most appropriate parameters. A comparison of the proposed method with other advanced methods clearly shows that the results obtained are clearly better and good.

Keywords: Kinship verification, face representation, feature representation, feature reduction, Classification.

Titre : La vérification de la parenté à travers des images faciales

Résumé

La vérification de parenté à travers des images faciales est parmi les domaines de recherche les plus actifs en biométrie et en vision par ordinateur, elle a été l'objet d'une attention croissante au cours de ces dernières années. C'est une technique qui vise à exploiter les caractéristiques du visage pour reconnaître le degré de parenté de deux individus à partir de leurs images faciales.

Les recherches dans le domaine de la vérification de parenté ont donné lieu à de nombreuses et multiples applications, elle constitue le noyau de plusieurs systèmes déjà opérationnels, ces derniers, peuvent aller des systèmes simples comme les systèmes d'organisation d'album photo et de recherche historique et généalogique, jusqu'à des systèmes plus importants et plus compliqués, tels que les systèmes de recherche des membres de famille disparus, de l'identification des personnes recherchées et les systèmes de contrebande d'enfants.

Notre travail s'inscrit dans le cadre de la vérification de parenté à travers des images faciales où nous proposons un système de vérification de parenté qui reçoit à son entrée une paire d'images faciales (parents, enfants) pour déterminer à sa sortie si deux personnes ont ou non une relation de parenté. L'approche proposée comporte six étapes :

1. Le prétraitement de l'image du visage dans le but d'obtenir des images faciales alignées et recadrées.
2. L'extraction des caractéristiques en se basant sur des descripteurs de texture et des modèles d'apprentissage en profondeur.
3. La représentation du visage à l'aide de la pyramide multi-niveaux (PML) pour augmenter le nombre de caractéristiques.
4. La représentation des caractéristiques jumelées avec leur normalisation.
5. La réduction des caractéristiques (projection ou sélection) pour conserver les meilleures caractéristiques.
6. La décision de l'existence d'une parenté ou non, et ce, en utilisant un classifieur (SVM) vecteur de support de machine.

L'approche proposée a été testée sur cinq bases de données de référence (Cornell KinFace, UB KinFace, Family101, KinFace W-I et KinFace W-II). Pour chaque étape, nous avons mené plusieurs expériences, et ce, dans le but de déterminer les meilleurs paramètres et les plus appropriés. La comparaison de la méthode proposée avec d'autres méthodes de pointe montre clairement que les résultats obtenus sont nettement meilleurs et bons.

Mots Clés : vérification de la parenté, représentation du visage, Représentation de caractéristiques, réduction de caractéristiques, Classification.

Liste des Publications

1. **Abdelhakim Chergui**, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche, Jean Sequeira, and Fadi Dornaika "Kinship verification through facial images using multiscale and multilevel handcrafted features," Journal of Electronic Imaging 29(2), 023017 (1 April 2020). <https://doi.org/10.1117/1.JEI.29.2.023017>
2. **Abdelhakim Chergui**, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche, Mohamed Lashab, Jean Sequeira (2020). "Kinship verification through facial images using CNN-based features". Traitement du Signal, Vol. 37, N. 1, pp. 1-8. DOI: 10.18280/ts.370101. <http://www.iieta.org/journals/ts/paper/10.18280/ts.370101>
3. Salim Ouchtati, **Abdelhakim Chergui**, Sébastien Mavromatis, Belmeguenai Aissa, Djemil Rafik, Jean Sequeira (2019). "Novel method for brain tumor classification based on use of image entropy and seven Hu's invariant moments". Traitement du Signal, Vol. 36, No. 6, pp. 483-491. <https://doi.org/10.18280/ts.360602>

Liste des Communications Internationales

1. **Abdelhakim Chergui**, Salim Ouchtati, Hichem TELLI, et al. "Lpq and ldp descriptors with ml representation for kinship verification". In: The second edition of the International Workshop on Signal Processing Applied to Rotating Machinery Diagnostics (SIGPROMD'2018). 2018. p. 1-10.
2. **Abdelhakim. Chergui**, Salim Ouchtati, Jean Sequeira, Salah Eddine Bekhouche, Faouzi Bougourzi "Kinship verification using BSIF and LBP," 2018 International Conference on Signal, Image, Vision and their Applications (SIVA), Guelma, Algeria, 2018. DOI: 10.1109/SIVA.2018.866108. <https://ieeexplore.ieee.org/document/8661085>
3. **Abdelhakim. Chergui**, Salim Ouchtati, Jean Sequeira, Salah Eddine Bekhouche, Bougourzi, F., Benlamoudi, A. (2018). "Discriminant analysis for facial verification using colorimages". In the Proceeding of the first International Conference on Electrical Engineering ICEEB18.
4. **Abdelhakim. Chergui**, Salim Ouchtati, Jean Sequeira *et al.* Robust kinship verification using local descriptors. In: *Proceeding of Third International Conference On Advanced Technologies And Electrical Engineering ICTAEE18*. 2018.
5. **Abdelhakim. Chergui**, Salim Ouchtati, Jean Sequeira, Salah Eddine Bekhouche, Faouzi Bougourzi and Hichem Telli, "Deep Features for Kinship Verification from Facial Images," 2019 International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia, 2019, pp. 64-67. DOI: 10.1109/ASET.2019.8871011. <https://ieeexplore.ieee.org/document/8871011>
6. **Abdelhakim. Chergui**, Salim. Ouchtati, Sébastien. Mavromatis, Salah Eddine Bekhouche, Jean Sequeira and Houssem. Zerrari, "Kinship Verification using Mixed Descriptors and Multi Block Face Representation," 2019 International Conference on Networking and Advanced Systems (ICNAS), Annaba, Algeria, 2019, pp. 1-6. DOI: 10.1109/ICNAS.2019.8807875. <https://ieeexplore.ieee.org/document/8807875>
7. **Abdelhakim Chergui**, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche and Jean Sequeira "Investigating Deep CNNs Models Applied in Kinship Verification through Facial Images," 2019 5th International Conference on Frontiers of Signal Processing (ICFSP), Marseille, France, 2019, pp. 82-87. DOI:10.1109/ICFSP48124.2019.8938055. <https://ieeexplore.ieee.org/document/8938055>

DEDICACE

Avec l'expression de ma reconnaissance, je dédie ce modeste travail :

- *A mes chers parents, Qui n'ont jamais cessé, de formuler des prières à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs.*
- *A ma femme et mes enfants (Mohamed et Takki)*
- *A mes frères, (Zouhir et Samir) pour leur appui et leur encouragement.*
- *À mes chères sœurs et à leurs maris. Pour leur soutien moral et leurs précieux conseils tout au long de mes études.*
- *A mes chers amis, (Lamine, Mohamed, Amine, Salah, Faïd.....), Pour leurs aides et supports dans les moments difficiles.*
- *A toute ma famille,*
- *A tous mes autres ami(e)s,*
- *A tous ceux que j'aime et ceux qui m'aiment.*

Remerciements

Je tiens à exprimer mes vives reconnaissances et mes sincères remerciements à Mr. Salim OUCHTATI Maître de Conférences « A » à l'université 20 Août 1955 -Skikda et Mr. Jean SEQUEIRA Professeur à l'université d'Aix Marseille – France-, respectivement, mon directeur et mon codirecteur de thèse d'avoir accepté de me proposer ce thème de recherche et pour l'aide et l'orientation qu'ils m'ont accordées tout au long de la préparation de cette thèse. Leurs encouragements, leur grande compétence, leur rigueur intellectuelle et leurs conseils avisés ont été très prépondérants pour la réussite de cette thèse.

Mes sincères remerciements vont aussi à Mr. Aissa BELMEGUENAI Professeur à l'université 20 Août 1955 –Skikda d'avoir bien voulu me faire l'honneur d'accepter de présider le jury de cette thèse.

Je voudrais également exprimer ma profonde reconnaissance pour Mr. Nadir DEROUICHE Maître de Conférences « A » à l'université 20 Août 1955 -Skikda et Mr. Djamel SAMAI Maître de Conférences « A » à l'université Kasdi Merbah Ouargla pour l'honneur qu'ils m'ont fait en acceptant de participer à ce jury et d'être examinateurs de mon travail.

Je tiens également à présenter mes sincères remerciements à Mr. Sébastien MAVROMATIS Maître de Conférences Habilité à Diriger les Recherches à l'université d'Aix Marseille –France- pour toute l'assistance précieuse qu'il m'a accordée afin que ce modeste travail soit réalisé et de m'avoir honoré de faire partie de ce jury en tant que membre invité.

Je voudrais aussi exprimer ma profonde gratitude et mes sincères reconnaissances à Mr. Fadi DORNAIKA Professeur à l'université du Pays Basque en Espagne et Salah Eddine BEKHOUCHE Maître de Conférences « B » à l'Université de Djelfa qui m'ont beaucoup aidé et conseillé et sans eux ce travail n'aurait pas vu le jour.

Finalement, je ne peux qu'à adresser mes vifs remerciements à tous les membres du Laboratoire de Recherche en Electronique de Skikda (LRES) de m'avoir assuré les moyens humains et matériels nécessaires pour la réalisation de cette thèse.

Table des matières

| | |
|--------------------------------|------|
| ملخص | i |
| Abstract | ii |
| Résumé | iii |
| Liste des publications | iv |
| Liste des Communications | iv |
| Dédicaces | v |
| Remerciements | vi |
| Table des matières | vii |
| Liste des figures | xi |
| Liste des tableaux | xiii |
| Abréviations | xiv |

Introduction générale

| | |
|---------------------------------------|----|
| 1 Introduction | 01 |
| 2 Problématique | 02 |
| 3 Contributions de la recherche | 03 |
| 4 Organisation de la thèse | 04 |

Chapitre 1 : Systèmes de vérification de la parenté : Aperçu global et Etat de l'art

| | |
|--|----|
| 1.1 Introduction | 06 |
| 1.2 Généralités sur la biométrie | 06 |
| 1.3 Systèmes d'analyse des images faciales | 08 |
| 1.4 Vérification de la parenté | 09 |
| 1.4.1 Généralités et définitions | 09 |
| 1.4.2 Structure du système de vérification de la parenté | 10 |
| 1.4.3 Applications | 11 |
| 1.5 Etat de l'art de la vérification de la parenté | 12 |

| | | |
|-------|---|----|
| 1.5.1 | Approches basées sur l'image-texture | 12 |
| 1.5.2 | Approches basées sur l'apprentissage multiple | 13 |
| 1.5.3 | Approches basées sur l'apprentissage en profond | 15 |
| 1.6 | Principales bases de données | 16 |
| 1.6.1 | Bases de données basées sur les images faciales | 16 |
| 1.6.2 | Bases de données basées sur les vidéos | 22 |
| 1.7 | Conclusion | 25 |

Chapitre 2 : Prétraitement et extraction des caractéristiques d'une image faciale

| | | |
|-------|---|----|
| 2.1 | Introduction | 26 |
| 2.2 | Prétraitement du visage | 26 |
| 2.2.1 | Détection du visage | 26 |
| 2.2.2 | Détection des yeux | 28 |
| 2.2.3 | Correction de position..... | 28 |
| 2.3 | Extraction des Caractéristiques | 30 |
| 2.3.1 | Descripteurs de texture | 30 |
| 2.3.2 | Descripteurs d'apprentissage profond | 41 |
| 2.4 | Représentation du visage | 45 |
| 2.4.1 | Multi Bloc (MB) | 45 |
| 2.4.2 | Multi Niveaux (ML)..... | 46 |
| 2.4.3 | Pyramide Multi Niveau (PML) | 46 |
| 2.4.4 | Combinaison (Descripteurs / Représentation faciale) | 47 |
| 2.5 | Conclusion | 47 |

Chapitre 3 : Décision finale de la vérification de la parenté

| | | |
|-------|--|----|
| 3.1 | Introduction | 48 |
| 3.2 | Représentation et Normalisation des caractéristiques | 48 |
| 3.2.1 | Représentation des caractéristiques | 48 |
| 3.2.2 | Normalisation des caractéristiques | 49 |

| | |
|--|-----------|
| 3.3 Réduction des caractéristiques | 49 |
| 3.3.1 Projection des caractéristiques | 50 |
| 3.3.2 Sélection des caractéristiques | 53 |
| 3.4 La Classification | 56 |
| 3.4.1 Similarité en Cosins | 56 |
| 3.4.2 Méthode des k plus proches voisins | 57 |
| 3.4.3 Machine à vecteurs de support (SVM) | 60 |
| 3.5 Validation croisée | 63 |
| 3.6 Performance d'un système biométrique | 64 |
| 3.6.1 Précision d'un système biométrique (Pr) | 64 |
| 3.6.2 Caractéristiques de fonctionnement du récepteur (ROC) | 64 |
| 3.7 Conclusion | 66 |

Chapitre 4 : Résultats et analyse

| | |
|--|-----------|
| 4.1 Introduction | 67 |
| 4.2 Approche Proposée | 67 |
| 4.3 Effet de descripteurs | 69 |
| 4.3.1 Descripteurs de texture | 69 |
| 4.3.2 Descripteurs d'apprentissage en profond | 70 |
| 4.3.3 Fusion des descripteurs | 71 |
| 4.4 Effet de la représentation du visage (représentation faciale) | 73 |
| 4.5 Effet de la représentation des caractéristiques | 75 |
| 4.6 Effet de la Réduction des caractéristiques (Projection et Sélection) | 76 |
| 4.6.1 Sélection des caractéristiques | 76 |
| 4.6.2 projections des caractéristiques | 78 |
| 4.6.3 Combinaison entre la sélection et la projection des caractéristiques..... | 79 |
| 4.7 Effet des Classifieurs | 79 |
| 4.8 Temps de calcul (CPU et Nombre des caractéristiques) | 80 |

| | |
|--|-----------|
| 4.9 Performances obtenues pour les différentes relations de parenté | 82 |
| 4.10 Courbes caractéristiques de fonctionnement du récepteur (ROC) | 83 |
| 4.11 Comparaison des approches existantes avec Notre approche | 84 |
| 4.12 Conclusion | 86 |

Conclusion générale

| | |
|----------------------------|-----------|
| 1 Conclusion | 87 |
| 2 Perspective | 88 |

Bibliographie

| | |
|----------------------------|-----------|
| Bibliographie | 89 |
|----------------------------|-----------|

Liste des figures

| Figure | Titre | N^o |
|--------------------|--|----------------------|
| Figure 1.1 | Les principales caractéristiques biométriques | 07 |
| Figure 1.2 | Structure générale d'un system de la vérification de parenté par image faciale | 11 |
| Figure 1.3 | Exemples d'images de la base de données « Cornell Kinship » | 17 |
| Figure 1.4 | Exemples d'images de la base de données « UB KinFace » | 18 |
| Figure 1.5 | Exemples d'images de la base de données « IITD Kinship » | 19 |
| Figure 1.6 | | 20 |
| Figure 1.7 | Exemples d'images de la base de données « KinFaceW-I » | 20 |
| Figure 1.8 | Exemples d'images de la base de données « KinFaceW-II » | 21 |
| Figure 1.9 | Exemples d'images de la base de données « TS KinFace » | 21 |
| Figure 1.10 | Exemples d'images de la base de données « WVU Kinship » | 22 |
| Figure 1.11 | Exemples d'images de la base de données « UvA-NEMO Smile » | 23 |
| Figure 1.12 | Exemples d'images de la base de données « KFVW » | 24 |
| Figure 1.13 | Exemples d'images de la base de données « KIVI » | 24 |
| Figure 2.1 | Correction de position de visage | 29 |
| Figure 2.2 | Construction d'un motif binaire et calcul du code LBP | 31 |
| Figure 2.3 | LBP multi-échelle. Exemple de voisinages obtenus pour trois valeurs de (P,R) | 32 |
| Figure 2.4 | Les étapes nécessaires du descripteur LTP | 33 |
| Figure 2.5 | Masques de bord Kirsch (LDP) à huit directions | 33 |
| Figure 2.6 | Les étapes nécessaires du descripteur LDP | 34 |
| Figure 2.7 | les étapes nécessaires du descripteur LPQ | 35 |
| Figure 2.8 | les étapes nécessaires du descripteur BSIF | 36 |
| Figure 2.9 | les étapes nécessaires du descripteur WLD | 38 |

| Figure | Titre | N° |
|--------------------|---|-----------|
| Figure 2.10 | Procédure simplifiée pour le descripteur HOG | 40 |
| Figure 2.11 | Couches des Modèles CNN (VGG-16 et VGG-Face) | 43 |
| Figure 2.12 | Couches des modèles CNN (ResNet) | 44 |
| Figure 2.13 | Exemple MB à quatre niveaux | 45 |
| Figure 2.14 | Exemple ML à quatre niveaux | 45 |
| Figure 2.15 | Exemple PML à quatre niveaux | 46 |
| Figure 2.16 | Exemple d'extraction de caractéristiques utilisant le descripteur LBP avec la représentation facial PLM | 46 |
| Figure 3.1 | Représentation des paramètres nécessaires à la méthode K -ppv | 57 |
| Figure 3.2 | Exemple d'hyperplan optimal, vecteurs de support et marge maximale | 62 |
| Figure 3.3 | Principe de la validation croisée | 64 |
| Figure 3.4 | La distribution du taux de vraisemblances | 65 |
| Figure 3.5 | La courbe de Caractéristiques de fonctionnement du récepteur (ROC) | 66 |
| Figure 4.1 | Structure générale de l'approche proposée | 68 |
| Figure 4.2 | La précision (%) et le nombre de caractéristiques aux différents niveaux de la représentation du visage | 74 |
| Figure 4.3 | Précision obtenue (%) en utilisant différents schémas de sélection des caractéristiques | 77 |
| Figure 4.4 | La précision (%) et le temps CPU (formation / test) en fonction de nombre de caractéristiques | 81 |
| Figure 4.5 | ROC de différentes effets dans la base de données KinFace W-II | 84 |

Liste des Tableaux

| Tableau | Titre | N^o |
|---------------------|--|----------------------|
| Tableau 1.1 | Résumé sur les bases de données de la vérification de parenté | 25 |
| Tableau 4.1 | Résultats obtenus pour chaque descripteur sur les bases de données utilisées | 70 |
| Tableau 4.2 | Résultats obtenus pour les descripteurs d'apprentissage en profondeur. | 71 |
| Tableau 4.3 | Performances du système avec les descripteurs LBP, BSIF et LPQ et la fusion entre eux. | 72 |
| Tableau 4.4 | Performances du système avec les descripteurs ResNet-18, ResNet-50 et ResNet-101 et la fusion entre eux. | 73 |
| Tableau 4.5 | Performances des différents schémas de fusion de paires pour chaque descripteur | 76 |
| Tableau 4.6 | Performances obtenues en utilisant la sélection des caractéristiques | 77 |
| Tableau 4.7 | Performances obtenues en utilisant la projection des caractéristiques | 78 |
| Tableau 4.8 | Performances obtenues en utilisant la combinaison entre la sélection et la projection | 79 |
| Tableau 4.9 | Performances obtenues en utilisant les trois (Cosin, KNN, SVM) | 80 |
| Tableau 4.10 | Performances obtenues et leurs CPU A et CPU T | 81 |
| Tableau 4.11 | Performances obtenues utilisant les différentes relations de parenté | 82 |
| Tableau 4.12 | Comparaison de l'approche proposée avec d'autres approches existantes | 85 |

Abréviations

| | |
|--|--|
| BSIF: Binarized Statistical Image Features | ML: Multi level |
| CNN: Convolutional Neural Networks | MNRML : Multiview Neighborhood Repulsed Metric Learning |
| CPU: Central Processing Unit | NRML : Neighborhood Repulsed Metric Learning |
| DFT: Discrete Fourier Transform | PCA: Principal Component Analysis |
| DMML : Discriminative Multimetric Learning | PDFL : Prototype Discriminative Feature Learning |
| ESL : Ensemble Similarity Learning | PML: Pyramide Multi Level |
| FA: False-Acceptance | PSM : Pictorial Structure Model |
| FAR: False Accept Rate | ResNet: Residual network |
| FR: False-Rejection | ROC: Receiver Operating Characteristic |
| FRR: False Reject Rate | SIFT: Scale Invariant Feature Transform |
| HOG: Histograms of Oriented Gradient | SSD : Self-Similarity Descriptor |
| KNN: K Nearest Neighbor | STFT: Short-Time Fourier Transform |
| KL: Kullback-Leibler | SVM : Support vector machines |
| LBP: Local Binary Patterns | TEE : Taux d'Égale Erreur. |
| LDA: Linear Discriminant Analysis | VGG : Visual Geometry Group |
| LDP: Local Directional Patterns | WGEML : Weighted Graph Embedding Metric Learning |
| LSDA : Locality Sensitive Discriminant Analysis | WLD : Weber Local Descript |
| LTP: Local Ternary Patterns | WVU : West Virginia University |
| LPP: The Locality Preserving Projections | |
| LPQ: Local Phase Quantization | |
| MB: Multi block | |

Introduction générale

Sommaire

| | |
|--|-----------|
| 1 Introduction | 01 |
| 2 Problématique | 02 |
| 3 Contributions de la recherche | 03 |
| 4 Organisation de la thèse | 04 |

1. Introduction

Depuis la fin de la seconde guerre mondiale, les avancées technologiques ne cessent de s'accélérer et de créer de nouvelles opportunités dans de nombreux domaines. Les caméras de surveillance, la géolocalisation, les drones, les robots intelligents, les réseaux sociaux, etc. sont des innovations technologiques qui ont révolutionné notre quotidien et bouleversé nos habitudes.

L'un des domaines qui ont connu les progrès les plus importants ces dernières années est la sécurité. Depuis les attentats du 11 septembre 2001, Madrid 2004 et Londres 2005, plusieurs États ont décidé de renforcer leurs dispositifs de sécurité en mettant en place une série de mesures pour lutter contre le terrorisme et la criminalité. L'une de celles-ci concerne l'exploitation des caractéristiques physiques des individus, c'est la biométrie [1].

La biométrie est la science qui porte sur l'analyse des caractéristiques propres à chaque individu, permettant l'authentification de son identité. Elle consiste à identifier une personne à partir de ses caractéristiques physiques ou comportementales. Le visage, les empreintes digitales, l'iris, etc. sont des exemples des caractéristiques physiques. La voix, l'écriture, le rythme de frappe sur un clavier, etc. sont des caractéristiques comportementales. Au sens littéral et de manière plus simplifiée, la biométrie signifie la « mesure du corps humain » [2].

La pertinence d'une caractéristique biométrique dépend de son unicité, de sa stabilité dans le temps ou de sa résistance à des éléments extérieurs (stress, bruit, lumière, etc.). En outre, elle doit être universelle (présente chez chaque individu), enregistrable et mesurable [3].

La biométrie repose sur des algorithmes statistiques. Elle ne peut donc être fiable à 100 % quand elle est mise en œuvre seule. Depuis plusieurs années, la combinaison de plusieurs biométries entre elles, appelée biométrie multimodale, par exemple le visage et l'iris ou l'iris et les empreintes digitales, a permis de réduire considérablement les taux d'erreur [4].

Le visage humain présente des caractéristiques universelles, telles que le genre, l'âge et les émotions, par conséquent, les images faciales peuvent être facilement exploitées dans de nombreuses applications à base d'analyse faciale. Parmi celles-ci, on trouve la vérification de la parenté par des images faciales.

Les images faciales peuvent indiquer la relation de parenté. On peut déterminer le lien de parenté entre les parents et les enfants lorsqu'ils se ressemblent, bien qu'il soit difficile de déterminer avec certitude si deux personnes sont parentes. Ainsi, la vérification de la parenté

à l'aide des images du visage a vu le jour dans le but d'exploiter les caractéristiques implicites que cette relation établit sur les images de deux personnes d'une même famille, et d'offrir une nouvelle alternative au test ADN parental, lorsque ce dernier n'est pas disponible [5].

Les applications potentielles de cette technique sont nombreuses ; citons entre autres la contribution au rapprochement social, la recherche des enfants perdus, les recherches généalogiques et l'analyse des échanges sur les réseaux sociaux (social media mining)

Dans ce but nous faisons appel aux techniques de traitement d'images et à l'apprentissage automatique (Machine Learning), un sous-domaine de l'intelligence artificielle.

Le traitement d'images est une science récente qui a pour but d'offrir aux spécialistes de différents domaines des outils de manipulation de ces données digitales issues du monde réel. De ce fait, le traitement de l'image numérique est l'utilisation d'algorithmes informatiques pour créer, traiter, transmettre et afficher des images numériques. Les algorithmes de traitement d'images numériques peuvent être utilisés pour [6]:

- Convertir les signaux provenant d'un capteur d'image en images numériques.
- Améliorer la clarté, supprimer le bruit et d'autres effets de parasite.
- Extraire la taille, l'échelle ou le nombre d'objets dans une scène.
- Préparer les images pour l'affichage ou l'impression.
- Compresser les images pour la communication à travers un réseau.

L'apprentissage automatique a comme objectif de comprendre la structure des données grâce aux algorithmes. Pour cette raison, l'apprentissage automatique se sert des algorithmes qui facilitent l'utilisation des ordinateurs dans la construction de modèles à partir des données d'échantillonnage afin d'automatiser les processus de prise de décision en fonction des données saisies [7].

2. Problématique

La ressemblance entre les visages humains a été généralement reconnue comme un élément important dans la reconnaissance de la parenté entre les parents et les enfants ; en effet, les images faciales véhiculent de nombreuses caractéristiques humaines importantes, telles que l'identité, le sexe, l'expression etc.

Pour vérifier la relation de parenté à partir des images faciales, nous sommes confrontés à nombreuses difficultés :

- Quelles sont les méthodes nécessaires à l'extraction des caractéristiques des images faciales pour la vérification de la parenté ?
- Comment utiliser ces caractéristiques pour prendre des décisions, autrement dit, quelles sont les méthodes de classification efficaces qui donnent de bons résultats pour la vérification de la parenté ?
- La variation de l'âge, du sexe, de l'ethnicité dans les images faciales (Parents / Enfants).
- Les images faciales utilisées (Parents / Enfants) proviennent-elles de la même image ou sont-elles coupées de deux images différentes ?
- En d'autres parts, les images faciales (Parents/Enfants) sont-elles soumises aux mêmes conditions ? (luminosité, arrière-plan, précision, l'angle de vision, variation de pose, l'état d'éclairage, etc.)
- Les changements entre les images (Parents/Enfants) dans l'expression du visage, la coiffure, et les effets du vieillissement.
- Les ressemblances entre les images faciales des personnes sans relation de parenté réelle.
- La différence des attributs faciaux et l'absence des ressemblances entre les personnes d'une même famille et d'une même relation de parenté.

Pour répondre à ces questions, nous proposons quelques contributions présentées dans la partie suivante.

3. Contributions de la recherche

Les principales contributions de notre travail sont les suivantes.

- L'évaluation des performances des différents descripteurs : (LBP, LTP, LDP, LPQ, BSIF, WLD, HOG, VGG-Face, VGG-16, ResNet), ainsi que les influences d'autres étapes en combinaison avec les descripteurs sur l'étape d'extraction des caractéristiques.
- L'étude de l'efficacité des trois différentes représentations de visage lorsqu'elles sont combinées avec les descripteurs de texture cités ci-dessus. Le nombre d'entités extraites peut être augmenté en variant la disposition géométrique de la représentation de l'image du visage en (MB, ML et PML).
- La proposition d'un schéma de fusion de paires permettant la fusion des caractéristiques des paires d'images et montrer l'efficacité de cette fusion.
- L'introduction d'un score robuste innovant pour le classement des caractéristiques. Ce score est basé sur l'utilisation de la relation entre deux poids connus. Le nouveau score peut encore améliorer le résultat du système de vérification de parenté.

- La proposition d'une nouvelle méthode pour la réduction des caractéristiques. Cette méthode est basée sur la fusion entre le score proposé pour la sélection des caractéristiques et PCA pour la projection des caractéristiques. Cette méthode nous permet d'avoir une réduction robuste et ne garder que les meilleures caractéristiques.
- De nombreuses expériences sont effectuées sur cinq bases de données publiques (Cornell Kinface, UB KinFace, Family 101, KinFace W-I et KinFace W-II). L'analyse des résultats obtenus montre que l'approche proposée est nettement meilleure et efficace par rapport à la plupart des approches de pointe.

4. Organisation de la thèse

Notre thèse est composée d'une introduction générale, quatre chapitres principaux et une conclusion générale organisés comme suit :

- L'introduction générale (partie actuelle) où nous présentons un aperçu général sur la biométrie et en particulier les systèmes de la vérification de la parenté ainsi que les différentes difficultés rencontrées dans ce domaine. Nous terminons cette partie par la présentation de la contribution que nous avons apportée à travers ce travail ainsi que l'organisation générale de notre thèse.
- Le premier chapitre est consacré à la présentation des systèmes biométriques et en particulier ceux basés sur l'analyse et le traitement des images faciales. Ensuite, notre présentation s'est étendue vers l'étude des systèmes de la vérification de la parenté, leurs structures ainsi que leurs différentes applications. Puis, nous présentons l'état de l'art de la vérification de la parenté à partir des images faciales, ou nous avons exposé les principaux travaux réalisés et les différentes approches proposées dans ce domaine, spécialement celles basées sur la texture d'image, l'apprentissage multiple et l'apprentissage profond. Nous terminons ce chapitre par la présentation des principales bases de données utilisées dans ce domaine (images faciales, vidéo).
- Dans le deuxième chapitre, nous exposons les étapes nécessaires pour l'extraction des caractéristiques. Nous présentons dans un premier temps, les différentes opérations de prétraitement du visage (détection de visage, détection des yeux et la correction de la position). Ensuite nous présentons le principe de fonctionnement de chaque descripteur de caractéristiques (LBP, LTP, LDP, BSIF, LPQ, WLD, HOG, VGG-16, VGG-Face, ResNet) à partir des bases mathématiques nécessaires à la compréhension de fonctionnement des algorithmes de ces méthodes. Enfin, nous discutons les trois types de

représentations du visage (MB, ML, PML) et leurs effets pour l'augmentation de caractéristiques.

- Le troisième chapitre présente les trois dernières étapes pour la décision finale pour déterminer la parenté. Dans la première partie nous présentons des équations connues et d'autre proposées pour transformer les paires de vecteurs (parents, enfants) en un seul vecteur représentatif. Ensuite, nous présentons les méthodes les plus significatives de réduction des caractéristiques (la projection et la sélection). Finalement, et pour déterminer l'existence d'une parenté entre les pairs (Parent/Enfant), nous présentons les trois types de classifieurs (Similarité, KNN, SVM) qui sont parmi les classifieurs les plus utilisés pour la classification.
- Dans le quatrième chapitre, nous présentons les détails de l'approche proposée ainsi que notre contribution. Ensuite, nous exposons les résultats expérimentaux obtenus pour chaque méthode en analysons leur performances et l'effet de chaque paramètre sur notre système. Nous terminons ce chapitre par une discussion avec interprétation des résultats.
- Finalement, nous terminons ce manuscrit par une conclusion générale et les perspectives ainsi que les travaux envisagés dans le futur.

Chapitre 1

Systemes de vérification de la parenté : Aperçu global et Etat de l'art

Sommaire

| | |
|--|-----------|
| 1.1 Introduction | 06 |
| 1.2 Généralités sur la biométrie | 06 |
| 1.3 Systemes d'analyse des images faciales | 08 |
| 1.4 Vérification de la parenté | 09 |
| 1.4.1 Généralités et définitions | 09 |
| 1.4.2 Structure d'un systeme de vérification de la parenté | 10 |
| 1.4.3 Applications | 11 |
| 1.5 Etat de l'art de la vérification de la parenté | 12 |
| 1.5.1 Approches basées sur l'image-texture | 12 |
| 1.5.2 Approches basées sur l'apprentissage multiple | 13 |
| 1.5.3 Approches basées sur l'apprentissage en profond | 15 |
| 1.6 Principales bases de données | 16 |
| 1.6.1 Bases de données basées sur les images faciales | 16 |
| 1.6.1 Bases de données basées sur les vidéos | 22 |
| 1.7 Conclusion | 25 |

1.1 Introduction

L'analyse des images faciales est parmi les sujets de recherche les plus importants en traitement d'images. Cette importance est due à la quantité d'informations significatives qu'on peut extraire à partir de l'image d'un visage. Au cours des dernières décennies, les recherches dans le domaine de l'analyse faciale ne sont pas limitées uniquement aux domaines classiques tels que la vérification du visage et les problèmes de reconnaissance des personnes, mais elle se sont étendues vers l'exploration d'un nouveau domaine de recherche basé toujours sur l'analyse des images faciales, à savoir, la vérification de la parenté [8].

La vérification de la parenté est un nouveau domaine de recherche en vision par ordinateur qui fait appel à l'analyse des images faciales et qui a été l'objet d'une attention croissante durant ces dernières années. Elle permet de vérifier la relation entre les personnes en fonction des images faciales, et ce, en exploitant les caractéristiques visuelles du visage [9].

Ce chapitre est consacré à l'étude des systèmes de vérification de la parenté à partir des images faciales, nous commençons par présenter quelques généralités sur la biométrie ainsi que les différentes techniques utilisées dans ce domaine. Ensuite, nous exposons les notions nécessaires pour la compréhension d'un système d'analyse faciale. Puis nous étudions la structure générale ainsi que les principaux objectifs et les applications potentielles d'un système de vérification automatique de la parenté. Un état de l'art relatif à ce domaine de recherche ainsi que les différentes bases de données utilisées sont exposés à la fin de ce chapitre.

1.2 Généralités sur la biométrie

La biométrie est la science qui porte sur l'analyse des caractéristiques physiques ou comportementales propres à chaque individu et permettant l'identification d'une personne ou l'authentification de son identité. Au sens littéral et de manière plus simplifiée, la biométrie signifie la « mesure du corps humain ». On distingue deux catégories de mesures biométriques : les mesures physiologiques telles que les empreintes digitales, le visage, la rétine, l'iris et la voix, et les mesures comportementales telles que la démarche, la dynamique de la signature et le geste de main [10] (voir **Figure 1.1**).

En terme de sécurité, les technologies biométriques sont considérées parmi les technologies les plus puissantes, en effet, et à la différence des mots de passe ou des PINs (numéros d'identification personnelle) qui sont facilement oubliés ou exposés à l'utilisation

frauduleuse, ou des clés ou des cartes magnétiques qui doivent être portées par l'individu et sont faciles à être volées, copiées ou perdues, les caractéristiques biométriques sont uniques à l'individu et il y a peu de possibilité que d'autres individus peuvent avoir les mêmes caractéristiques [11].

Selon la nature du système biométrique, l'entrée de ce dernier peut être composée d'une seule caractéristique biométrique ou une combinaison de caractéristiques physiologiques et comportementales.

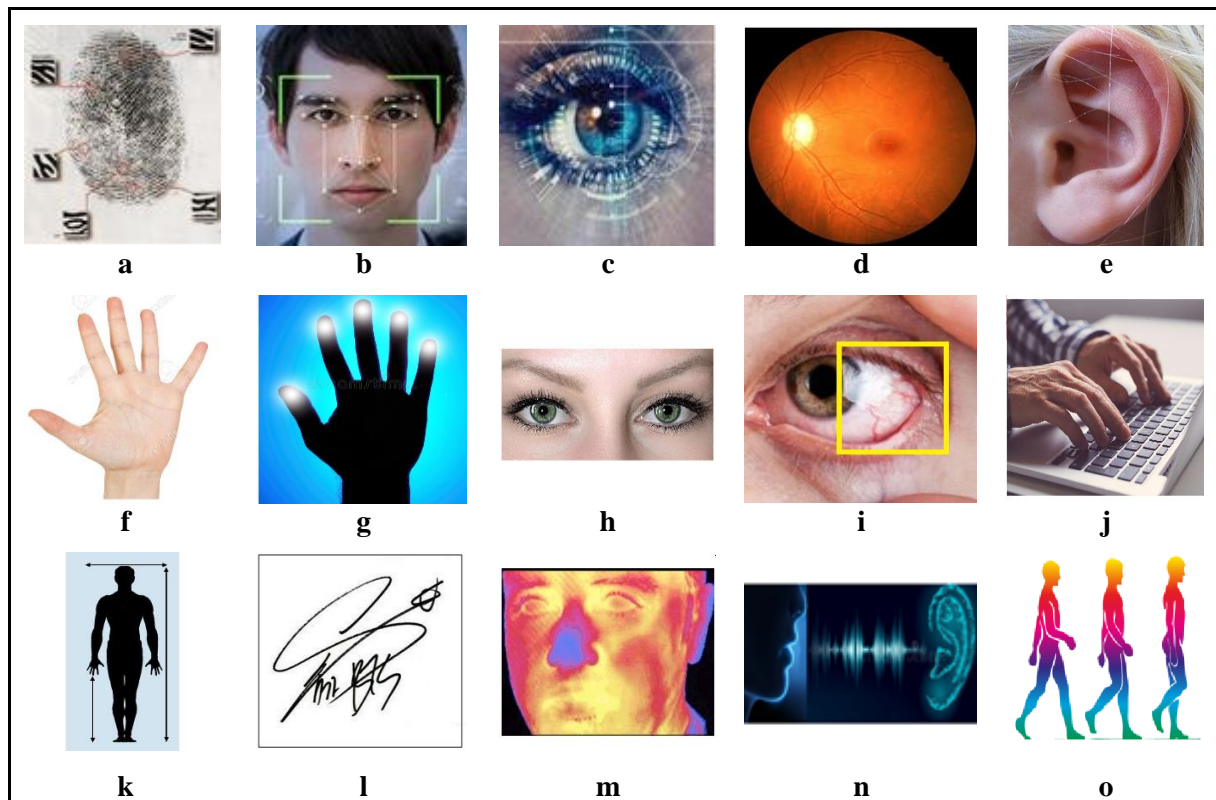


Figure 1.1: Les principales caractéristiques biométriques: **a)** Empreinte digitale, **b)** Visage, **c)** Iris, **d)** Rétine, **e)** Forme de l'oreille, **f)** Paume, **g)** Géométrie de la Main, **h)** Paire oculaire, **i)** Conjonctivale Vasculaire, **j)** Dynamique de frappe, **k)** Anthropométrie, **l)** Signature, **m)** Thermo-gramme du Visage, **n)** la parole, **o)** Démarche.

Selon le contexte de l'application, un système biométrique peut être un système de vérification ou un système d'identification [12]:

1. **Vérification** : le système authentifie l'identité d'une personne en comparant les caractéristiques biométriques capturée avec la Template de l'identité prétendue qui est stockée dans le système. Dans un tel système, l'utilisateur devra entrer son identité et sa caractéristique biométrique, ce qui conduira à une comparaison de la donnée capturée

avec la donnée stockée. Le résultat sera un rejet « *la personne n'est pas qui elle prétend être* » ou une acceptation « *la personne est bien qui elle prétend être* ».

2. **Identification** : le système identifie un individu en recherchant les signatures (Template) de tous les utilisateurs dans la base de données. Par conséquent, le système conduit plusieurs comparaisons pour établir l'identité d'un individu (ou échoue si les informations ne sont pas inscrites dans la base de données de système) sans avoir à soumettre et réclamer une identité.

1.3 Systèmes d'analyse des images faciales

Les systèmes d'analyse des images faciales sont parmi les systèmes biométriques les plus utilisés. L'analyse des images faciales est un axe de recherche très répandu permettant d'atteindre un objectif bien déterminé (Identifier ou authentifier une personne, détecter l'existence de la parenté, etc.), et ce, en faisant des traitements appropriés à partir des images faciales. L'intérêt de cet axe de recherche réside dans le fait que le visage humain présente des caractéristiques universelles, telles que le genre, l'âge et les émotions, par conséquent, les images faciales peuvent être facilement exploitées dans de nombreuses applications [13]. C'est un domaine de la vision par ordinateur basé sur l'utilisation des techniques de l'intelligence artificielle et qui a été largement étudié, par conséquent, plusieurs méthodes d'analyse des images faciales ont été proposées, suivant deux grands axes : l'analyse faciale à partir des images fixes et l'analyse faciale à partir de séquence d'images (vidéo).

L'analyse des images faciales est une technologie qui ne cesse de prendre de plus en plus de l'ampleur. Elle reste la plus acceptable puisqu'elle correspond à ce que les humains utilisent dans l'interaction visuelle, et par rapport aux autres méthodes biométriques, elle s'avère plus avantageuse, d'une part, c'est une méthode non intrusive, c'est-à-dire elle n'exige pas la coopération du sujet (en observant les individus à distance), et d'une autre part, les capteurs utilisés sont peu coûteux [14].

Les technologies de l'analyse des images faciales servent principalement les systèmes de contrôle biométrique dédiés à la sécurité. Elles sont plus rapides, moins invasives (peu ou pas d'interaction avec le sujet) et traitent des données plus fiables que les autres systèmes d'analyse morphologique (empreintes digitales, reconnaissance de l'iris, etc.) ou comportementale (analyse des déplacements, reconnaissance vocale, etc.) [15].

L'analyse des images faciales s'appuie sur la détection faciale. En effet, la détection faciale consiste simplement à détecter des visages dans une image. En d'autres termes, elle a

pour but de donner une région d'intérêt appelé "bounding box" contenant le visage. Cette fonctionnalité est largement utilisée dans de nombreuses applications, particulièrement celles orientées vers la vérification de la parenté à partir des images faciales (Parents, Enfants).

S'investir dans le domaine de l'analyse des images faciales est sans doute motivé par la multiplicité et la variété des champs d'application en vidéo-surveillance, biométrie, robotique, indexation d'images et de vidéos, recherche d'images par le contenu, etc. Ces systèmes sont généralement utilisés à des fins de sécurité. A cet effet, plusieurs recherches très intéressantes portant sur le traitement et l'analyse des images faciales ont été développées, à titre d'exemples on peut citer : l'estimation d'âge [16], la sélection de genre [17], l'anti spoofing [18], la reconnaissance des expressions faciales [19], la reconnaissance du visage [20] et la vérification de la parenté.

1.4 Vérification de la parenté

1.4.1 Généralités et définitions

Motivés par les découvertes psychologiques, plusieurs scientifiques ont étudié la capacité des évaluateurs humains à la perception et à la reconnaissance de la parenté à partir des images faciales humaines, Ils ont classé les relations de parenté différemment. Les conclusions de la littérature incluent les définitions suivantes :

1. La parenté (similitude, familiarité) est un mot utilisé pour reconnaître les relations entre les membres d'une famille [21].
2. La parenté est le degré de parenté génétique entre deux membres de la famille [22].
3. La parenté est l'état de parenté par la naissance, l'ascendance commune, le mariage ou l'adoption. La parenté, c'est estimer les relations des enfants et de leurs parents, à partir des signes et de leur ressemblance [23].
4. La parenté est un système qui établit des relations entre les individus et les groupes sur le modèle des relations biologiques entre les parents et les enfants, entre les frères et sœurs et entre les partenaires conjugaux [24].

De plus, la relation biologique et les similitudes entre les traits de la même famille nous poussent à exploiter cette caractéristique pour dégager deux types de systèmes destinés soit à « Reconnaître la parenté » soit à « Vérifier la parenté », et ce, à partir des images faciales. Ces deux types de systèmes sont définis comme suit [25] :

1. Les systèmes de reconnaissance de parenté entre les personnes dont le but est l'étude de la similitude entre les visages humains et par conséquent identifier les personnes qui ont une relation de parenté
2. Les systèmes de vérification de parenté commencent par une paire d'images faciales de (Parents, Enfants), ensuite détermine s'il existe ou non une parenté entre une paire de visages. Ce système se base sur l'apprentissage de reconnaître le parent génétique et le non-parent sur la base de caractéristiques extraites d'images faciales.

En vision par ordinateur, la vérification de la parenté reste un sujet très difficile car le visage véhicule différentes caractéristiques faciales. Le problème de vérification de la parenté rencontre également de nombreuses difficultés, telles que les variations d'âge, de sexe, d'appartenance ethnique, les ressemblances entre personnes sans relation réelle et les différences d'attributs faciaux entre personnes d'une même famille [26].

1.4.2 Structure d'un système de vérification de la parenté

Le système de la vérification de parenté consiste à utiliser l'image du visage de deux personnes pour déterminer s'il existe une relation familiale entre elles.

En général la structure du système de parenté comprend cinq étapes : (1) l'acquisition d'image, (2) la détection ou le prétraitement des visages, (3) l'extraction de caractéristiques, (4) la représentation de caractéristiques, (5) la classification ou la vérification de parenté.

- 1. Acquisition d'image :** C'est l'opération qui permet d'extraire du monde réel une représentation bidimensionnelle pour des objets en 3D, cette opération peut être statique (Appareil photo, Scanner, etc.), à ce niveau on aura une image, ou dynamique (Caméra, Web Cam), dans ce cas on aura une séquence vidéo.
- 2. Prétraitement du visage :** C'est une étape consacrée pour détecter et localiser les visages humains à partir d'une image d'entrée.
- 3. Extraction des caractéristiques :** L'étape d'extraction des caractéristiques est une étape très importante, elle est destinée à extraire les caractéristiques des images du visage (parents, enfants) utilisant des différents descripteurs pour convertir les images en vecteurs représentatifs et trouver la relation entre eux (parents, enfants).
- 4. Représentation des caractéristiques :** Dans cette étape, les deux vecteurs de caractéristiques (enfant, parent) doivent être fusionnés en un seul vecteur de

caractéristiques à l'aide des équations appliquées entre les deux vecteurs (bit par bit). Pour attaquer directement le classifieur (l'étape suivante).

5. **Classification** : La dernière étape du système est l'étape de classification où la décision de vérification de la parenté est effectuée, à l'aide des classifieurs utilisant les vecteurs représentatifs passés par l'étape d'apprentissage puis le test pour donner la décision finale.

Un exemple d'une structure générale d'un système de vérification de la parenté à partir des images faciales est donné par la **Figure 1.2**.

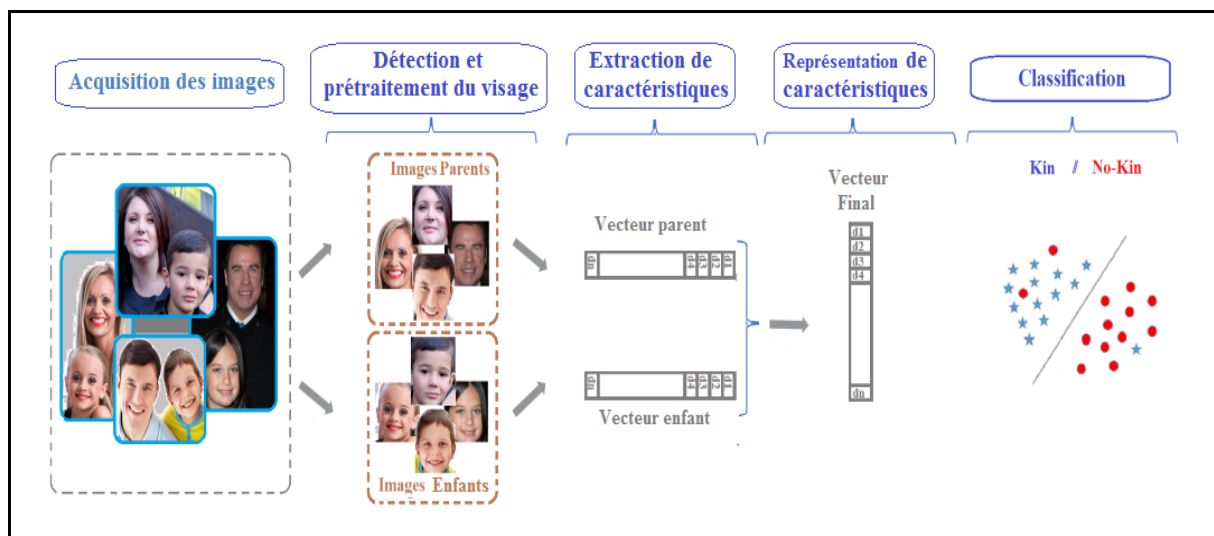


Figure 1.2: Structure générale d'un système de la vérification de parenté par image faciale.

1.4.3 Applications

La vérification de la parenté à l'aide d'images faciales est un problème relativement nouveau et difficile dans la recherche de la biométrie. Récemment, il a attiré une attention considérable de la société de biométrie, principalement motivée par les caractéristiques relatives selon lesquelles les enfants ressemblent généralement plus à leurs parents qu'à d'autres personnes en ce qui concerne l'apparence du visage. Les chercheurs ont commencé à étudier le problème de la vérification de la parenté, où les objectifs sont d'estimer la parenté entre des individus étroitement liés en fonction des caractéristiques du visage [27].

La vérification de la parenté a un impact significatif sur les applications réelles, et ce, dans de nombreux domaines. Nous récapitulons les domaines possibles d'application comme suit :

1. L'organisation d'albums familiaux à partir des images faciales des mêmes membres de famille.

2. Création automatiquement de l'arbre généalogique à partir des collections d'images lorsque les relations entre les personnes sont reconnues dans les réseaux sociaux.
3. Rechercher des personnes dans des bases de données publiques à partir d'un membre de la même famille.
4. Enrichir les bases de données par des nouvelles familles ajoutés ou détectés en ligne pour renforcer l'efficacité du système de la vérification de parenté.
5. La recherche des personnes perdues (Enfants, les personnes âgées, les victimes d'accidents, etc.) à partir d'un membre de la même famille.
6. L'identification des personnes recherchées (fugitifs, criminels, etc.) à partir d'un membre de la même famille.
7. La détermination de la parenté ne se limite plus exclusivement à l'analyse génétique et biologique, mais elle s'étend désormais au domaine de la biométrie. L'analyse biologique ne peut être utilisée dans de nombreuses situations car le processus prend des jours pour obtenir des résultats avec des coûts élevés, par contre la vérification biométrique de la parenté se fait d'une façon rapide et économique.

Cependant, la recherche sur ce sujet reste complexe et difficile. Pour atteindre l'objectif ciblé, l'analyse de la parenté à travers les visages humains doit être menée de façon exhaustive. Les motivations et les applications susmentionnées attirent un nombre important de chercheurs pour présenter leurs contributions dans ce domaine. Évidemment, la recherche dans ce domaine est toujours active.

1.5 Etat de l'art de la vérification de la parenté

De nombreux chercheurs en vision par ordinateur ont étudié le problème de la vérification de la parenté et de nombreuses approches ont été proposées et classées dans deux catégories : approches basées sur les descripteurs et des approches basé sur l'apprentissage métrique. Dans notre recherche, nous avons classé ces approches en trois catégories à base : d'image-texture, d'apprentissage multiple et d'apprentissage en profond.

1.5.1 Approches basées sur les images-texture

Les premières méthodes de vérification de la parenté extraient généralement les caractéristiques fabriquées à la main à l'aide de descripteurs de texture à partir d'images faciales, puis entraînent ces caractéristiques à l'aide d'un classificateur.

Fang et al. [28] ont proposé un système de vérification de la parenté basé sur un modèle de structure illustrée (PSM). En outre, ils introduisent la base de données Cornell.

Premièrement, ils extraient 22 types de traits faciaux composés de couleurs, de parties faciales, de distances faciales et d'histogrammes de traits dégradés. Enfin, ils calculent les différences entre les vecteurs caractéristiques des parents et des enfants correspondants et appliquent les classificateurs KNN et SVM.

Zhou et al. [29] ont introduit une nouvelle base de données sur les particuliers et ses images sont dans des conditions non contrôlées. Ils ont proposé un descripteur d'entité basé sur l'apprentissage par une pyramide spatiale qui utilisait à la fois des informations locales et globales, et ils utilisaient SVM pour la phase de classification. Les résultats obtenus étaient prometteurs.

Un autre travail intéressant a été proposé par Shao et al. [30], dans lequel ils ont utilisé la version 2 de la base de données UB KinFace pour vérifier la parenté sur la base de filtres de Gabor locaux robustes permettant d'extraire des caractéristiques invariantes génétiques. En d'autres termes, une métrique et un apprentissage de sous-espace de transfert ont été adoptés pour réduire l'écart entre les enfants et leurs parents âgés.

Kohli et al [31] ont proposé de coder la similarité de parenté à l'aide d'un descripteur d'auto-similarité (SSD) et de vérifier formellement la vérification de parenté en tant que problème de classification. Ils ont appliqué leur méthode sur la base de données de parenté de l'IIITD annotée en ce qui concerne la relation de parenté particulière, l'ethnie et le sexe.

A. Moujahid et al [32] ont proposé un système basé sur la covariance (statistiques de second ordre) entre diverses entités locales utilisant la représentation PML pour modéliser les images de visage, ils ont appliqué leur méthode sur les bases de données : UB Kinface, KinfaceW-I et KinFaceW-II.

De notre part, nous avons proposé une approche basée sur les descripteurs (LPQ et LDP) utilisant (ML) pour la représentation faciale [33], et dans un autre travail nous avons proposé une approche basée sur les descripteurs (BSIF et LBP) utilisant (PML) pour la représentation faciale [34], en plus nous avons testé les descripteur LTP [35] avec le (MB) pour la représentation faciale, et nous avons appliqué nos méthodes sur les bases de données (Cornell, UB KinFac , KinFace W-I et KinFace W-II)

1.5.2 Approches basées sur l'apprentissage multiple

Les approches basées sur l'apprentissage multiple ont été motivées par l'idée de modéliser des données de haute dimension en utilisant un sous-collecteur approximatif de basse

dimension de l'espace d'origine. Ils ont de bonnes performances, cependant, ils ont des coûts de calcul élevés.

Xia et al. [36] ont utilisé une autre base de données appelée UB KinFace, qui contient des images de visages d'enfants, de jeunes parents et de vieux parents et a utilisé une méthode d'apprentissage de sous-espace de transfert étendu pour atténuer l'énorme divergence des distributions entre les enfants et les vieux parents. Une distribution intermédiaire a été utilisée pour fermer le pont et réduire la divergence entre les distributions sources.

Lu et al. [37] ont proposé une méthode d'apprentissage métrique repoussé par quartier (NRML) pour la vérification de la parenté. De plus, ils ont proposé une méthode NRML multivue (MNRML) pour rechercher une distance métrique commune afin de mieux utiliser les caractéristiques de descripteurs multiples et ils ont appliqué leur méthode aux jeux de données KinFaceW-I et KinFaceW-II. Hu et al. [38] ont proposé une méthode d'apprentissage multimétrique à grande marge et ont appliqué leur méthode aux jeux de données KinFaceW-I et KinFaceW-II.

Yan et al. [39] ont proposé une méthode d'apprentissage multimétrique discriminant (DMML) pour la vérification de la parenté. Premièrement, ils ont extrait plusieurs caractéristiques en utilisant différents descripteurs de visage ; ensuite, ils ont appris conjointement plusieurs métriques de distance pour ces caractéristiques de descripteur sous lesquelles la probabilité d'une paire d'images de visage où la relation de parenté avait une distance plus petite que la paire qui n'avait pas de relation de parenté. Dans leur travail, ils ont appliqué leur méthode à deux bases de données : les bases de données Cornell et UB KinFace.

Zhou et al. [40] ont proposé l'apprentissage de la similitude d'ensemble (ESL). Premièrement, ils ont introduit une fonction de similarité bilinéaire clairsemée pour modéliser la relative des propriétés codées dans les données de parenté. La fonction de similarité paramétrée par une matrice diagonale jouit d'une supériorité en termes d'efficacité de calcul, ce qui la rend plus pratique pour les applications de vérification de parenté de grande dimension du monde réel.

Liang et al. [41] ont développé un cadre d'apprentissage métrique basé sur l'intégration de graphe pondéré (WGEML) pour la vérification de la parenté faciale. Ils extraient quatre types d'entités : LBP, HOG, transformation d'entité invariante à l'échelle et face de groupe de géométrie visuelle (VGGFACE). Ils ont ensuite construit un graphique intrinsèque et deux graphiques de pénalités pour caractériser la compacité intra-classe et la séparabilité interclasse

pour chaque représentation d'entité. Ils ont mené des expériences approfondies sur les bases de données KinFaceW-I, KinFaceW-II et TSKinFace.

Yan et al. [42] ont proposé une méthode d'apprentissage des caractéristiques discriminantes (PDFL) basée sur un prototype pour la vérification de la parenté. Cette méthode vise à apprendre des caractéristiques de niveau intermédiaire discriminantes où ils ont construit un ensemble d'échantillons de visage avec des relations de parenté sans étiquette à partir d'un ensemble de données sauvage qui est considéré comme l'ensemble de référence. Ensuite, chaque échantillon de l'ensemble de données de parenté de visage d'apprentissage est représenté comme un vecteur d'entité de niveau intermédiaire, où chaque entrée correspond à la valeur de décision de SVM. Ils ont appliqué leur méthode aux bases de données Cornell et UB KinFace.

De notre part, nous avons proposé une approche basée sur WLD comme descripteur et nous avons étudié l'influence de l'Analyse Discriminante Sensible aux Localités (LSDA) pour réduire le nombre de caractéristiques et de l'Analyse Discriminante Linéaire (LDA) pour la classification de parenté, et nous avons appliqué notre approche sur trois bases de données (Cornell, UB KinFace, Family 101) [43].

1.5.3 Approches basées sur l'apprentissage en profond

Les approches d'apprentissage en profond utilisent principalement des réseaux de neurones convolutifs (CNN), qui sont un type de réseaux de neurones artificiels à action directe dans lesquels le modèle de connectivité entre ses neurones est inspiré par l'organisation du cortex visuel animal. Ses points forts incluent l'obtention de résultats stables et de bonne qualité, l'adaptation aux applications en temps réel et l'immunité contre les poses faciales ; l'inconvénient est le coût de calcul élevé.

Zhang et al. [44] ont proposé d'extraire des caractéristiques de haut niveau basées sur un CNN profond. Ces caractéristiques sont produites à partir des activations des neurones de la dernière couche cachée, puis introduites dans un classificateur softmax pour vérifier la parenté de deux personnes. Ils ont appliqué leur méthode sur les bases de données KinFaceW-I et KinFaceW-II.

Kohli et al. [45] ont proposé une vérification de parenté hiérarchique via un cadre d'apprentissage de la représentation pour apprendre la représentation des différentes régions du visage de manière non supervisée. Ils ont proposé une approche pour la représentation des caractéristiques appelée réseaux de croyances profondes contractives filtrées et ont appliqué

leur méthode sur cinq bases de données : Cornell, UB KinFace, KinFaceW-I, KinFaceW-II et WVU Kinship.

Dehghan et al. [46] ont proposé un algorithme utilisant l'apprentissage en profond qui fusionne les caractéristiques et les métriques découvertes via des auto-encodeurs à portes avec une couche de réseau neuronal discriminante. Ils ont ensuite analysé la corrélation entre ces caractéristiques détectées automatiquement et celles trouvées dans les études anthropologiques. Ils ont appliqué leur méthode sur les bases de données KinFaceW-I et KinFaceW-II.

Wang et al. [47] ont proposé un modèle de vérification de la parenté profonde. Ils ont intégré une architecture d'apprentissage en profond dans l'apprentissage métrique pour sélectionner des caractéristiques non linéaires, qui peuvent trouver l'espace de projet approprié pour s'assurer que la marge des paires négatives est aussi grande que possible et la marge des paires positives est aussi petit que possible. Ils ont appliqué leur méthode aux bases de données KinFaceW-I et KinFaceW-II. Wang et al. ont proposé un jeune modèle de génération croisée pour la vérification de la parenté. Ils ont utilisé une architecture profonde avec une nouvelle perte métrique discriminante clairsemée pour l'extraction des fonctionnalités et ont appliqué leur méthode sur les bases de données (familles à l'état sauvage).

De notre part, nous avons proposé une approche basée sur les caractéristiques profondes du descripteur VGG-FACE en utilisant le score de Fisher pour la sélection de caractéristiques et SVM pour la classification [48], et dans un autre travail aussi basé sur un modèle de caractéristiques profondes du descripteur ResNet utilisant T-Test pour la sélection de caractéristiques et SVM pour la classification [49], les deux approches sont appliquées sur cinq bases de données (Cornell, UB KinFace, Family 101, Kinface W-I et KinFace W-II).

1.6 Principales bases de données

Les bases de données utilisées dans le domaine de la vérification de la parenté se divisent en deux grandes classes, à savoir, les bases de données à base des images faciales et les bases de données à base des images vidéo.

1.6.1 Les bases de données à base des images faciales

Les bases de données de la vérification la parenté ont été établies depuis 2010. Au début, cela dépendait de l'image des célébrités capturé spontanément ensuite par la recherche d'un une photo du père ou du fils correspondant.

a. Cornell Kinship (2010)

Cette base de données d'images faciales des paires parents-enfants est recueillie grâce à une recherche en ligne pour des images de personnalités publiques et de célébrités et de leurs parents ou enfants. Il y a 150 paires avec des variations dans l'âge, le sexe, la raceetc., pour couvrir la large distribution de l'aperçu du visage de la base de données d'images faciales recueillies. La plupart des images ont été trouvées sur Google Images, Pour s'assurer que les traits faciaux extraits sont de haute qualité, les images de visage sont choisies pour être frontales et une expression faciale neutre. Cette base de données comprend environ 50% de Caucasiens, 40% d'Asiatiques, 7% d'Afro-Américains et 3% d'autres ; 40% des 150 images sont des paires père-fils, 22% sont père-fille, 13% sont mères-fils et 26% sont mère-fille. Par conséquent, il présente une distribution étendue des caractéristiques faciales qui dépendent de la race, du sexe, de l'âge, de la carrière, etc. [28]. **La Figure 1.3** présente des exemples (positive et négative) de paire des images pour différentes relations de parenté pour la base de données « Cornell Kinship ».

| | Père - Fils | Père - Fille | Mère - Fils | Mère - Fille |
|---------|---|---|--|---|
| Positif |  |  |  |  |
| Négatif |  |  |  |  |

Figure 1.3: Exemples d'images de la base de données « Cornell Kinship »

b. UB KinFace (2011)

La version 1 UB KinFace : c'est la première base de données contenant des images d'enfants et de leurs parents à différents âges. Toutes les images de la base de données sont des collections réelles de personnalités publiques (célébrités et politiciens) provenant d'Internet, ces images de visage sans contrainte montrent une large gamme de variations, y compris la pose, l'éclairage, l'expression, l'âge, le sexe, le fond, la race, la saturation des couleurs, etc. Cette base de données comprend 600 images de 400 personnes qui peuvent être séparées en 200 groupes. Chaque groupe est composé d'images d'enfants, de jeunes parents et de parents âgés [30].

La version 2 UB KinFace : principalement divisé en deux parties en termes de race, c'est-à-dire, asiatique et non-asiatique, chacune composée de 100 groupes, 200 personnes et 300 images. Typiquement, il y a quatre relations de parenté, c'est-à-dire «fils-père», «fils-mère», «fille-père» et «fille-mère», illustrent les statistiques du point de vue de la race et de parenté. les célébrités masculines asiatiques et non asiatiques, que ce soit le fils ou le père, dominent dans la base de données UB KinFace. Lorsque quatre relations parentales possibles sont considérées dans notre problème, les «pères» deviennent raisonnablement les rôles essentiels, avec 46,5% et 38,5% sur tous les groupes dans les relations «fils-père» et «fille-père». Ce phénomène peut s'expliquer par le fait qu'il y a statistiquement plus d'hommes que de femmes dans les communautés gouvernementales, de divertissement et sportives actuelles [30].

La **Figure 1.4** présente des exemples (positive et négative) de paire des images pour différentes relations de parenté pour la base de données « UB KinFace ».









| | Père - Fils | Père - Fille | Mère - Fils | Mère - Fille |
|---------|---|---|--|---|
| Positif |  |  |  |  |
| Négatif |  |  |  |  |

Figure 1.4: Exemples d'images de la base de données « UB KinFace »

c. IITD Kinship (2012) :

IITD Kinship : comprenant 544 images de 272 paires dans un environnement non contraint. La base de données est constituée d'images de célébrités téléchargées sur Internet. Les images ont été annotées par rapport à la relation de parenté et à l'ethnicité des individus. La base de données est également augmentée de 272 paires sans parenté. Les images ont été classées en quatre ethnies différentes : afro-américaine, américaine, indienne et asiatique sauf indienne. La relation de parenté a été catégorisée dans les sept relations suivantes : Frère - Frère, Frère - Soeur, Père - Fille, Père - Fils, Mère -Fille, mère - fils, et soeur – soeur.

Exemples d'images de la base de données de parenté de l'IITD illustrant différentes relations de parenté: a) mère-fille, b) père-fils, c) père-fille, d) frère-sœur, e) frère-frère, f) mère-fils, g) sœur-sœur [31].

La Figure 1.5 présente des exemples de paire des images pour différentes relations de parenté pour la base de données « IITD Kinship ».


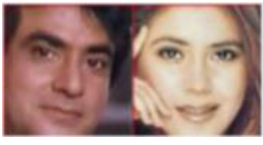





| Père-Fils | Père-Fille | Mère-Fils | Mère-Fille | | |
|---|---|--|---|---|--|
|  |  |  |  | | |
| Frère -Frère | | Frère -Sœur | | Sœur -Sœur | |
|  | |  | |  | |

Figure 1.5: Exemples d'images de la base de données « IITD Kinship »

d. Family101 (2013)

"Famille 101" contenant 101 arbres généalogiques différents, dont 206 familles, 607 individus, avec 14 816 images. L'ensemble de données comprend des familles publiques renommées. L'ensemble de données final comprend environ 72% de Caucasiens, 23% d'Asiatiques et 5% d'Afro-Américains pour garantir une distribution étendue des caractéristiques faciales qui dépendent de la race, du sexe et de l'âge, il y a 213 relations père-fils, 147 relations père-fille, 184 relations mère-fils et 148 relations mère-fille. Pour certains membres de la famille dans l'ensemble de données Family 101 la recherche sur Internet renvoie relativement peu de photographies uniques, nous utilisons seulement 546 membres de la famille pour lesquels il y a au moins 5 photographies uniques. La Figure 1.6 présente des exemples (positive et négative) de paire des images pour différentes relations de parenté pour la base de données « Family101 » [50].

| | Père - Fils | Père - Fille | Mère - Fils | Mère - Fille |
|---------|---|---|--|---|
| Positif |  |  |  |  |
| Négatif |  |  |  |  |

Figure 1.6: Exemples d'images de la base de données « Family101 »

e. KinFaceW-I et KinFaceW-II (2014)

Il existe quatre relations de parenté dans les ensembles de données KinFaceW-I et KinFaceW-II : père-fils (F-S), père-fille (F-D), mère-fils (M-S) et mère-fille (M-D). Pour KinFaceW-I, il existe 156, 134, 116 et 127 paires d'images de parenté pour ces quatre relations. Pour l'ensemble de données KinFaceW-II, chaque relation contient 250 paires d'images de parenté, La différence entre KinFaceW-I et KinFaceW-II est que chaque paire d'images faciales de parenté dans KinFaceW-I a été acquise à partir de différentes photos et que dans KinFaceW-II a été obtenue à partir de la même photo.

Plusieurs exemples d'image de base de données KinFaceW-I./W-II de haut en bas sont les relations de parenté père-fils (FS), père-fille (FD), mère-fils (MS) et mère-fille (MD), et les deux images voisines dans chaque rangée sont avec relation de parenté, respectivement [37].

La Figure 1.7 présente des exemples (positive et négative) de pair des images pour différentes relations de parenté pour la base de données « KinFaceW-I ».

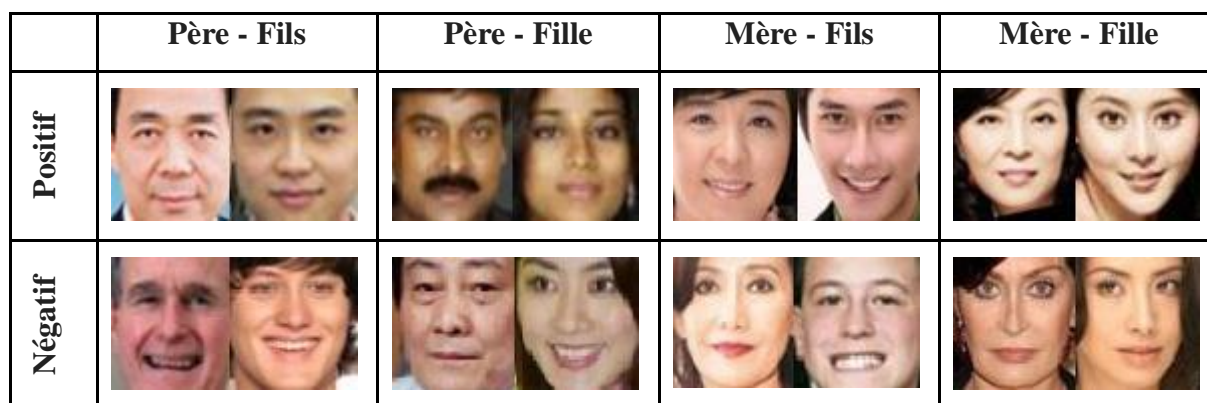


Figure 1.7: Exemples d'images de la base de données « KinFaceW-I »

La Figure 1.8 présente des exemples (positive et négative) de pair des images pour différentes relations de parenté pour la base de données « KinFaceW-II».

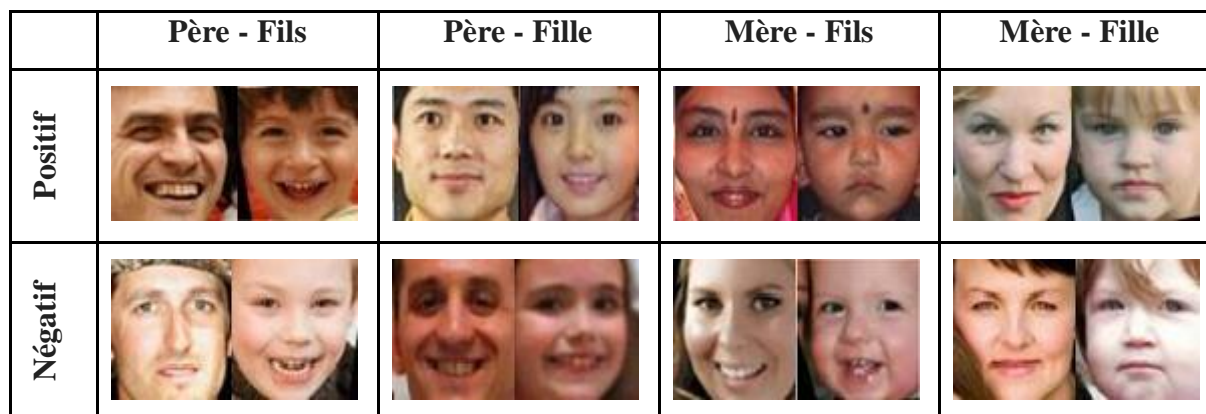


Figure 1.8: Exemples d'images de la base de données « KinFaceW-II »

f. TS KinFace (2015)

(Base de données de face de parenté à trois sujets) : Toutes les images de la base de données sont récoltées sur Internet en se basant sur la connaissance de la famille des figures publiques et du réseau social de partage de photos tel que flickr.com. Lors de la collecte des images, aucune restriction en termes de pose, d'éclairage, d'expression, d'arrière-plan, de race, de qualité d'image, etc. Cette base de données est caractérisée par le plus grand nombre de personnes et de familles, dans la base de données TSKinFace il y a deux types de relations de parenté: Père-Mère-Fils (FMS) et Père-Mère-Fille (FM-D). Le FM-S et le FM-D contiennent respectivement 513 et 502 groupes de relations de parenté tri-sujet. Nous avons donc 1015 groupes dans cette base de données, Les familles incluses dans notre base de données sont également diverses en termes de races. Pour la relation FM-S, il y a respectivement 343 et 170 groupes de relations de parenté tri-sujet pour les Asiatiques et les Non-Asiatiques, les chiffres pour les groupes asiatiques et non asiatiques sont respectivement de 331 et 171. **La Figure 1.9** présente des exemples (positive et négative) de pair des images pour différentes relations de parenté pour la base de données « TS KinFace» [51].

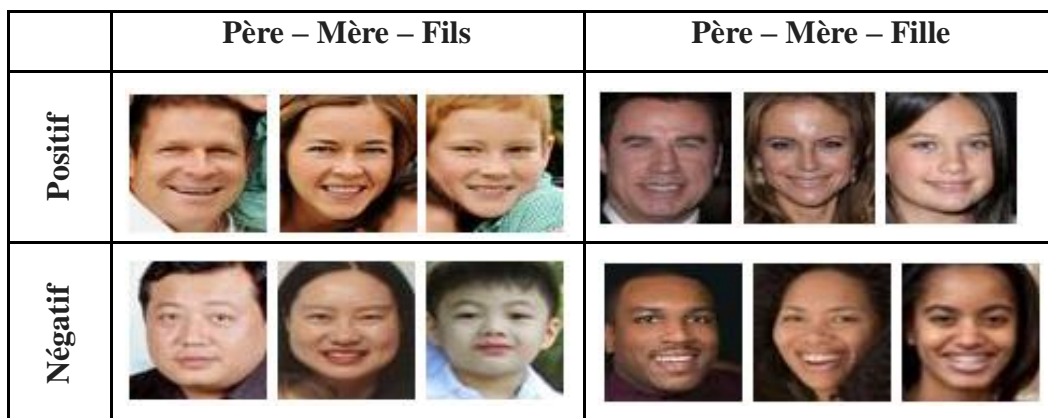


Figure 1.9: Exemples d'images de la base de données « TS KinFace »

g. WVU Kinship (2016)

La base de données WVU Kinship (**West Virginia University**) se compose de plusieurs images de parents. Les images utilisées dans cette base de données sont collectées à partir de trois bases de données : Vadana, Kinship Verification database et UB Kin database. La base de données comprend 150 paires de parents et 150 paires de non-parents avec 39 combinaisons soeur-soeur (SS), 36 combinaisons frère-soeur (BS), 35 combinaisons frère-frère (BB), 50 combinaisons père-fils (FS), 40 père-fille (FD), 41 combinaisons mère-fille (MD) et 59 combinaisons mère-fils (MS) [45].

La **Figure 1.10** présente des exemples de paire des images pour différentes relations de parenté pour la base de données « WVU Kinship ».

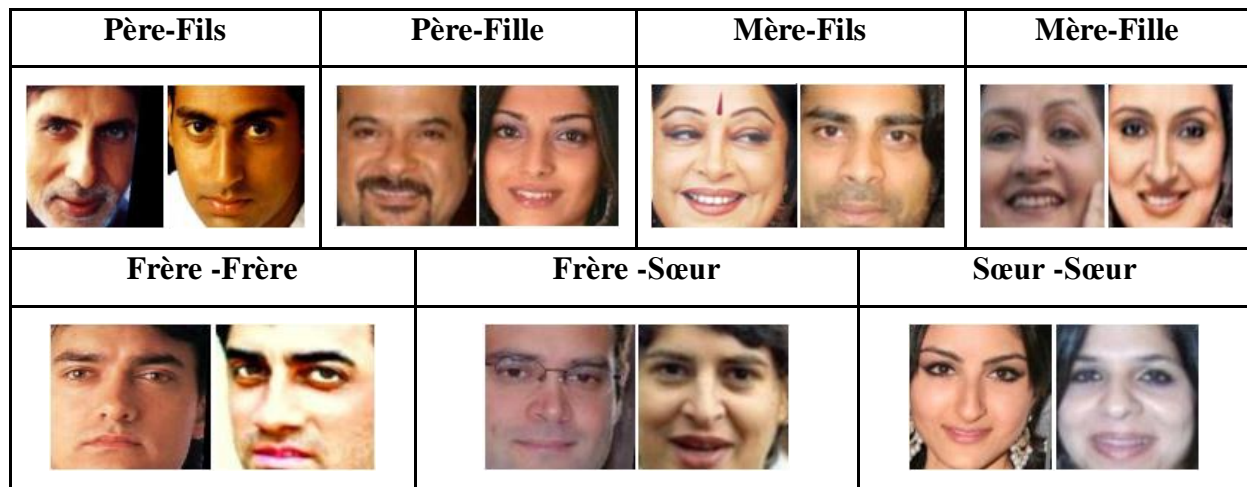


Figure 1.10: Exemples d'images de la base de données « WVU Kinship »

1.6.2 Les bases de données à base des vidéos

a. UvA-NEMO Smile

Est une base de données de sourire à grande échelle qui contient 1240 vidéos de sourire (597 spontanées et 643 posées) de 400 personnes.

L'âge des personnes varie de 8 à 76 ans. Les vidéos sont en couleur RVB et enregistrées avec une résolution de (1920 * 1080) pixels à un taux de 50 images par seconde dans des conditions d'éclairage contrôlées. Pour plus d'éclairage et de normalisation des couleurs, un nuancier est présent à l'arrière-plan des vidéos. De nombreuses familles ont participé à la collecte de la base de données, permettant son utilisation pour l'évaluation de la parenté automatique à partir de vidéos. Au total, 95 paires de vidéos de visages ont été identifiées entre 152 personnes dans la base de données. Il existe sept relations de parenté différentes entre les paires de vidéos: sœur-sœur (SS), frère-frère (FF), sœur-frère (SF), mère-fille (MFe), mère-fils (MFs), père-fille (PFe) et Père-Fils (PFs). L'association des vidéos de personnes ayant des relations de parenté donne 228 paires de vidéos spontanées et 287 paires de vidéos de sourire posées [52].

La **Figure 1.11** présente des exemples de pair des images extraits a partir des séquences de vidéo pour différentes relations de parenté pour la base de données « UvA-NEMO Smile»

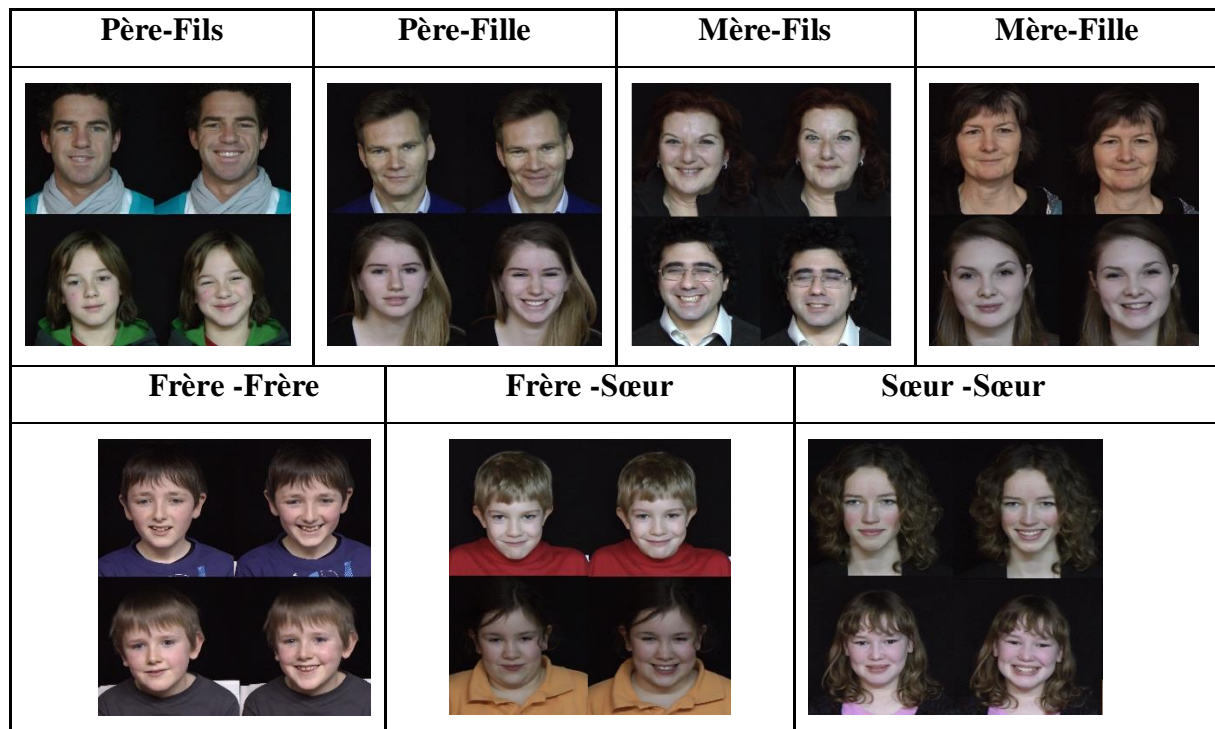


Figure 1.11: Exemples d'images de la base de données « UvA-NEMO Smile »

b. Kinship Face Videos in the Wild (KFVW)

L'ensemble de données KFVW comprenait 418 paires de vidéos de visages collectées à partir d'émissions télévisées sur le Web dans des conditions non contrôlées telles que l'éclairage, la pose, l'occlusion, l'âge, l'expression, le maquillage, l'arrière-plan, etc. Chaque vidéo contient environ (100 – 500) images. La taille moyenne d'une image vidéo est d'environ (900 * 500) pixels. Il existe quatre types de relations de parenté dans l'ensemble de données KFVW: père-fils (P-Fs), père-fille (P-Fe), mère-fils (MFs) et mère-fille (MFe), et il y a 107, 101, 100 et 110 paires de vidéos de visages de parenté pour les relations de parenté (P-Fs), (P-Fe), (MFs) et (MFe) respectivement [53]. La Figure 1.12 présente des exemples de paires d'images extraites à partir des séquences de vidéo pour différentes relations de parenté pour la base de donnée «KFVW»

c. KIVI (Kinship Verification)

C'est une base de données de visages présentée dans des vidéos de parenté, contient de 503 individus formant 355 paires de parents positifs de 211 familles présentées dans des variations sauvages dues à la pose, à l'éclairage, l'occlusion, l'ethnicité et l'expression sont introduites. La durée moyenne de la vidéo est de 18,78 secondes avec une fréquence d'images moyenne de 26,79 images par seconde (ips) et le nombre total d'images fixes dans cette base de données est supérieur à 250000, correspondant à sept relations de parenté :

père-fils (P-Fs), père-fille (P-Fe), mère-fils (M-Fs) et mère-fille (M-Fe), Frère-Frère (F-F), Frère-Sœur (F-S), sœur-sœur (S-S) [54].

La Figure 1.13 présente des exemples de pair des images extraits a partir des séquences de vidéo pour différentes relations de parenté pour la base de donnée « KIVI »

| Père - Fils | Père - Fille | Mère - Fils | Mère - Fille |
|---|---|--|---|
|  |  |  |  |

Figure 1.12: Exemples d'images de la base de données « KFVW »


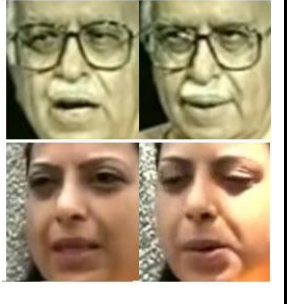
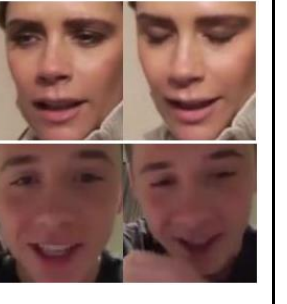


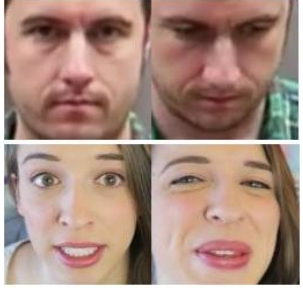

| Père-Fils | Père-Fille | Mère-Fils | Mère-Fille |
|---|---|--|---|
|  |  |  |  |
| Frère-Frère | Frère -Sœur | Sœur - Sœur | |
|  |  |  | |

Figure 1.13: Exemples d'images de la base de données « KIVI »

Le **Tableau 1.1** présente un résumé sur les bases de données de la vérification de parenté

Tableau 1.1: Résumé sur les bases de données de la vérification de parenté

| Année | Base de données | Nombre de paire | Nombre des images | Nombre de relation | |
|-------|-----------------|-----------------|-------------------|--------------------|--------|
| 2010 | Cornell Kinship | 150 | 300 | 4 | Images |
| 2011 | UB KinFace | 200 | 600 | 4 | |
| 2012 | IITD Kinship | 272 | 544 | 7 | |
| 2013 | Family101 | 206 | 14 816 | 4 | |
| 2014 | KinFaceW-I | 533 | 1066 | 4 | |
| | KinFaceW-II | 1000 | 2000 | 4 | |
| 2015 | TS Kinface | 1015 | 787 | 4 | |
| 2016 | WVU Kinship | 226 | 904 | 7 | |
| 2013 | UvA-NEMO Smile | 95 | 513 | 7 | Vidéos |
| 2014 | KFWW | 418 | 418 | 4 | |
| 2018 | KIVI | 503 | 503 | 7 | |

1.7 Conclusion

Dans ce chapitre, nous avons présenté quelques définitions et notions de base relatives aux systèmes biométriques et particulièrement ceux basés sur l'utilisation et l'exploitation des images faciales. Les concepts nécessaires à la compréhension de la vérification de la parenté à partir des images faciales ont été largement exposés ainsi que les différentes parties de ces types de systèmes et leurs principaux domaines d'application. L'état de l'art présentant les principaux travaux dans le domaine de la vérification de la parenté ainsi que les différentes bases de données existantes (images et vidéos) ont été présentés à la fin de ce chapitre.

Chapitre 2

Prétraitement et extraction des caractéristiques d'une image faciale

Sommaire

| | |
|---|-----------|
| 2.1 Introduction | 26 |
| 2.2 Prétraitement du visage | 26 |
| 2.2.1 Détection du visage | 26 |
| 2.2.2 Détection des yeux | 28 |
| 2.2.3 Correction de position..... | 28 |
| 2.3 Extraction des Caractéristiques | 30 |
| 2.3.1 Descripteurs de texture | 30 |
| 2.3.2 Descripteurs d'apprentissage profond | 41 |
| 2.4 Représentation du visage | 45 |
| 2.4.1 Multi Bloc (MB) | 45 |
| 2.4.2 Multi Niveaux (ML)..... | 46 |
| 2.4.3 Pyramide Multi Niveau (PML) | 46 |
| 2.4.3 Combinaison (Descripteurs / Représentation faciale) | 47 |
| 2.5 Conclusion | 47 |

2.1 Introduction

L'extraction des caractéristiques est l'étape la plus importante dans les systèmes de vision par ordinateur et en particulier les systèmes biométriques basés sur la reconnaissance des visages. Elle consiste à exprimer l'entité à reconnaître sous une forme d'un vecteur appelé vecteur caractéristique composé d'un ensemble de paramètres pertinents et non redondants et utiles pour la perception, la mémorisation et la reconnaissance d'un visage. Les performances de ce type de systèmes dépendent en grande partie de l'efficacité de la méthode choisie pour extraire ces caractéristiques [55]. Ce chapitre est consacré à la présentation des différentes méthodes d'extraction de caractéristiques. Nous commençons par décrire les différentes techniques utilisées pour le prétraitement des images faciales, et en particulier les techniques de détection du visage, la détection des yeux et la correction de la position. Ensuite, nous exposons un panorama des méthodes les plus significatives utilisées pour l'extraction de caractéristiques à partir des images faciales tout en basant sur les méthodes basées sur les descripteurs de texture et les descripteurs d'apprentissage en profond. Nous terminons le chapitre par l'étude des trois méthodes de représentation du visage, à savoir : MB, ML, PML.

2.2 Prétraitement du visage

Le Prétraitement du visage est un ensemble d'opérations imposées par la nature des images faciales d'une part, et par les contraintes rencontrées lors de l'acquisition de l'image faciale d'une autre part. Elles sont appliquées sur les images faciales dans le but de les préparer pour les traitements futurs. L'obtention de haute performance pour un système de reconnaissance de visage dans le monde réel avec les scénarios les plus pratiques est fortement liée à la nature des opérations de prétraitement appliquées. La détection du visage, L'extraction des yeux, du nez et de la bouche du visage humain sont parmi les opérations de prétraitement les plus utilisées dans les systèmes de vision par ordinateur utilisant les images faciales. Localiser ces régions anatomiques pertinentes du visage est souvent la première étape dans de la majorité des systèmes d'analyse et de traitement des images faciales [56].

Parmi les opérations de prétraitement des images faciales les plus utilisées on trouve les suivantes :

2.2.1 Détection du visage

La détection du visage peut se faire par détection de la couleur de la peau, la forme de la tête ou par des méthodes détectant les différentes caractéristiques du visage. Cette étape est autant plus délicate quand l'image acquise contient plusieurs objets de visage ou un fond non

uniforme qui crée une texture perturbant la bonne segmentation du visage. Cette étape est dépendante de la qualité des images acquises. Dans la littérature scientifique, le problème de localisation de visages est aussi désigné par la terminologie "détection de visages" [57]. Les performances globales de tout système d'analyse et de traitement des images faciales dépendent amplement des performances de la détection de visages. Cette étape consiste à identifier et à localiser le visage dans l'image acquise au départ, indépendamment de la position, de l'échelle, de l'orientation et de l'éclairage. C'est un problème de classification où on assigne l'image à la classe visage ou à la classe non visage.

Les algorithmes de détection de visage dans les images faciales 2D (images couleurs ou à niveaux de gris) peuvent être classés en quatre groupes [58].

- **Approches basées sur les connaissances** (Knowledge-Based Methods): ces méthodes se basent sur les caractéristiques du visage comme le nez, la bouche, les yeux et la relation entre eux pour classer les images «visage et non visage. L'inconvénient de ce type de méthodes est qu'elles présentent un problème pour détecter le visage lorsque l'arrière-plan est complexe.
- **Approches basées sur les caractéristiques invariantes** (Feature Invariant Approaches) : l'objectif de ces algorithmes est de trouver des caractéristiques structurelles qui existent même lorsque les conditions sont variées (l'éclairage, la pose, l'expression faciale, etc.) telle que la texture ou la signature de couleur de la peau. Puis, ces caractéristiques invariantes ont été utilisées pour la localisation des visages en présence de ces problèmes.
- **Approches de correspondance de modèle** (Template Matching): l'idée de ces méthodes est de calculer la corrélation entre l'image d'entrée et un modèle pour la détection. Plusieurs modèles standards d'un visage sont stockés pour décrire le visage. Ces méthodes rencontrent encore quelques problèmes de robustesse liés aux variations de lumière et d'échelle.
- **Approches basées sur l'apparence** (Appearance based methods): ces méthodes utilisent le même principe des approches correspondance de modèle. Néanmoins, les modèles ici sont tirés d'un ensemble d'images d'apprentissage qui devrait capturer la variabilité que représente l'apparence du visage.

Plusieurs travaux de recherches ont été effectués dans ce domaine. Ils ont donné lieu au développement d'une multitude de techniques allant de la simple détection du visage, à la

localisation précise des régions caractéristiques du visage, tels que les yeux, le nez, les narines, les sourcils, la bouche, les lèvres, les oreilles, etc. Cependant, les solutions proposées jusqu'à maintenant sont loin d'être satisfaisantes car elles fonctionnent uniquement dans des environnements contrôlés, et par conséquent, elles ne gèrent pas la variabilité des conditions d'acquisition de la vie quotidienne, notamment : La pose, la présence ou absence des composantes structurales (la barbe, la moustache et les lunettes), les occultations et les conditions d'illumination [59].

2.2.2 Détection des yeux

Il y a principalement deux contraintes pour la détection des yeux, la première pour déterminer la présence des yeux et la seconde pour localiser précisément leur position. Pour localiser l'œil, généralement les équipes de recherche utilisent le centre de la pupille. Pour la localisation des yeux, il existe des méthodes intrusives où on utilise un système sous forme de casque devant être porté par un individu et des méthodes dites non intrusives où la détection des yeux s'effectue sans avoir à installer aucun appareil contraignant sur le sujet. Dans chacune de ces méthodes, on peut retrouver des méthodes de localisation actives dans lesquelles on utilise un éclairage proche infrarouge où la localisation s'effectue à partir de séquences d'images vidéo ainsi que des méthodes de localisation passives où seulement une image originale est utilisée [60].

La méthode retrouvée à plusieurs reprises dans la littérature consiste à déduire l'image de différence illuminée/éteinte. Cette technique utilise une image sur deux qui a un éclairage IR allumés et l'autre image où l'éclairage est éteint ou encore l'éclairage de la seconde image est décentré par rapport au centre optique de la caméra. Le problème avec cette méthode est qu'il faut développer un système électronique qui permet de synchroniser la capture d'image avec l'éclairage [61].

2.2.3 Correction de position

Pour diverses raisons, en particulier à cause du jeu mécanique de la fixation des pièces, ou un déplacement involontaire de la caméra ou des pièces elles-mêmes, les objets observés n'occupent pas toujours exactement la même position dans le champ de vue. Si on est capable, de déterminer la position effective des pièces, on peut ramener celles-ci en position nominale par réalignement (translation et rotation, éventuellement remise à l'échelle). Cela permet de ramener l'objet inspecté dans une position telle que les régions d'intérêt utiles peuvent rester fixes [62].

C'est une méthode simple qui augmente en général les performances du système. Elle effectue souvent une première réduction des données et elle atténue les effets d'une différence de conditions lors des prises de vues, pour restituer aussi fidèlement que possible le signal d'origine [63].

Dans notre travail, les images faciales que nous avons utilisées présentent une pose frontale. Nous commençons par convertir les images RVB en images à niveaux de gris, ensuite, la taille de ces images a été normalisée pour pouvoir faire la comparaison. La normalisation de l'image faciale est effectuée en utilisant les centres des yeux pour pivoter le visage dans le sens horaire avec un angle θ donné par la formule 2.1. La Figure 2.1 présente un exemple de correction de la position du visage.

$$\theta = \tan^{-1} \left(\frac{R_y - L_y}{R_x - L_x} \right) \quad 2.1$$

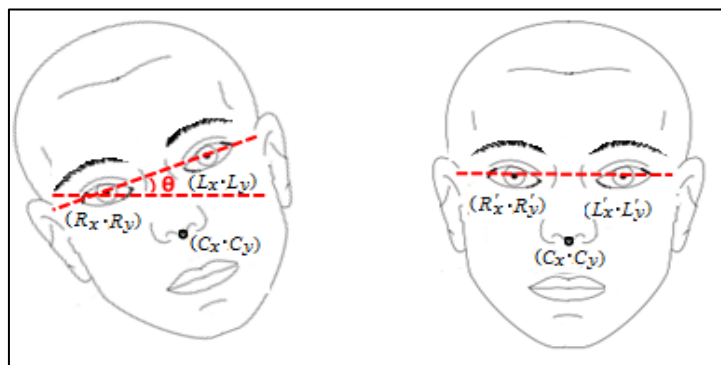


Figure 2.1: Correction de la position du visage

Après la rotation, les nouvelles coordonnées des centres des yeux sont données par les équations :

$$R'_x = C_x + (R_x - C_x). \cos \theta - (R_y - C_y). \sin \theta \quad 2.2$$

$$R'_y = C_x + (R_x - C_x). \sin \theta + (R_y - C_y). \cos \theta \quad 2.3$$

$$L'_x = C_x + (L_x - C_x). \cos \theta - (L_y - C_y). \sin \theta \quad 2.4$$

$$L'_y = C_x + (L_x - C_x). \sin \theta + (L_y - C_y). \cos \theta \quad 2.5$$

Avec : (C_x, C_y) comme centre de rotation de l'image .

(R_x, R_y) et (L_x, L_y) les coordonnées des yeux gauche et droite pour le visage avant la correction.

(R'_x, R'_y) et (L'_x, L'_y) les coordonnées des yeux (gauche et droite) pour le visage après la correction de pose.

2.3 Extraction des caractéristiques

L'extraction des caractéristiques consiste à exprimer l'image du visage (l'image faciale) sous une forme d'un vecteur appelé vecteur caractéristique composé d'un ensemble de caractéristiques pertinentes et non redondantes et utiles pour la perception, la mémorisation et la reconnaissance d'un visage. C'est l'étape la plus importante dans la quasi-totalité des systèmes de vision par ordinateur. Parmi les méthodes utilisées dans les systèmes de vérification de la parenté pour extraire les caractéristiques d'une image faciale on trouve :

2.3.1 Descripteurs de texture

La texture est souvent employée pour décrire l'état d'une surface d'un objet. Elle joue ainsi un rôle prépondérant dans l'analyse et l'interprétation d'une grande variété d'images.

On définit la texture d'un pixel comme étant l'information apportée par l'ensemble des pixels appartenant à son voisinage, alors que, la texture d'une région donnée désigne les caractéristiques de la répartition spatiale des niveaux de gris des pixels dans cette région [64].

La diversité des images, ainsi que la complexité de donner une définition précise de la texture a permis l'émergence de plusieurs méthodes d'analyse de la texture.

Les méthodes d'analyse de la texture ont pour but d'extraire un ensemble d'attributs ou paramètres pouvant décrire les caractéristiques de la texture. Ces attributs doivent être représentatifs, pertinents et discriminants de façon qu'on puisse discerner une texture parmi d'autres [65].

Dans notre travail, les descripteurs de texture que nous avons utilisés sont les suivants :

a. Motifs binaires locaux (LBP : Local Binary Pattern)

Les motifs binaires locaux ont initialement été proposés par T. Ojala en 1996 [66] afin de caractériser les textures présentes dans des images à niveaux de gris. Ils consistent à attribuer à chaque pixel P de l'image $I(i, j)$ à analyser, une valeur caractérisant le motif local autour de ce pixel. Ces valeurs sont calculées en comparant le niveau de gris du pixel central P aux valeurs des niveaux de gris des pixels voisins.

Le concept du *LBP* est simple, il propose d'assigner un code binaire à un pixel en fonction de son voisinage. Ce code décrivant la texture locale d'une région est calculé par seuillage d'un voisinage avec le niveau de gris du pixel central. Afin de générer un motif binaire, tous les voisins prendront alors une valeur "1" si leur valeur est supérieure ou égale au pixel

courant et "0" autrement (voir **Figure 2.2**). Les pixels de ce motif binaire sont alors multipliés par des poids et sommés afin d'obtenir un code LBP du pixel courant. On obtient donc pour toute l'image, des pixels dont l'intensité se situe entre 0 et 255 comme dans une image à 8 bits ordinaire. Plutôt que de décrire l'image par la séquence des motifs *LBP*, on peut choisir comme descripteur de texture un histogramme de dimension 255.

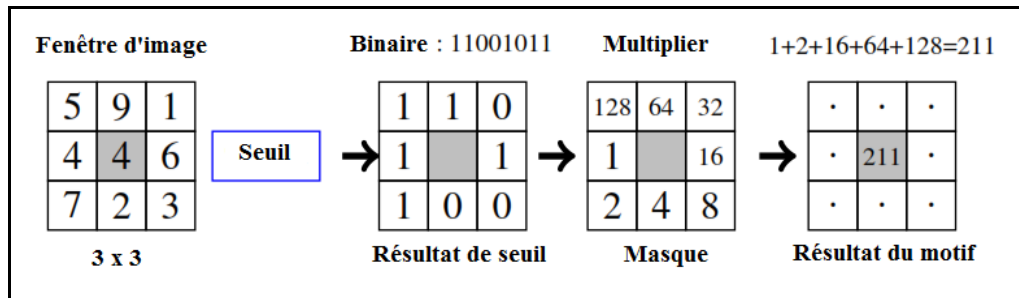


Figure 2.2: Construction d'un motif binaire et calcul du code LBP

Pour calculer un code *LBP* dans un voisinage de p pixels, on compte simplement les occurrences de niveaux de gris i_n plus grands ou égaux la valeur du pixel central (i_c) de coordonnées (x_c, y_c) .

$$LBP(x_c, y_c) = \sum_{n=0}^{p-1} s(i_n - i_c) \times 2^n \quad 2.6$$

Où $s(x)$ est la fonction signe et où i_n et i_c sont respectivement les niveaux de gris d'un pixel voisin et du pixel central.

$$s(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{Sinon} \end{cases} \quad 2.7$$

1. LBP multi échelle

Le concept du *LBP* multi-échelle, est fondé sur le choix du voisinage afin de calculer un code LBP pour pouvoir traiter les textures à différentes échelles. Un voisinage pour un pixel central est réparti sur un cercle et construit à partir de deux paramètres : le nombre de voisins "P" sur le cercle et un rayon "R" pour définir une distance entre un pixel central et ses voisins. Les coordonnées d'un voisin ne sont pas forcément situées au centre d'un pixel. Dans ce cas, le niveau de gris est déterminé par l'intermédiaire d'une interpolation. **La Figure 2.3** illustre différents voisinages obtenus pour différentes valeurs du couple (P,R) [68].

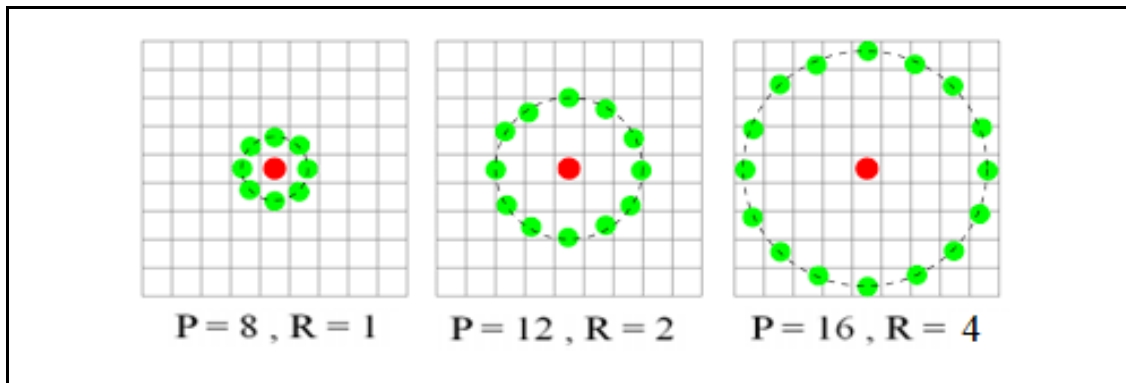


Figure 2.3 : LBP multi-échelle. Exemple de voisinages obtenus pour trois valeurs de (P,R)

2. LBP uniforme

$LBP_{P,R}^U$ est une extension à l'opérateur d'origine. Cette extension est utilisée pour réduire la longueur du vecteur d'entité. Il a été inspiré par le fait que certains motifs binaires se produisent plus fréquemment dans les images texturées que d'autres. LBP est considéré comme uniforme si le motif binaire contient au plus deux transitions binaires (de 0 à 1 ou de 1 à 0).

Une fois le code $LBP_{P,R}^U$ calculé pour tous les pixels de l'image, on calcule l'historgramme de cette image pour former un vecteur de caractéristiques représentant l'image faciale. Le nombre de cases d'historgramme LBP (N_{bins}) calculé utilisant l'équation suivante :

$$N_{bins} = p(p - 1) + 3 \quad 2.8$$

L'utilisation d'un code LBP uniforme, noté LBP^{U_2} à deux avantages. Le premier est le gain en mémoire et en temps de calcul. Le deuxième est que LBP^{U_2} permet de détecter uniquement les textures locales importantes, comme les spots, les fins de ligne, les bords et les coins [69].

b. Motifs ternaires locaux (LTP : Local Ternary Patterns)

Il s'agit de la généralisation des motifs binaires locaux au système ternaire. Cette méthode a été proposée par Liao [70] comme solution au problème de sensibilité qu'éprouve le LBP face au bruit aléatoire et celui de quantification. Le principe est le suivant : alors que les LBP appliquaient un seuil égal à la valeur du pixel central, la conversion en Motifs ternaires locaux (LTP) attribue à la valeur 0 aux pixels dont la valeur se trouve dans un voisinage de la valeur du pixel central, 1 à ceux dont la valeur est au-delà de ce voisinage et -1 à ceux dont la valeur

est en dessous. Autrement dit, pour u un pixel périphérique d'un entourage à convertir, i_c la valeur du pixel central et t le voisinage, la valeur $S(u, i_c, t)$ est donnée par l'équation 2.9.

$$S(u, i_c, t) = \begin{cases} 1 & \text{si } u \geq i_c + t \\ 0 & \text{si } |u - i_c| < t \\ -1 & \text{si } u \leq i_c - t \end{cases} \quad 2.9$$

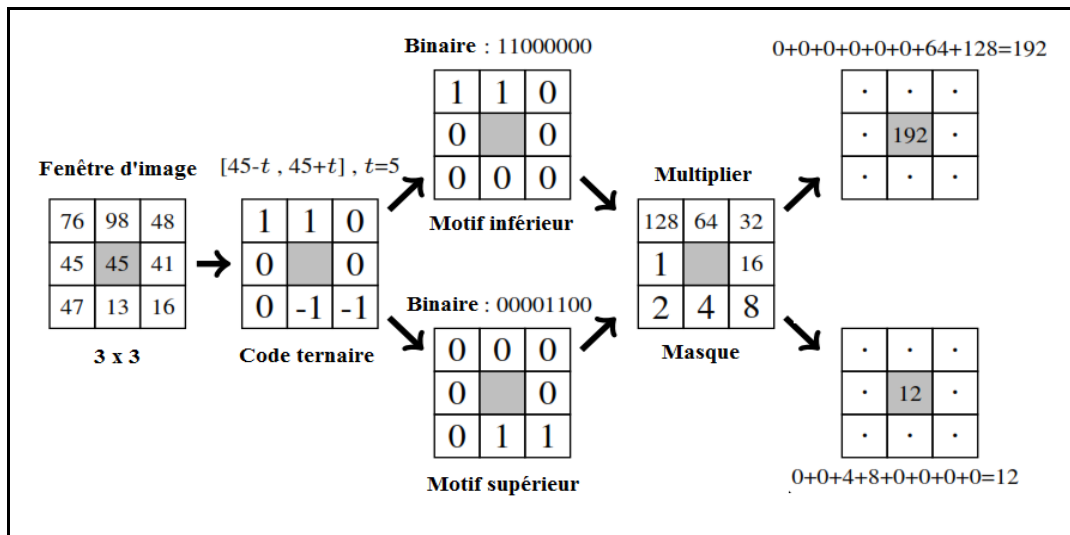


Figure 2.4 : Les étapes nécessaires du descripteur LTP.

Ensuite, ce code ternaire peut être transformé soit en un nombre réel soit en code binaire pour passer à la phase suivante. Dans le premier cas on pourrait utiliser un code de valeurs 3^n similaire au code binaire 2^n . Quant à Liao et al [71], ils ont divisé le code ternaire en deux codes binaires traités séparément et rassemblés ensuite lors de la phase de comparaison. Cette méthode a l'avantage de garder le système simple d'élimination des patterns non-uniformes (voir la Figure 2.4).

c. Motifs directionnels locaux (LDP : Local Directional Patterns)

Proposé par Jabid et al. [72], le LDP est un motif de texture local qui calcule les valeurs de réponse du bord dans huit directions à chaque position de pixel et code la texture à l'aide de masques de Kirsch dans huit orientations différentes centrées sur sa position $M_0, M_1, M_2, \dots, M_7$. La Figure 2.5 montre les masques de Kirch utilisés dans cette méthode.

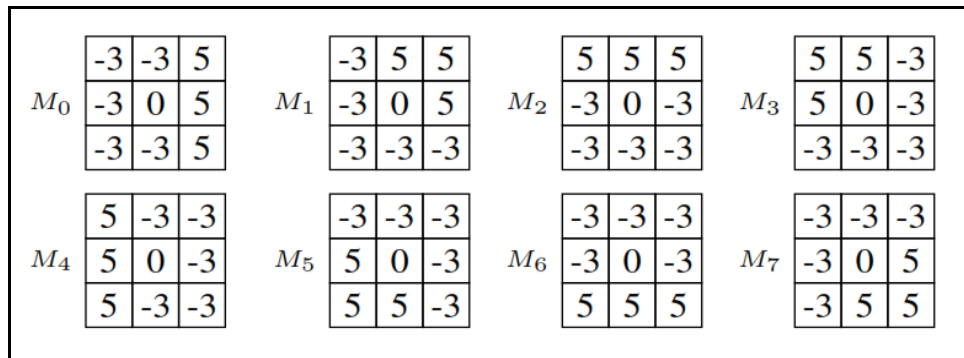


Figure 2.5 : Masques de bord Kirsch (LDP) à huit directions.

Pour chaque pixel d'une image d'entrée, le résultat obtenu est huit valeurs de réponse de bord m_0, m_1, \dots, m_7 . En fait, toutes les réponses ne sont pas équivalentes, où les k directions les plus importantes sont sélectionnées [73]. Les k premières réponses binaires directionnelles sont définies sur 1, et le reste des bits directionnels devient 0. Par conséquent, le code LDP pour chaque pixel est obtenu à l'aide de la formule suivante :

$$LDP_k = \sum_{i=0}^7 b_i (m_i - m_k) 2^i \tag{2.10}$$

$$b_i(a) = \begin{cases} 1 & \text{si } a \geq 0 \\ 0 & \text{autrement} \end{cases} \tag{2.11}$$

Après avoir obtenu le code LDP pour tous les pixels (r, c) , la formule suivante donne le nombre d'histogrammes obtenus par LDP.

$$H(\tau) = \sum_{r=1}^M \sum_{c=1}^N f(LDP_k(r, c), \tau) \tag{2.12}$$

La formule suivante donne le nombre d'histogrammes obtenus par LDP.

$$N_{bins} = \frac{8!}{k!(8-k)!} \tag{2.13}$$

La Figure 3.6 illustre le processus de codage LDP, par exemple $K = 3$.

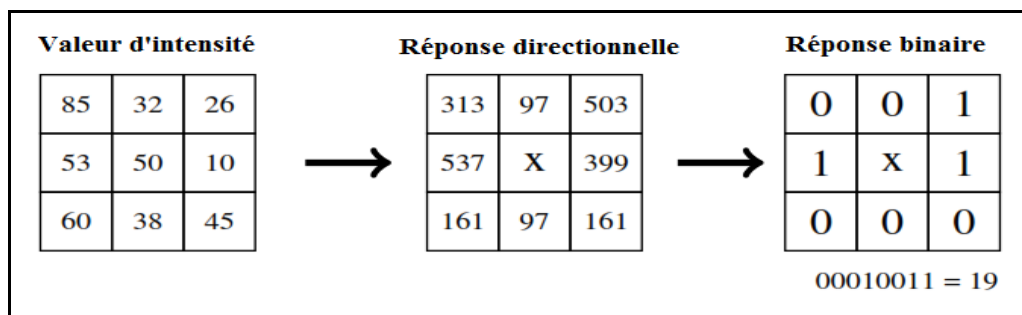


Figure 3.6 : Les étapes nécessaires du descripteur LDP.

d. Quantification de Phase Locale (LPQ)

Le descripteur de texture Quantification de Phase Locale (ou Local Phase Quantization LPQ) a été introduit pour la première fois par Ojansivu et Heikkila [74]. LPQ est un descripteur de texture basé sur l'application de la transformée de Fourier à court terme (STFT) (Short-Time Fourier Transform), l'information de LPQ extraite en utilisant (STFT) 2D calculée sur un voisinage $M * M$ rectangulaire centré à chaque position de pixel x de l'image $f(x)$ définie par l'équation suivante :

$$F(u, x) = \sum_{y \in N_x} f(x - y) e^{-j2\pi u^T y} = W_u^T f_x \quad 2.14$$

Où w_u est le vecteur de base du 2-DDFT à la fréquence u (un vecteur 2-D) et $f(x)$ est un autre vecteur contenant tous les échantillons d'images M^2 de N_x .

Dans LPQ, seuls quatre coefficients complexes sont considérés, correspondant à des fréquences 2D : $u_1 = [a, 0]^T$, $u_2 = [0, a]^T$, $u_3 = [a, a]^T$, $u_4 = [a, -a]^T$, où a est un scalaire suffisamment petit.

Pour chaque pixel, le vecteur obtenu est représenté par l'équation suivante :

$$F_x = [F(u_1, x), F(u_2, x), F(u_3, x), F(u_4, x)] \quad 2.15$$

Les informations de phase dans les coefficients de Fourier sont enregistrées en observant les signes des parties réelles et imaginaires de chaque composante dans $F(x)$. Cela se fait en utilisant une quantification scalaire définie par l'équation suivante :

$$q_i = \begin{cases} 1 & \text{si } g_i \geq 0 \\ 0 & \text{Sinon} \end{cases} \quad 2.16$$

Où g_i est la i ème composante du vecteur $G(x) = [R_e \{F(x)\}, Im\{F(x)\}]$.

Les huit coefficients binaires résultants q_i représentent le modèle de code binaire. Ce code est converti en nombres décimaux compris entre 0 et 255. À partir de cela, l'histogramme LPQ a 256 bacs cases (bins). La chaîne binaire est convertie en nombre décimal par l'expression suivante pour produire une étiquette de LPQ. **La Figure 2.7** résume l'ensemble de ces étapes.

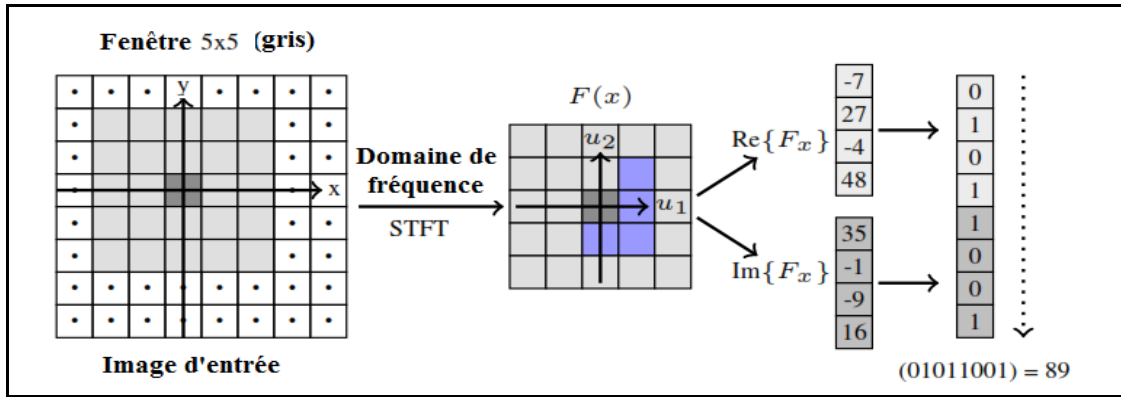


Figure 2.7: les étapes nécessaires du descripteur LPQ.

e. Caractéristiques d'Image Statistiques Binarisés (BSIF : Binarized Statistical Image Features)

Le BSIF (Binarized Statistical Image Features) est un descripteur de texture d'image proposé par Kannala et Rahtu [75].

L'idée de BSIF est d'apprendre automatiquement un ensemble fixe de filtres à partir d'un petit ensemble d'images naturelles au lieu d'utiliser des filtres fabriqués à la main. L'ensemble de filtres est appris à partir d'un ensemble d'apprentissage de correctifs d'image naturels en maximisant l'indépendance statistique des réponses de filtre.

Etant donné un patch d'image I de taille $L * L$ pixels et un filtre linéaire W_k de même taille, la réponse du filtre S_k est obtenue par l'équation suivante :

$$S_k = \sum_{i,j} W_k(i,j) I(i,j) = W_k'^T I' \tag{2.17}$$

Où W_k' et I' sont les vecteurs de taille $L * L$ (forme vectorisée des tableaux $2d$, W et I).

La caractéristique binarisée b_k est obtenue par l'équation suivante :

$$q_i = \begin{cases} 0 & \text{si } S_k \geq 0 \\ 1 & \text{Sinon} \end{cases} \tag{2.18}$$

Les filtres W_k sont appris en utilisant une analyse indépendante des composants en maximisant l'indépendance statistique de S_k .

Le nombre de cases d'historgramme (N_{bins} N cases) obtenu par le descripteur BSIF est calculé à l'aide de l'équation suivante :

$$N_{bins} = 2^{N_f} \tag{2.19}$$

La Figure 2.8 montre les étapes utilisées pour avoir l'image BSIF.

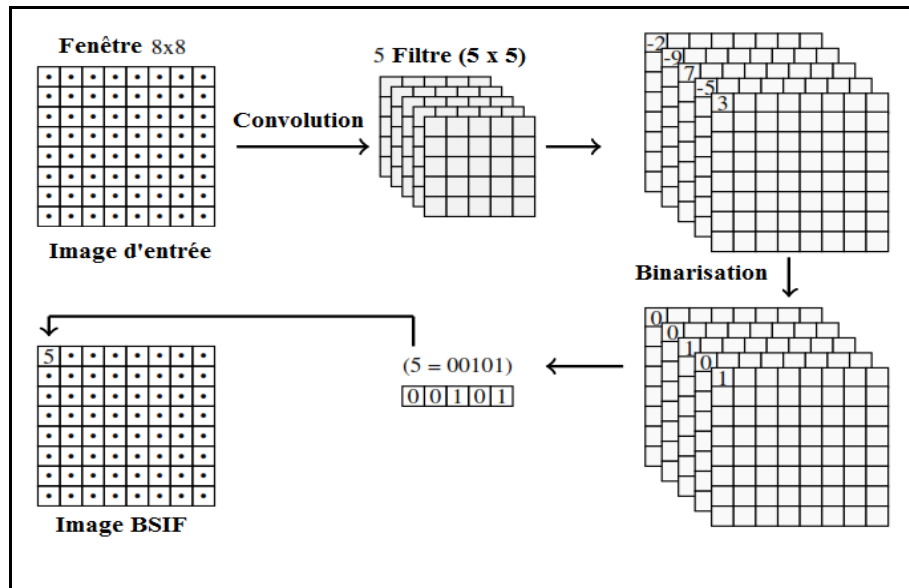


Figure 2.8 : les étapes nécessaires du descripteur BSIF

f. Descripteur local Weber (WLD)

Le WLD, repose sur le fait que la perception humaine d'un modèle dépend non seulement du changement d'un stimulus comme le son et l'éclairage, mais aussi de l'intensité originelle du stimulus. Le WLD se compose plus précisément, de deux composantes : l'excitation différentielle et l'orientation. La composante d'excitation différentielle est une fonction du rapport entre deux termes : l'une représente les différences d'intensités relatives d'un pixel courant par rapport à ses voisins ; l'autre est l'intensité du pixel courant. La composante d'orientation est quant à elle, l'orientation du gradient du pixel courant [76].

Le WLD est inspiré par la loi de Weber, qui est une loi psychologique. La loi de Weber, plus simplement énoncée, dit que la taille d'une différence juste perceptible (c'est-à-dire, ΔI) est une proportion constante de la valeur du stimulus d'origine. Ainsi, par exemple, dans un environnement bruyant, il faut crier pour être entendu tandis qu'un chuchotement fonctionne dans une pièce calme.

$$\frac{\Delta I}{I} = k \tag{2.20}$$

Où ΔI représente le seuil d'incrément (différence juste perceptible pour la discrimination); I représente l'intensité initiale du stimulus et k signifie que la proportion du côté gauche de l'équation reste constante malgré les variations du terme I . La fraction $\Delta I/I$ est connue sous le nom de fraction de Weber.

Il indique que le changement d'un stimulus (son, éclairage) qui sera juste perceptible, est un rapport constant du stimulus original. Quand le changement est plus petit que ce rapport constant du stimulus original, un être humain le reconnaîtrait comme un bruit de fond plutôt qu'un signal valide. Motivé par ce point, pour un pixel donné, la composante d'excitation différentielle du descripteur local de Weber proposé (WLD), est calculée sur la base du rapport entre les deux termes : l'une représente les différences d'intensités relatives d'un pixel courant par rapport à ses voisins (par exemple, 3×3 région carrée) ; l'autre est l'intensité du pixel courant.

Avec la composante d'excitation différentielle, nous tentons d'extraire les motifs saillants locaux de l'image d'entrée. En outre, nous calculons également l'orientation du gradient du pixel courant. C'est-à-dire, pour chaque pixel de l'image d'entrée, on calcule deux composantes de la caractéristique WLD (c'est-à-dire l'excitation différentielle et l'orientation en gradient). En combinant la caractéristique WLD par pixel, nous représentons une image d'entrée (ou région d'image) avec un histogramme, que nous appelons ici, histogramme WLD. Dans notre cas, la fonction WLD est calculée en pixel [77].

La Figure 2.8 montre les étapes utilisées pour avoir l'image WLD.

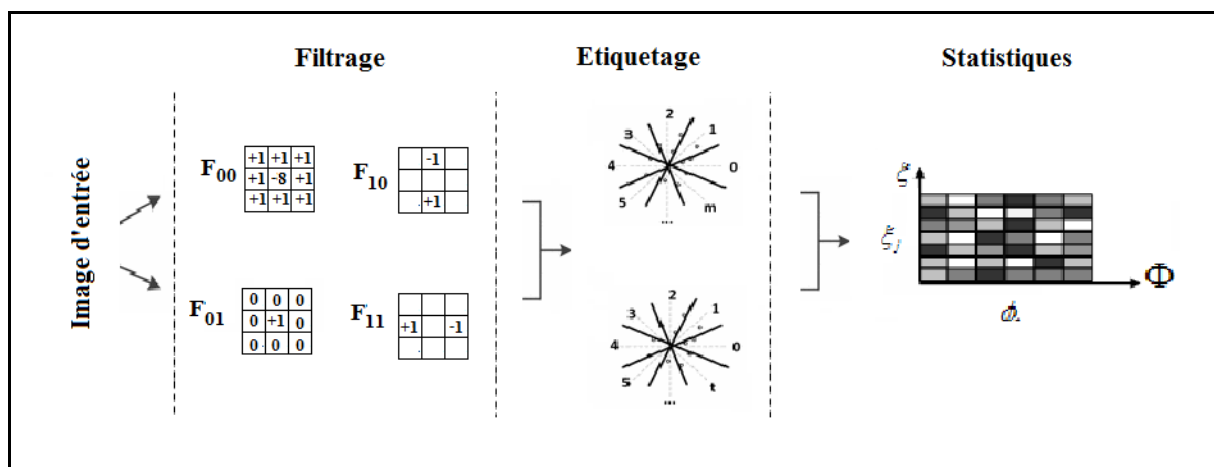


Figure 2.9: les étapes nécessaires du descripteur WLD.

En ce qui concerne le descripteur LBP, il représente une image d'entrée en construisant des statistiques sur les variations de micro-patterns locales. Par contre, le WLD, calcule d'abord les micro-schémas saillants (c'est-à-dire l'excitation différentielle), puis construit des statistiques sur ces motifs saillants avec l'orientation du gradient de point courant.

La plus petite taille des régions de soutien pour WLD permet de capturer plus de traits saillants locaux. En outre, le WLD peut être facilement étendu pour extraire les fonctionnalités multi-granularité par des techniques d'analyse multi-échelle.

g. Histogramme de Gradient Orienté (HoG)

Le descripteur HoG est introduit par Dalal et Triggs [78]. L'idée essentielle derrière l'histogramme de gradient orienté c'est que l'apparence locale et la forme d'objet dans une image peut être décrite par la distribution d'intensité des gradients ou de direction des contours. La mise en œuvre de ce descripteur peut être obtenue en divisant l'image en petites régions connectées, appelées cellules, et pour chaque cellule on calcule un histogramme des directions de gradient ou des orientations de contour pour les pixels dans la cellule. La combinaison de ces histogrammes représente alors le descripteur.

Le descripteur HoG maintient quelques avantages clés par rapport aux autres méthodes. Puisque le descripteur histogramme de gradient orienté opère sur les cellules localisées, la méthode maintient l'invariance à des transformations géométriques et photométriques, ces changements ne feront leur apparition que dans les larges régions d'espaces.

➤ Calcul histogramme d'orientation de gradient :

- **Normalisation de l'image :** Les images d'entrée de notre système sont dans un premier temps converti en niveau de gris. Elles sont ensuite normalisées selon l'information de luminance.
- **Calcul de l'image gradient :** Le gradient, en un pixel d'une image numérique, est un vecteur caractérisé par son amplitude et sa direction. L'amplitude est directement liée à la quantité de variation locale des niveaux de gris. La direction du gradient est orthogonale à la frontière qui passe au point considéré.

Le calcul du gradient est une étape critique dans la formation de descripteurs. La justesse des orientations calculées, et des histogrammes, dépend de cette étape et les résultats sont donc étroitement liés à la méthode employée pour calculer le gradient dans l'image. Calcul rapide du gradient peut être fait par des filtres de dérivation simple: Gaussien, Médian....etc.

Le gradient d'une image se calcule comme suit (équation 2.21) :

$$G(x, y) = \sqrt{(G_x(x, y))^2 + (G_y(x, y))^2} \quad 2.21$$

Et la direction du gradient est donnée par :

$$\alpha(x, y) = \arctan \left(\frac{G_y(x, y)}{G_x(x, y)} \right) \quad 2.22$$

$G_x(x, y)$ et $G_y(x, y)$ sont les gradients dans la direction horizontale et verticale, au pixel (x, y) .

Un histogramme est un tableau de nombres dans lequel chaque élément correspond à la fréquence d'apparition d'un intervalle de valeurs pour un ensemble donnée. Dans le cadre d'une image, par exemple, chaque case de l'historgramme peut représenter les pixels de la même couleur. L'image est découpée en plusieurs cellules de petite taille (Voir **Figure 2.10**) et pour chaque cellule un histogramme est calculé. Chaque pixel d'une cellule vote pour une orientation entre 0 et 180 dans le cas non signé, ou entre 0 et 360 dans le cas signé [78].

L'étape suivante est la normalisation des descripteurs, afin d'éviter les disparités dues aux variations d'illumination, ainsi que l'introduction de redondance dans le descripteur. Pour cela, les cellules sont regroupées par bloc (concaténation des histogrammes des cellules d'un bloc), le vecteur de valeur du bloc est ensuite normalisé. Les blocs se recouvrent, donc une même cellule peut participer plusieurs fois au descripteur final.

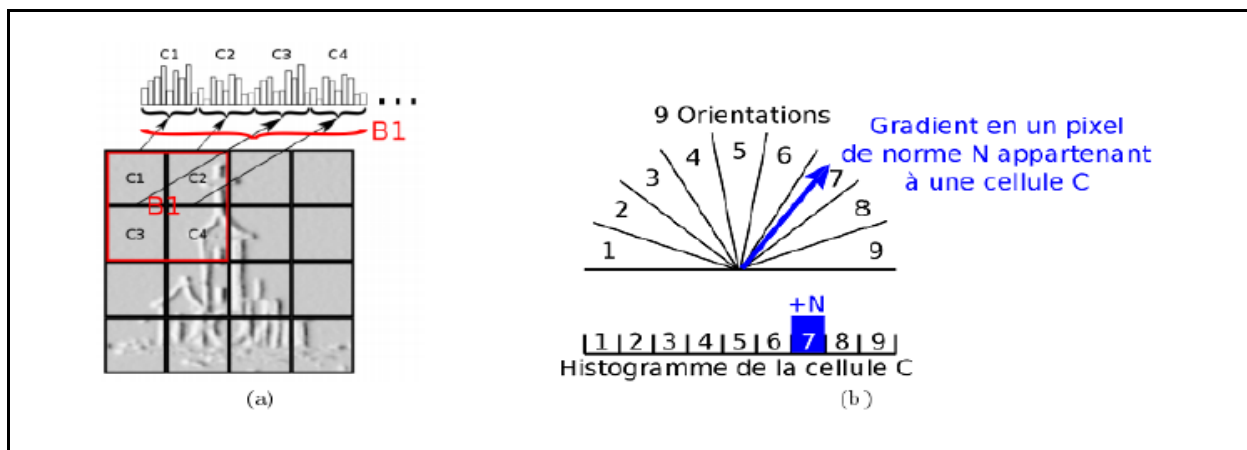


Figure 2.10: Procédure simplifiée pour le descripteur HOG : (a) découpage en cellules et formation de l'historgramme par bloc, (b) vote d'un pixel selon l'orientation de ses gradients.

Le vecteur de HOG pour chaque cellule est normalisé en utilisant des vecteurs dans leurs blocs respectifs. En utilisant la méthode norme L_2 pour le bloc de normalisation, le facteur de normalisation est calculé en utilisant l'équation :

$$V_n = \frac{V}{\sqrt{\|V_2\|^2 + e^2}} \tag{2.23}$$

Où : V est le vecteur non normalisé contenant tous les histogrammes d'un bloc donné.

V_2 : Est la norme L_2 du vecteur (v) du descripteur. e : est une petite constante.

2.3.2 Descripteurs d'apprentissage en profond

Les réseaux de neurones convolutifs (CNN) (Convolutional Neural Network) sont un type de réseau de neurones spécialisés pour le traitement des données ayant une topologie semblable à une grille. Les exemples comprennent des données de type série temporelle, qui peuvent être considérées comme une grille 1D en prenant des échantillons à des intervalles de temps réguliers et des données de type image, qui peuvent être considérées comme une grille 2D de pixels. Les réseaux convolutifs ont connu un succès considérable dans les applications pratiques. Le nom « réseau de neurones convolutif » indique que le réseau emploie une opération mathématique appelée convolution. La convolution est une opération linéaire spéciale. Les réseaux convolutifs sont simplement des réseaux de neurones qui utilisent la convolution à la place de la multiplication matricielle dans au moins une de leurs couches. Ils ont de larges applications dans la reconnaissance de l'image et de la vidéo, les systèmes de recommandations et le traitement du langage naturel [79].

Le réseau convolutif est un type de réseau de neurones feed-forward, il a été inspiré par des processus biologiques. Le modèle de connectivité entre les neurones d'un CNN s'inspire de l'organisation du cortex visuel animal. Les neurones corticaux individuels répondent aux stimuli dans une région restreinte de l'espace connu sous le nom de champ réceptif. Les champs réceptifs des différents neurones se chevauchent partiellement de sorte qu'ils couvrent le champ visuel. La réponse d'un neurone individuel aux stimuli dans son champ réceptif peut être approchée mathématiquement par une opération de convolution [80].

Dans sa forme la plus générale, la convolution est une opération sur deux fonctions d'argument réel. Pour comprendre la motivation derrière la convolution, nous commençons par des exemples de deux fonctions qu'on pourrait utiliser. Supposons que nous suivons l'emplacement d'un vaisseau spatial avec un capteur laser. Notre capteur laser fournit une sortie $x(t)$ qui est la position du vaisseau spatial au moment t . x et t sont des réelles, c'est-à-dire que nous pouvons obtenir une lecture différente du capteur laser à tout moment. Supposons maintenant que notre capteur laser soit quelque peu bruité. Pour obtenir une estimation moins bruitée de la position du vaisseau spatial, nous aimerions combiner plusieurs mesures. Bien sûr, les mesures plus récentes sont plus pertinentes, donc nous voulons que ces mesures soient une moyenne pondérée et donner plus de poids aux mesures récentes [81]. Nous pouvons le faire avec une fonction de pondération $w(a)$, où a est l'âge de la mesure. Si nous appliquons une telle opération de moyenne pondérée à chaque instant, nous obtenons une nouvelle fonction qui fournit une estimation lisse de la position du vaisseau spatial :

$$s(t) = \int x(a)w(t - a)da \quad 2.24$$

Cette opération s'appelle convolution. L'opération de convolution est généralement désignée par un astérisque :

$$s(t) = (x * w)t \quad 2.25$$

Dans notre exemple, l'idée d'un capteur laser qui peut fournir des mesures à chaque instant dans le temps n'est pas réaliste. Habituellement, lorsque nous travaillons avec des données sur l'ordinateur, le temps sera discrétisé (numérisé), et notre capteur fournira des données à des intervalles réguliers. Dans notre exemple, il est plus réaliste de supposer que notre laser fournit une mesure une fois par seconde. L'index de temps t ne peut alors prendre que des valeurs entières. Si on suppose maintenant que x et w sont des entiers, on peut définir la convolution discrète :

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a) \quad 2.26$$

Enfin, nous utilisons souvent des convolutions sur plus d'un axe à la fois. Si nous utilisons une image bidimensionnelle x comme entrée.

$$s(i, j) = (x * w)(i, j) = \sum_m \sum_n x(m, n)w(i - m, j - n) \quad 2.27$$

Dans la terminologie du réseau convolutif, le premier argument (dans cet exemple, la fonction x) de la convolution est souvent appelé l'entrée (input) et le second argument (dans cet exemple, la fonction w) comme noyau (kernel). La sortie est parfois appelée (feature map) [82].

Les descripteurs basés sur L'apprentissage en profond utilisés pour notre approche sont les suivants :

a. VGG_16 (Visual Geometry Group 16):

VGG_16 est un modèle de réseau de neurones convolutif proposé par Simonyan et Zisserman [83]. Ce modèle a été formé par l'ensemble de données ImageNet où ImageNet est un ensemble de données de plus de 14 millions d'images appartenant à 1000 classes. L'architecture VGG-16 contient la couche d'entrée et de sortie, ainsi que plusieurs couches. La couche d'entrée est de taille fixe, l'image est passée à travers une pile de couches convolutionnelles, en utilisant des filtres avec le petit champ (3 * 3) et elle utilise également

des filtres de convolution ($1 * 1$), et a toujours utilisé le même remplissage et couche max-pool de filtre ($2 * 2$). Il suit cet arrangement trois couches FC suivant un empilement de couches convolutionnelles : les deux premières ont 4096 canaux chacune, la troisième effectue ILSVRC 1000 voies, et enfin une couche softmax 1000 voies.

b. VGG_Face (Visual Geometry Group Face):

Ce réseau de neurones à convolution préformé fournit des descripteurs spécialement formés pour la reconnaissance faciale. L'architecture du réseau est inspirée de l'implémentation CNN basée sur l'architecture VGG-16 mais ce modèle entraîné par l'ensemble de données VGG-Face, qui est un ensemble de données spécialement pour les images faciales (2.6 M images, 2.6 K personnes). Il a la même architecture et contient les mêmes couches (couches de convolution, pool max, entièrement connecté et soft max). Les caractéristiques sont extraites de la couche FC7 (la couche fully connected), et le nombre de ces caractéristiques est de 4096 [84].

La **Figure 2.11** présente l'extraction des caractéristiques de l'image faciale apprises de VGG-face (un réseau neuronal convolutionnel pré-entraîné). Les couches FC7 ont été choisies car les couches plus profondes contiennent des caractéristiques de niveau supérieur par rapport aux couches précédentes.

Constituée de 11 blocs contenant chacun un opérateur linéaire suivi d'une ou de plusieurs non linéarités telles que ReLU et pooling maximum.

Les 8 premiers blocs sont linéaires, tandis que les 3 derniers sont entièrement connectés avec la fonction d'activation ReLU. L'image d'entrée est RGB et sa taille est de 224×224 pixels. Les deux premières couches entièrement connectées contiennent 4 096 neurones et le dernier réduit la dimensionnalité à 2622 (nombre de classes de l'ensemble de données d'apprentissage). Le réseau a été formé par 2600000 images. Afin d'utiliser ce réseau, nous avons supprimé la dernière couche, qui a effectué une réduction de dimensionnalité pour la classification de l'ensemble de données d'apprentissage, obtenant ainsi un vecteur de caractéristiques de dimension ($1 * 4096$). Nous avons également dû redimensionner les images d'entrée de la base de données pour avoir les dimensions 224×224 ([85], [86]).

| layer | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|-----------|-------|---------|--------|---------|--------|-------|---------|---------|---------|---------|-------|---------|---------|---------|---------|---------|---------|-------|---------|
| type | Input | conv | relu | conv | relu | mpool | conv | relu | conv | relu | mpool | conv | relu | conv | relu | conv | relu | mpool | conv |
| name | | conv1_1 | relu_1 | conv1_2 | relu_2 | pool1 | conv2_1 | relu2_1 | conv2_2 | relu2_2 | pool2 | conv3_1 | relu3_1 | conv3_2 | relu3_2 | conv3_3 | relu3_3 | pool3 | conv4_1 |
| support | - | 3 | 1 | 3 | 1 | 2 | 3 | 1 | 3 | 1 | 2 | 3 | 1 | 3 | 1 | 3 | 1 | 2 | 3 |
| filt dim | - | 3 | - | 64 | - | - | 64 | - | 128 | - | - | 128 | - | 256 | - | 256 | - | - | 256 |
| num filts | - | 64 | - | 64 | - | - | 128 | - | 128 | - | - | 256 | - | 256 | - | 256 | - | - | 512 |
| stride | - | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| pad | - | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |

| layer | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 |
|-----------|---------|---------|---------|---------|---------|-------|---------|---------|---------|---------|---------|---------|-------|------|-------|------|-------|------|---------|
| type | relu | conv | relu | conv | relu | mpool | conv | relu | conv | relu | conv | relu | mpool | conv | relu | conv | relu | conv | softmax |
| name | relu4_1 | conv4_2 | relu4_2 | conv4_3 | relu4_3 | pool4 | conv5_1 | relu5_1 | conv5_2 | relu5_2 | conv5_3 | relu5_3 | pool5 | fc6 | relu6 | fc7 | relu7 | fc8 | prob |
| support | 1 | 3 | 1 | 3 | 1 | 2 | 3 | 1 | 3 | 1 | 3 | 1 | 2 | 7 | 1 | 1 | 1 | 1 | 1 |
| filt dim | - | 512 | - | 512 | - | - | 512 | - | 512 | - | 512 | - | - | 512 | - | 4096 | - | 4096 | - |
| num filts | - | 512 | - | 512 | - | - | 512 | - | 512 | - | 512 | - | - | 4096 | - | 4096 | - | 2622 | - |
| stride | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| pad | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Output

Figure 2.11: Couches des Modèles CNN (VGG-16 et VGG-Face).

c. ResNet (Residual network)

ResNet c'est un descripteur de visage proposé par Kaiming He et al [87]. Se compose d'une couche d'entrée et d'une sortie, ainsi que de plusieurs couches cachées. Les couches cachées d'un CNN se composent généralement de couches convolutives, de couches de mise en commun, de couches entièrement connectées et de couches de normalisation. Les caractéristiques profondes sont extraites de la couche f1000 (couche entièrement connectée) de cette architecture, et le nombre de ces fonctionnalités est de 1000. La Figure 2.12 présente les architectures pour ResNet, à partir de différents modèles de ResNet: ResNet-18, Resnet-50, ResNet -101 et ResNet -152 (un réseau neuronal convolutionnel pré-entraîné) et les blocs de construction sont indiqués entre parenthèses, avec le nombre de blocs empilés. L'échantillonnage à la baisse est effectué par conv3, conv4 et conv5 avec une foulée de 2. Le principe est le même pour les modèles ResNet, la différence dans le nombre de couches de convolution utilisant des filtres 3x3. Les couches 1000 ont été choisies en raison du fait que les couches plus profondes contiennent des caractéristiques de niveau supérieur par rapport aux couches précédentes.

| layer name | output size | 18-layer | 34-layer | 50-layer | 101-layer | 152-layer |
|------------|-------------|---|---|---|--|--|
| conv1 | 112×112 | 7×7, 64, stride 2 | | | | |
| conv2_x | 56×56 | 3×3 max pool, stride 2 | | | | |
| | | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| conv3_x | 28×28 | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$ |
| conv4_x | 14×14 | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$ |
| conv5_x | 7×7 | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| | 1×1 | average pool, 1000-d fc, softmax | | | | |
| FLOPs | | 1.8×10 ⁹ | 3.6×10 ⁹ | 3.8×10 ⁹ | 7.6×10 ⁹ | 11.3×10 ⁹ |

Figure 2.12 : Couches des Modèles CNN (ResNet).

2.4 Représentation du visage

Cette étape consiste à représenter l'image faciale d'une manière bien déterminée, et ce, dans le but d'augmenter les dimensions des descripteurs utilisés et par conséquent, obtenir un nombre important de caractéristiques des images faciales, ce qui nous permettra d'avoir plus de probabilités pour choisir les meilleures.

Les représentations de l'image faciale que nous avons utilisées dans ce travail sont les suivantes :

2.4.1 Multi bloc (MB)

MB est une variante qui remplace les valeurs d'intensité dans le calcul du descripteur par la valeur d'intensité moyenne des blocs d'image. La représentation des faces MB est une technique qui divise l'image en (n x n) blocs où n est le niveau voulu de MB. La Figure 2.13 illustre les représentations de la face MB ([88], [89]).

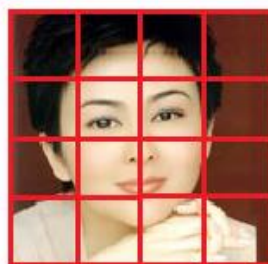


Figure 2.13: Exemple MB a quatre niveaux.

2.4.2 Multi Niveau (ML, Multi level)

La représentation des faces ML est une représentation pyramidale spatiale construite par séries de représentations MB. Le niveau de représentation des visages de niveau ML n est construit à partir des représentations de visage des niveaux 1, 2, ... n MB. La **Figure 2.14** illustre les représentations de visage ML ([90], [91]).

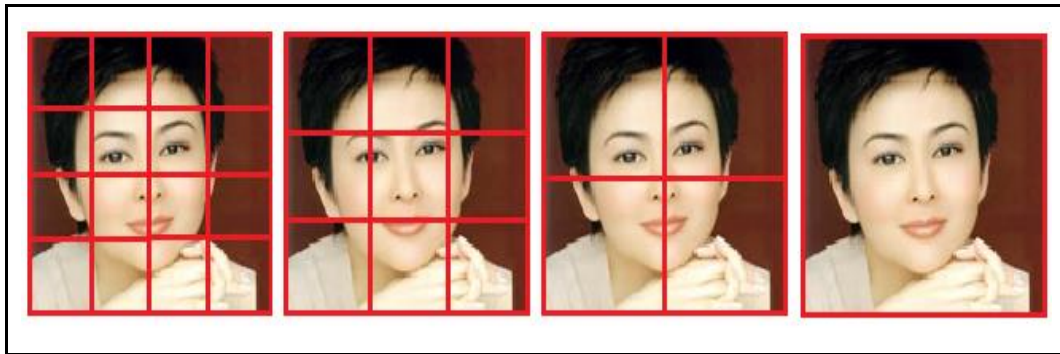


Figure 2.14 : Exemple ML a quatre niveaux.

2.4.3 Pyramide Multi niveaux (PML, Pyramide Multi level)

Inspiré des représentations ML et multi-échelles basées sur des pyramides, la représentation PML adopte une représentation pyramidale explicite de l'image d'origine. Cette pyramide représente l'image à différentes échelles. Pour chaque niveau ou échelle, une représentation MB correspondante est utilisée. Les sous-blocs PML ont la même taille, qui est déterminée par la taille de l'image et le niveau choisi. La **Figure 2.15** illustre les représentations de face PML [92].

$$\left[w' = w \cdot \frac{(n-1)}{n} , h' = h \cdot \frac{(n-1)}{n} \right] \quad 2.26$$

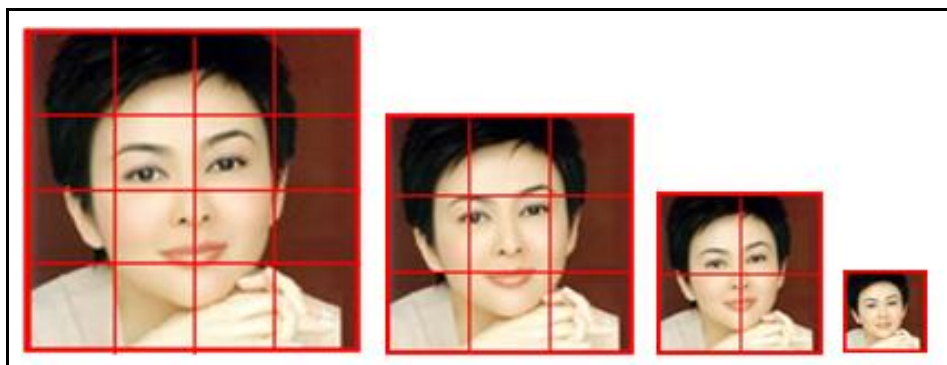


Figure 2.15: Exemple PML a quatre niveaux.

2.4.4 Combinaison (Descripteurs / Représentation faciale)

Cette étape proposée vise à combiner des descripteurs avec les représentations faciales, afin de créer de nombreuses nouvelles caractéristiques [93]. Tout descripteur peut être utilisé avec toutes les représentations faciales à tous les niveaux. La **Figure 2.16** montre la procédure d'extraction de caractéristiques utilisant un descripteur LBP avec la représentation PML, niveau 3.

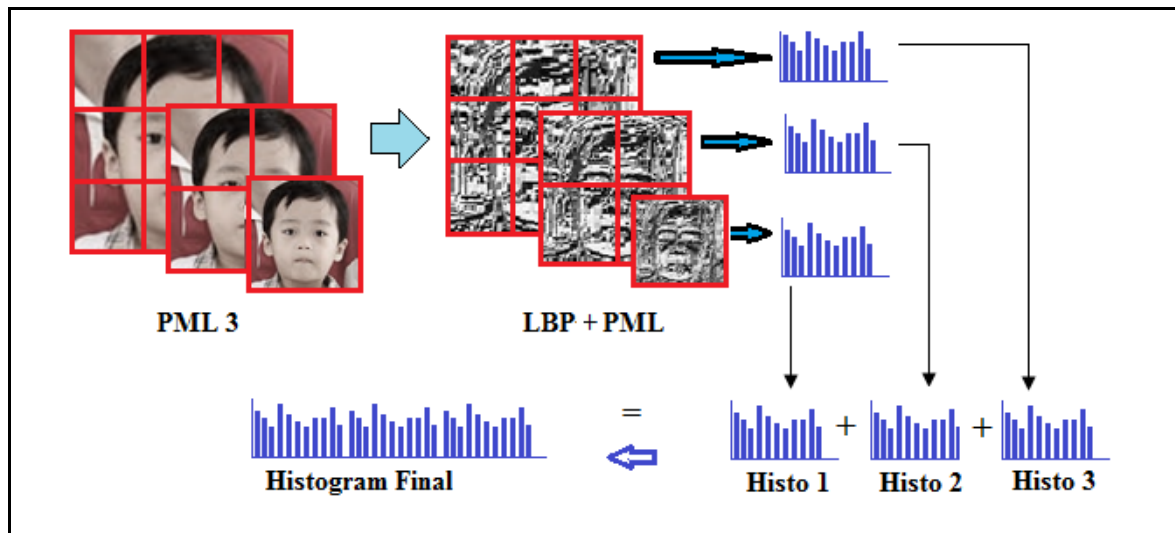


Figure 2.16: Exemple d'extraction de caractéristiques utilisant le descripteur LBP avec la représentation facial PLM.

2.5 Conclusion

Ce chapitre est consacré à la présentation des différentes méthodes utilisées pour l'extraction des caractéristiques. Nous avons commencé par présenter les techniques les plus importantes pour le prétraitement du visage. Ensuite, nous avons mis en évidence les méthodes d'extraction de caractéristiques les plus utilisées et qui ont reçu une attention croissante pendant ces dernières années. Les bases mathématiques nécessaires à la compréhension de fonctionnement de ces méthodes ont été largement présentées, où nous avons abordé les méthodes basées sur des descripteurs de texture ainsi que les méthodes basées sur l'apprentissage en profond (Deep Learning). Finalement, nous avons terminé ce chapitre par l'étude des méthodes de représentation du visage dans les traitements futurs, en particulier la représentation MB, la représentation ML et la représentation PML.

Chapitre 3

Décision finale de la vérification de la parenté

Sommaire

| | |
|--|-----------|
| 3.1 Introduction | 48 |
| 3.2 Représentation et Normalisation des caractéristiques | 48 |
| 3.2.1 Représentation des caractéristiques | 48 |
| 3.2.2 Normalisation des caractéristiques | 49 |
| 3.3 Réduction des caractéristiques | 49 |
| 3.3.1 Projection des caractéristiques | 50 |
| 3.3.2 Sélection des caractéristiques | 53 |
| 3.4 La Classification | 56 |
| 3.4.1 Similarité en Cosins | 56 |
| 3.4.2 Méthode des k plus proches voisins | 57 |
| 3.4.3 Machine à vecteurs de support (SVM) | 60 |
| 3.5 Validation croisée | 63 |
| 3.6 Performance d'un système biométrique | 64 |
| 3.6.1 Précision d'un système biométrique (Pr) | 64 |
| 3.6.2 Caractéristiques de fonctionnement du récepteur (ROC) | 64 |
| 3.7 Conclusion | 66 |

3.1 Introduction

Dans ce chapitre, nous présentons la dernière étape du système proposé qui consiste à la prise d'une décision finale qui nous permettra de déterminer l'existence (ou l'inexistence) d'une parenté entre deux personnes, et ce, à partir de leurs images faciales.

Tout d'abord, nous commençons par transformer les paires de vecteurs (parents, enfants) obtenus dans l'étape précédente (extraction de caractéristiques) à un seul vecteur représentatif, et ce, en utilisant certaines méthodes déjà connues et d'autres que nous avons proposées.

Ensuite, nous avons procédé à une opération de sélection des caractéristiques significatives et non redondantes. Cette opération est indispensable, elle nous permet d'éliminer les caractéristiques redondantes et non significatives dans le vecteur représentatif, et ce, en utilisant deux types de méthodes de réduction, à savoir : la projection et la sélection.

Finalement, la classification est réalisée en utilisant trois types de classifieurs, à savoir : la Similarité (Cosin), KNN et SVM.

3.2 Représentation et Normalisation de caractéristiques

3.2.1 Représentation de caractéristiques

Les vecteurs (enfant / parent) obtenus au cours de l'étape précédente (extraction de caractéristiques) doivent être représentés par un seul vecteur caractéristique. Pour atteindre cet objectif, nous avons appliqué les équations ci-dessous (3.1- 3.6) sur les deux vecteurs (Parent et Enfant).

$$F_1 = |F_{Parent} - F_{Enfant}| \quad 3.1$$

$$F_2 = |F_{Parent} + F_{Enfant}| \quad 3.2$$

$$F_3 = F_{Parent} / F_{Enfant} \quad 3.3$$

$$F_4 = F_{Parent} * F_{Enfant} \quad 3.4$$

$$F_5 = \frac{F_{Parent} * F_{Enfant}}{|F_{Parent} - F_{Enfant}|} \quad 3.5$$

$$F_6 = \frac{F_{Parent} + F_{Enfant}}{|F_{Parent} - F_{Enfant}|} \quad 3.6$$

Où F_{Parent} et F_{Enfant} sont respectivement le vecteur caractéristique du parent et le vecteur caractéristique de l'enfant, et F_i est le vecteur caractéristique final. Il est à noter que dans les équations précédentes, les opérations sont effectuées élément par élément.

3.2.2 Normalisation des caractéristiques

La normalisation des caractéristiques est une opération qui consiste à normaliser les valeurs du vecteur représentatif final et les arranger dans un intervalle entre 0 et 1. L'objectif de cette opération est de réduire le temps de calcul et par conséquent, éviter tous les problèmes qui en résultent.

La normalisation est effectuée en utilisant les équations 3.7 et 3.8

$$F_{Norm\ 1} = \frac{F - F_{Min}}{F_{Max} - F_{Min}} \quad 3.7$$

$$F_{Norm\ 2} = \frac{F}{\sqrt{\sum_{j=1}^N F_j^2}} \quad 3.8$$

Où, : $F_{Norm\ i}$ le vecteur caractéristique normalisé, F le vecteur caractéristique et F_j est un élément de vecteur avec l'indice j . F_{Max} et F_{Min} sont respectivement la plus grande et la plus petite caractéristique dans le vecteur F .

3.3 Réduction de caractéristiques

Le nombre important de caractéristiques obtenues lors de l'étape précédente, est un phénomène qui peut être une source de beaucoup de problèmes liés au temps de calcul et à l'espace mémoire utilisé d'une part, et à l'existence de caractéristiques redondantes et non significatives d'une autre part. Ce phénomène a des effets indésirables sur la nature des résultats obtenus et sur les performances du système proposé, par conséquent, une opération de réduction des caractéristiques est fortement imposée [94].

La réduction des caractéristiques est une opération qui vise à réduire le nombre de caractéristiques obtenues, et ce, en supprimant toutes les caractéristiques redondantes et non significatives contenant dans le vecteur caractéristique et ne gardant que les caractéristiques vraiment utiles pour la prise de la décision de parenté [95]. Elle est effectuée en utilisant les méthodes suivantes :

3.3.1 Projection de caractéristiques

Des techniques de projection de caractéristiques ont été utilisées pour projeter les caractéristiques extraites dans un nouvel espace dans lequel la distance intra-classe est minimisée tout en maximisant la distance inter-classe, pour faciliter la tâche de classification [96].

Dans cette partie, trois algorithmes basés sur l'apparence populaire sont présentés, à savoir l'analyse de composantes principales (Principal Components Analysis- PCA), l'analyse linéaire discriminante (Linear Discriminant Analysis - LDA) et les projections de préservation de la localité (Locality Preservation Projections - LPP).

a. Analyse en Composantes Principales (PCA)

L'Analyse en Composantes Principales permet de définir, à partir d'un jeu de données d'apprentissage, un sous espace permettant de simultanément conserver l'information discriminante et supprimer les informations secondaires (non informatives). Cette méthode consiste à trouver une nouvelle base de l'espace des données dont tous les vecteurs sont orthogonaux entre eux. Le premier de ces vecteurs correspond à la direction de variance maximale des données d'apprentissage. Les autres composantes sont déterminées par la contrainte d'orthogonalité entre les vecteurs tout en respectant une direction de variance maximum. Ainsi au lieu de stocker une matrice d'image de résolution \mathbf{N} , on stocke un vecteur de taille $\mathbf{R} \leq \mathbf{M}$ nombre d'image d'apprentissage. Dans cette approche, la normalisation d'éclairage est toujours indispensable [97].

L'Analyse en Composantes Principales est une technique rapide, simple et populaire dans l'identification de modèle, c'est l'une des meilleures techniques. Ses projections sont optimales pour la reconstruction d'une base de dimension réduite.

Les étapes d'application de cette méthode sont comme suit :

1. Sélectionnez Matrice de données, X^T moyenne nulle.
2. Calculer la moyenne :

$$\psi = \frac{1}{N} \sum_{i=1}^N X_i \quad 3.9$$

3. Soustraire la moyenne de la distribution à partir de l'ensemble de données.

$$X_i = X^T - \psi \quad 3.10$$

4. Calculer la matrice de covariance XX^T :

$$C = \sum_{i=1}^N X_i X_i^T \quad 3.11$$

5. Calculer les valeurs propres et les vecteurs propres V de la matrice de covariance :

$$\lambda_i = C_x V \quad 3.12$$

Où $i = 1 \dots \dots \dots N$

6. Commander les vecteurs propres $V_i = (i = 1 \dots \dots \dots N)$ par leurs valeurs propres correspondantes λ_i , par ordre décroissant.

7. Ne conservez que les vecteurs propres avec les valeurs propres les plus importantes (les composants principaux), $k (k \ll N) X^k = V^k \cdot X$

8. Résoudre pour PCA.

$$\lambda V_{X^T} = C_x V_{X^T} \quad 3.13$$

b. Analyse Discriminante Linéaire (LDA)

L'analyse discriminante linéaire est utilisée pour trouver la combinaison linéaire des caractéristiques qui séparent le mieux les classes d'objet ou d'événement. Les combinaisons résultantes peuvent être employées comme classificateur linéaire, ou généralement dans la réduction de caractéristiques.

L'analyse discriminante linéaire est une technique qui cherche les directions qui sont efficaces pour la discrimination entre les données. Son principal est l'axe de projection qui maximise la séparation entre les deux classes. Il est clair que cette projection est optimale pour la séparation des deux classes par rapport à la projection sur l'axe principal calculé par l'Analyse en Composantes Principales [98].

Le principe d'application de cette méthode se résume comme suit :

1. Soit des classes C qui doivent être classées dans l'espace original.
2. Calculez la moyenne de chaque ensemble de données et la moyenne de l'ensemble des données :

$$\mu_i = \frac{1}{N} \sum_{i=1}^N C \quad 3.14$$

3. Soit N_i le nombre d'échantillons en classe $i, i = 1, 2, 3, \dots, C$
4. N est le nombre total d'échantillons :

$$N = \sum_{i=0}^c N_i \quad 3.15$$

5. Calculer la matrice de dispersion dans la classe

$$S_w = \sum_{i=1}^c \sum_{j=1}^{N_i} (x_i - \mu_i)(x_i - \mu_i)^T \quad 3.16$$

6. La dispersion des classes est la covariance attendue de chacune des classes :

$$S_w = \sum_{i=1}^c (x_i - \mu_i)(x_i - \mu_i)^T \quad 3.17$$

Où $\mu = \frac{1}{c} \sum_{i=1}^c \mu_i$

7. Par conséquent, $J(w)$ est une mesure de la différence entre les moyens de classe (encodés dans la matrice de dispersion entre classes) normalisés par une mesure de la matrice de dispersion de classe interne :

$$J(w) = \frac{|w^T S_B W|}{|w^T S_w W|} \quad 3.18$$

8. Résolution le problème des valeurs propres généralisées :

$$\lambda W = S_w^{-1} S_B W \quad 3.19$$

9. Calculez la projection optimale W_{opt} dont les colonnes sont des vecteurs propres correspondant aux plus grandes valeurs propres qui sont toutes linéairement indépendantes et sont invariantes sous la transformation :

$$W_{opt} = \operatorname{argmax}_W \left(\frac{|w^T S_B W|}{|w^T S_w W|} \right) \quad 3.20$$

c. Localité de Projections Préservé (LPP)

Cette méthode utilise la notion de Laplace pour construire un graphique incorporant les informations de voisinage de l'ensemble de données, et la matrice qui transforme les points de données vers un sous-espace créé. Cette transformation linéaire optimale préserve l'information locale du quartier. La carte de représentation générée par l'algorithme pourrait être considérée

comme une approximation discrète linéaire à une carte continue qui résulte naturellement de la géométrie du collecteur. Les cartes sont conçues pour minimiser un critère objectif différent de la technique linéaire classique [99].

Le principe d'application de cette méthode se résume comme suit :

1. Compte tenu d'un ensemble de données de N échantillons

$$X = \{x_1; x_2; \dots; x_N\} \text{ où } x_i \in \mathfrak{R}^D$$

2. Trouvez la matrice de transformation W de la taille $D \times d$.

3. Carte $y_i = WT x_i$ où $y_i \in \mathfrak{R}^d$

4. Construire le graphique d'adjacence soit par :

- ε – Quartier (Neighborhood)
- K – Voisins les plus proches (K-nearest neighbors)
- Trouver la matrice de similarité S (utilisant le noyau de chaleur)

$$S_{ij} = \begin{cases} \frac{\exp\|x_i - x_j\|}{t} & i, j \text{ liés} \\ 1 & \text{Sinon} \end{cases} \quad 3.21$$

5. Calculer un problème de valeurs propres généralisées

$$XLX^T.W = \lambda DX^T W \quad 3.22$$

Où D : matrice diagonale $D_{ij} = \sum_j S_{ij}$ et $L = D - S$

3.3.2 Sélection des caractéristiques

Les techniques de sélection ont été introduites pour réduire la taille des vecteurs caractéristiques tout en maintenant leurs données. Elles visent à améliorer les résultats obtenus et les performances de la classification. Parmi les méthodes de sélection de caractéristiques qu'on peut trouver dans la littérature on trouve :

a. Score de ReliefF

Une des méthodes de filtrage les plus connues pour la sélection des caractéristiques est la méthode reliefF. Cette méthode fut proposée en 1992 par Kira et Rendell [100].

Son principe est de calculer une mesure globale de la pertinence des caractéristiques en accumulant la différence des distances entre des exemples d'apprentissage choisis aléatoirement et leurs plus proches voisins de la même classe et de l'autre classe.

La simplicité, la facilité de la mise en œuvre ainsi que la précision même sur des données bruitées, représentent les avantages de cette méthode. En revanche, sa technique aléatoire ne peut pas garantir la cohérence des résultats lorsqu'on applique plusieurs fois la méthode sur les mêmes données. Par ailleurs, cette méthode ne prend pas en compte la corrélation éventuelle entre les caractéristiques.

b. Score de Fisher

Le score de Fisher est très utilisé pour la sélection supervisée de caractéristiques en raison de sa bonne performance. C'est une méthode pour déterminer les caractéristiques les plus pertinentes pour la sélection. Il utilise des méthodes discriminatives et des modèles statistiques génératifs pour y parvenir [101].

Le score de Fisher évalue chaque caractéristique séparément en mesurant le degré de séparabilité de la classe en question comme suit (Equation 3.9) :

$$F(i) = \frac{\sum_{c=1}^C \eta_c (\mu_c^i - \mu^i)^2}{\sum_{c=1}^C \eta_c (\sigma_c^i)^2} \quad 3.23$$

Avec :

- η_c : le nombre d'éléments appartenant à la classe c ;
- μ_c^i : la moyenne de la caractéristique i dans la classe c ;
- σ_c^i : la déviation standard de la caractéristique i dans la classe c ;
- μ^i : la moyenne globale de la caractéristique i.

c. Score Z

Le score z (La note z ou encore score standard) correspond à l'expression d'un écart à la moyenne exprimé en fraction d'écart-type. Pour une distribution de notes de moyenne m et d'écart-type σ , la note z correspondant au score x se calcule facilement en utilisant l'équation suivante :

$$Z_x = \frac{x - m}{\sigma} \quad 3.24$$

Comme pour tous les scores étalonnés les notes z ont du sens contrairement à un score brut. Il exprime un score par rapport à la moyenne des scores d'un groupe dans une unité (fraction d'écart-type) comparable quel que soit la mesure. Il faut être prudent cependant : la distribution des notes brutes doit être normale ou a minima symétrique et unimodale pour que ces comparaisons aient du sens [102].

d. Score T-test :

Ce schéma de sélection de caractéristiques est donné par le T-test ([103], [104]). Ses poids sont basés sur le T-test à deux échantillons de valeur absolue avec une estimation de variance groupée. Le poids est donné par la formule suivante :

$$W_{T-test} = \frac{(m_k - m_n)}{\sqrt{\frac{\sigma_k^2}{N_k} + \frac{\sigma_n^2}{N_n}}} \quad 3.25$$

Ou W_{T-test} est le score de la caractéristique F_i , N_i est le nombre d'échantillons dans la classe de parenté i ($1 \rightarrow \text{kin} / 2 \rightarrow \text{non-kin}$), m_i et σ_i sont la moyenne et la variance de la classe de parenté dans la caractéristique prévue.

e. Score Kullback-Leibler (KL) :

En théorie des probabilités et en théorie de l'information, la divergence de Kullback-Leibler (K-L) est une mesure de dissimilarité entre deux distributions de probabilités. Elle doit son nom à Solomon Kullback et Richard Leibler ([104], [105]).

Ce schéma de sélection de caractéristiques est donné par le Kullback-Leibler KL. Ses poids sont basés sur la divergence et le mesure de dissimilarité entre deux distributions. Le poids est donné par la formule suivante :

$$W_{KL} = \frac{\frac{\sigma_k^2}{\sigma_n^2} + \frac{\sigma_n^2}{\sigma_k^2} - 2}{2} + (m_k - m_n)^2 \frac{\frac{1}{\sigma_n^2} + \frac{1}{\sigma_k^2}}{2} \quad 3.26$$

Ou W_{KL} est le score de la caractéristique F_i , m_i et σ_i sont la moyenne et la variance de la classe de parenté i ($1 \rightarrow \text{kin} / 2 \rightarrow \text{non-kin}$).

3.4 Classification

La classification permet de partitionner un ensemble d'observations en différentes classes, en regroupant dans une même classe les observations présentant les mêmes caractéristiques. On parle de classification des images lorsque les observations correspondent à des images et où chaque image peut être caractérisée par un ensemble d'attributs tels que les attributs de texture ou couleur [106].

La classification des images intervient comme une étape très importante dans des applications de reconnaissance des formes telles que l'identification des images, la reconnaissance des visages, de la parole, des empreintes, etc.....

Les méthodes de classification peuvent être divisées en deux catégories selon l'information disponible à priori sur les observations. Lorsque les différentes classes des observations sont connues au préalable, la classification est dite supervisée. Cette catégorie de méthodes est préférable pour la classification des images. En revanche, dans le cas où l'on n'a aucune connaissance a priori sur l'appartenance des observations aux classes, on parle de classification non-supervisée. Dans ce cas, ces observations sont appelées prototypes de la classe [107].

La classification a pour but de définir à partir des prototypes des modèles mathématiques pour chaque classe. Ces modèles peuvent être un simple centre de gravité, une fonction de densité de probabilité ou une surface de séparation entre des classes. Les étapes de caractérisation et de classification des observations prototypes des classes constituent ce qu'on appelle la phase d'apprentissage. Celle-ci se distingue de la phase de décision qui consiste à affecter une nouvelle observation à l'une des classes définies lors de la phase d'apprentissage. Il existe une multitude de méthodes de classification supervisée. Les plus significatives sont (la similarité en Cosinus, le k plus proches voisins (K -ppv) et la Machine à vecteurs de support (SVM).

3.4.1 Similarité en Cosinus [108]

L'idée générale est d'apprendre une matrice de transformation à partir des données d'apprentissage afin que la similarité en Cosinus fonctionne bien dans le sous-espace transformé.

La similarité cosinus est le produit scalaire de deux vecteurs représentant la requête et les descripteurs des images (Parents, Enfants).

La similarité en cosinus est une mesure de la similitude entre deux vecteurs (Parent, Enfant) non nuls. Elle est représentée par l'équation suivante :

$$\cos \theta = \frac{\vec{v}_p \cdot \vec{v}_c}{\|\vec{v}_p\| \|\vec{v}_c\|} \quad 3.27$$

Où (v_p, v_c) représentent les vecteurs de caractéristiques du parent / enfant.

Nous avons utilisé cette métrique pour voir dans quelle mesure la représentation des vecteurs (Parent, Enfant) est de chaque paire par chaque caractéristique. Sa valeur tend vers 1 lorsque v_p et v_c sont similaires et 0 lorsqu'ils ne sont pas similaires.

Les meilleures caractéristiques devraient être celles qui ont une valeur proche de 1 lorsque les vecteurs de la paire appartiennent à la classe positive et proches de 0 s'ils appartiennent à la classe négative.

3.4.2 Méthode des k plus proches voisins

La méthode des k plus proches voisins (*Knn k-nearest neighbor*). (Indyk et Motwani [109]) se base sur une comparaison directe entre le vecteur caractéristique représentant l'entité à classer et les vecteurs caractéristiques représentant des entités de référence.

La comparaison consiste en un calcul de distances entre ces entités. L'entité à classer est assignée à la classe majoritaire parmi les classes des k entités les plus proches au sens de la distance utilisée.

Lors de l'initialisation, on doit spécifier au programme le nombre de classes k que nous voulons obtenir de même qu'un ensemble d'apprentissages de départ. Cet ensemble est composé d'un nombre prédéterminé de vecteurs choisis au hasard ou par l'utilisateur parmi les vecteurs devant être classés. Cette étape complétée, chaque (vecteur i) de l'ensemble est alors présenté au classifieur qui l'associe à une classe spécifique (classe j). (Voir **Figure 3.1**).

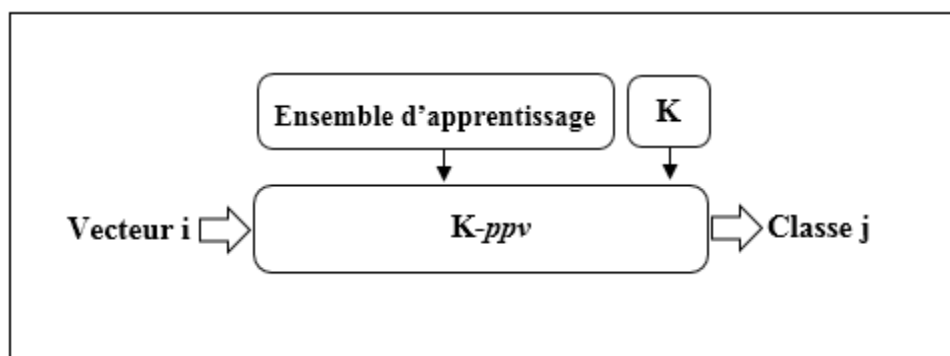


Figure 3.1 : Représentation des paramètres nécessaires à la méthode K -ppv

a. Procédure de K -ppv [110]

Pour expliquer la procédure de K -ppv, prenons l'exemple suivant : classifier un vecteur X dans un système possédant deux classes distinctes A, B avec un k égal à trois. À l'ajout du vecteur, K -ppv suivra la démarche suivante :

1. Calcul de la distance entre le vecteur X et tous les vecteurs déjà classifiés.
2. Sélection des k vecteurs (trois dans le cas qui nous concerne) dont la distance entre le vecteur à classifier et les vecteurs du classifieur est la plus petite.
3. Association du nouveau vecteur à la classe possédant le plus grand nombre de voisins se trouvant à proximité. Exemple : Pour un nouveau vecteur X , si nous avons deux classes comportant chacune trois membres soit $A(A_1, A_2, A_3)$ et $B(B_1, B_2, B_3)$ et si les vecteurs les moins distants sont le vecteur A_1 , le vecteur A_2 et le vecteur B_3 , alors X sera associé à la classe A .

b. Particularité de K -ppv [110]

Plusieurs facteurs influent sur les résultats probants de cet algorithme de classification. Nous observons les éléments suivants :

- **Le nombre K**

Si nous prenons un k égal à trois, nous aurons alors une comparaison avec ses 3 plus proches voisins. Il est conseillé d'avoir un nombre impair afin qu'il y ait toujours une classe prédominante lors du traitement. Plus le nombre k est élevé, plus le nombre de comparaisons est grand. Si aucune classe ne prédomine, celle dont les distances entre le vecteur à classer et les vecteurs la composant sont les plus petites est choisie.

- **Le choix des vecteurs des éléments initiaux.**

Les vecteurs initiaux doivent être choisis de manière à représenter l'ensemble des données et non au hasard. Choisir des vecteurs initiaux trop rapprochés ou trop éloignés aura pour effet d'associer les vecteurs à classifier une classe en particulier plutôt que de les répartir sur l'ensemble des vecteurs servant de point central aux classes.

- **L'ordre dans lequel les vecteurs sont présentés à l'algorithme K -ppv**

Une trop faible convergence entraînera une plus grande difformité des classes. Si les vecteurs présentés en entrée sont trop rapprochés les uns des autres, tous les vecteurs s'agglutineront à la même classe, celle-ci devenant la classe ayant le plus de voisins respectifs. Nous notons deux aspects importants de l'algorithme K -ppv: d'une part, à chaque nouvelle classification il est nécessaire de parcourir l'ensemble de la base d'apprentissage, ce qui en fait un algorithme qui n'est pas nécessairement très efficace (surtout que, habituellement, on cherche à avoir la base d'apprentissage la plus grande possible afin d'avoir un meilleur classifieur), et d'autre part, un point crucial de cet

algorithmes est la fonction de distance utilisée pour mesurer la proximité des objets. Il n'existe pas de distance/similarité universellement optimale et une bonne connaissance du problème traité guide généralement le choix de cette distance/similarité.

c. Calcul de distance [111]

Pour mesurer la distance entre deux objets, une notion de distance, dépendante du type de données utilisées, est nécessaire. Plusieurs méthodes de calcul ont été développées (et continuent encore à être à l'étude) afin de calculer la distance entre deux éléments à classer.

Mathématiquement, une distance est une application qui formalise l'idée de distance qui est la longueur entre deux points. La distance permet de regrouper un groupe d'individus similaire et séparer les individus qui ne présentent aucune similarité.

La fonction de distance d entre deux points x_i et x_j est une application de $(\mathbb{R}^d \times \mathbb{R}^d)$ dans \mathbb{R}^+ satisfaisant les propriétés suivantes :

- Non négativité : $d(x_i, x_j) \geq 0$
- Symétrie : $d(x_i, x_j) = d(x_j, x_i)$
- Séparation : $d(x_i, x_j) = 0 \Rightarrow x_i = x_j$
- Minimalité : $d(x_i, x_i) = 0$
- Intégralité triangulaire : $d(x_i, x_j) \leq d(x_i, x_k) + d(x_k, x_j)$

Plusieurs fonctions de distance ont été définies dans la littérature. Ces fonctions ont une valeur proche de 0 si un couple de points est très proche et $+\infty$ si le couple de points est très éloigné.

Les distances les plus utilisées sont :

- **Distance Euclidienne [112]**

Pour deux vecteurs x_i^t et x_j^t la distance euclidienne est définie comme suit (équation 3.28) :

$$d(x_i^t, x_j^t) = \sqrt{\sum_{t=1}^n (x_i^t - x_j^t)^2} \quad 3.28$$

Elle permet de définir la distance entre deux vecteurs représentés comme deux coordonnées sur le plan cartésien. La distance représente le calcul de l'hypoténuse d'un triangle rectangle dont la longueur des côtés correspond à la projection de la distance de chaque dimension des vecteurs sur leurs axes respectifs. Si nous avons deux vecteurs VI

et V_2 à deux dimensions soit $V_1 = (A_1, B_1)$ et $V_2 = (A_2, B_2)$, la longueur de chacun des côtés correspond donc à la valeur absolue de la différence des composantes de chaque dimension.

- **Distance Cosinus [113]**

La distance cosinus est décrite comme suit (équation 3.29) :

$$d(x_i^t, x_j^t) = \frac{\sum_{t=1}^n x_i^t \times x_j^t}{\sqrt{\sum_{t=1}^n (x_i^t)^2} \times \sqrt{\sum_{t=1}^n (x_j^t)^2}} \quad 3.29$$

- **Corrélation [114]**

La fonction de corrélation de deux variables x_i^t et x_j^t est donnée par l'équation 3.30) :

$$d(x_i^t, x_j^t) = 1 - c_{x_i^t, x_j^t} \quad 3.30$$

Où c est le coefficient de corrélation de Pearson.

3.4.3 Machine à vecteurs de support (SVM)

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 [115].

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.

a. 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 [116].

b. 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 l'*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 [117].

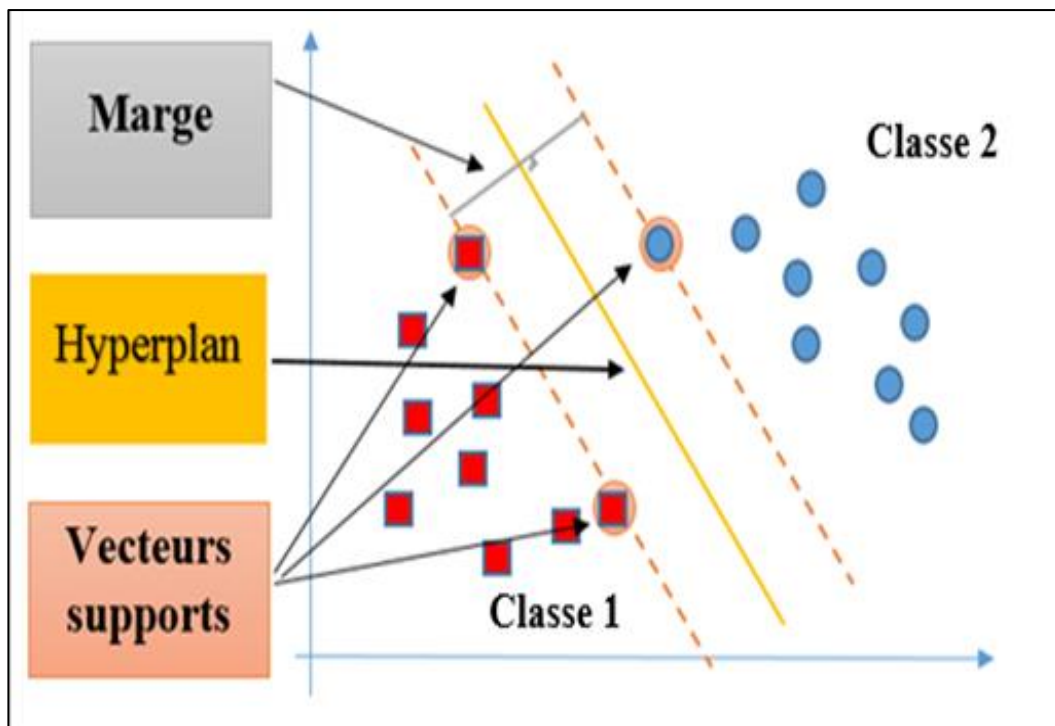


Figure 3.2 : Exemple d'hyperplan optimal, vecteurs de support et marge maximale.

c. La classification linéaire [118].

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 : $w \cdot x - b = 0$

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 :

$$w \cdot x - b = 1 \quad \text{et} \quad w \cdot x - b = -1.$$

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 :

$$w \cdot x_i - b \geq 1 \quad \text{ou} \quad w \cdot x_i - b \leq -1, \quad \text{ou sous une autre équation} \quad c_i (w \cdot x_i - b) \geq 1 \quad \text{avec} \quad 1 \leq i \leq p.$$

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 :

$$\varepsilon = \begin{cases} 1 & \text{si } w \cdot x + b \geq 0 \\ -1 & \text{si } w \cdot x + b \leq 0 \end{cases} \quad \mathbf{3.31}$$

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)

f. 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.

3.5 Validation croisée

La validation croisée est une procédure qui structure les itérations nécessaires à l'évaluation des performances du classifieur. Avant toute chose, il faut déterminer le nombre de réplifications désirées. Typiquement, la validation croisée est effectuée sur dix itérations, on parle alors d'une validation croisée à dix partitions. Dans un premier temps, la base de données doit être séparée en dix partitions disjointes stratifiées d'une taille approximativement égale. À la première itération, les neuf premières partitions sont utilisées pour l'apprentissage du classifieur, et la dernière est utilisée pour les tests. Ensuite, les partitions 2 à 10 sont utilisées pour l'apprentissage et la partition 1 est utilisée pour les tests [119].

À la troisième itération, les partitions 3 à 10 et la partition 1 sont utilisées pour l'apprentissage, et la partition 2 est utilisée pour les tests, ainsi de suite. **La Figure 3.3** présente le principe de la validation croisée sur 10 partitions.

Ainsi, l'algorithme de classification est entraîné dix fois sur autant de segments de données différents et dix tests indépendants sont conduits par la suite. Le nombre de partitions (itérations) peut être modifié pour répondre aux besoins particuliers d'une expérimentation.

Witten et Frank [120] mentionnent qu'une validation à 5 ou 20 partitions est probablement un choix aussi valide qu'un nombre fixé à 10.

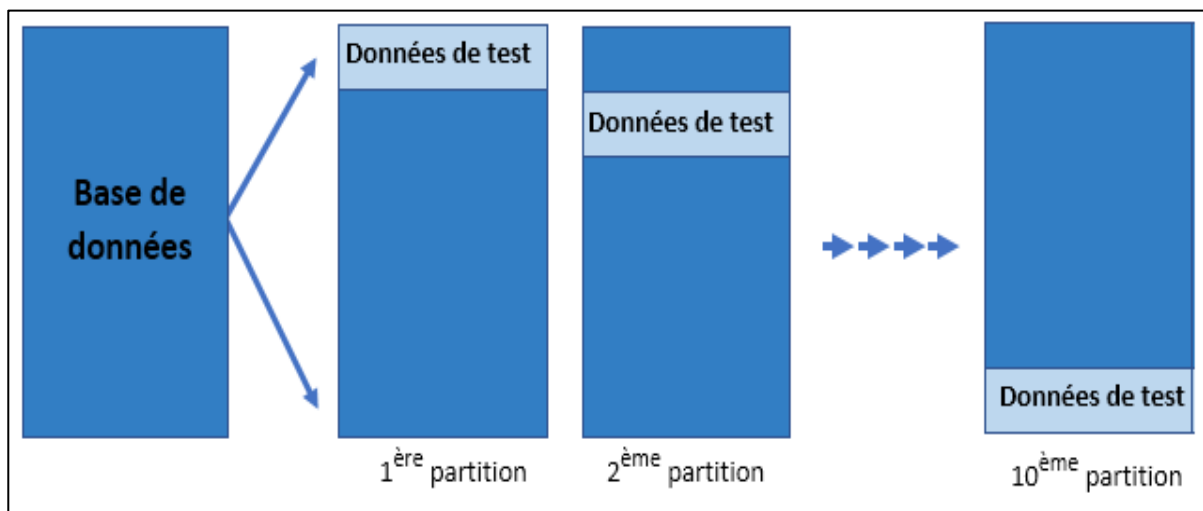


Figure 3.3 : Principe de la validation croisée

3.6 Performance d'un système biométrique

Pour évaluer les performances d'un système biométrique, il y a plusieurs paramètres qui expriment le taux de réussite, nous présentons les paramètres les plus significatives sur ce côté :

3.6.1 Précision d'un système biométrique (Pr)

Il est parmi les paramètres les plus utilisés, dans notre travail on calcule la précision de vérification de parenté à partir de nombre de test correspondant à une parenté correcte par rapport de nombre total de test.

L'erreur commise par ce type de système est d'attribuer à l'enfant présenté un autre parent que la sienne. Les performances de ces systèmes sont mesurées à l'aide du taux d'identification correcte ou la précision du système.

$$P_r = \frac{\text{Nombre de test correspondant a une parenté correcte}}{\text{Le nombre total de test}} \quad 3.32$$

3.6.2 Caractéristiques de fonctionnement du récepteur (Receiver Operating Characteristic ROC)

Un système de vérification ne peut donner, lors d'une comparaison entre deux échantillons qu'un résultat sous forme de probabilité de coïncidence ou score (S), ce score est comparé à un seuil de décision (T) que l'on détermine grâce à de nombreux tests ; Si S est supérieur à T le sujet en question est accepté et dans le cas contraire le sujet est considéré comme imposteur. Dans ce cas, deux types d'erreurs peuvent être commis (FR et FA), et d'autre part la fiabilité d'un système biométrique est caractérisée par deux valeurs statistiques (TFR, TFA) [121].

- a. **Faux Rejets FR (False-Rejection)** : correspond au cas où le système rejette un client légitime.
- b. **Fausse Acceptation FA (False-Acceptance)** : correspond au cas où le système accepte un imposteur.
- c. **Taux de Faux Rejets (TFR) ou False Reject Rate (FRR)**: il exprime le pourcentage d'utilisateurs légitimes rejetés.

$$TFR = \frac{\text{Nombre de FR}}{\text{Le nombre de clients}} \quad 3.33$$

- d. **Taux de Fausses Acceptations (TFA) ou False Accept Rate (FAR)**: il exprime le pourcentage d'imposteurs acceptés.

$$TFA = \frac{\text{Nombre de FA}}{\text{Le nombre d'imposteurs}} \quad 3.34$$

Comme la **Figure 3.4** illustre, on a toujours une zone de recouvrement. On aimerait évidemment avoir les deux distributions parfaitement disjointes, ce qui permettrait idéalement de séparer les clients des imposteurs, mais ce n'est jamais le cas dans la réalité.

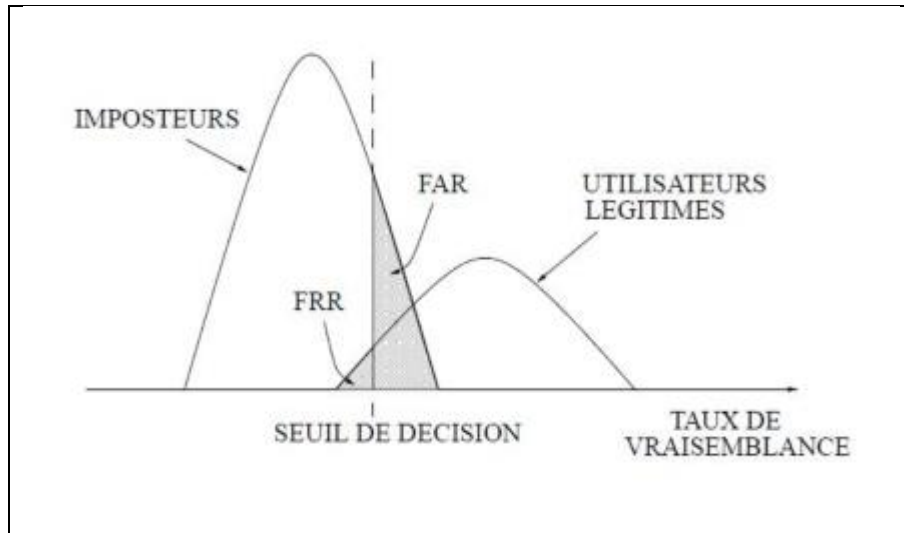


Figure 3.4 : La distribution du taux de vraisemblances.

On peut représenter graphiquement les performances d'un système de reconnaissance de visage à l'aide de la courbe ROC (Receiver Operating Characteristic) [122]. Les FRR sont données en fonction des FAR pour les différentes valeurs du seuil T, (voir **Figure 3.5**)

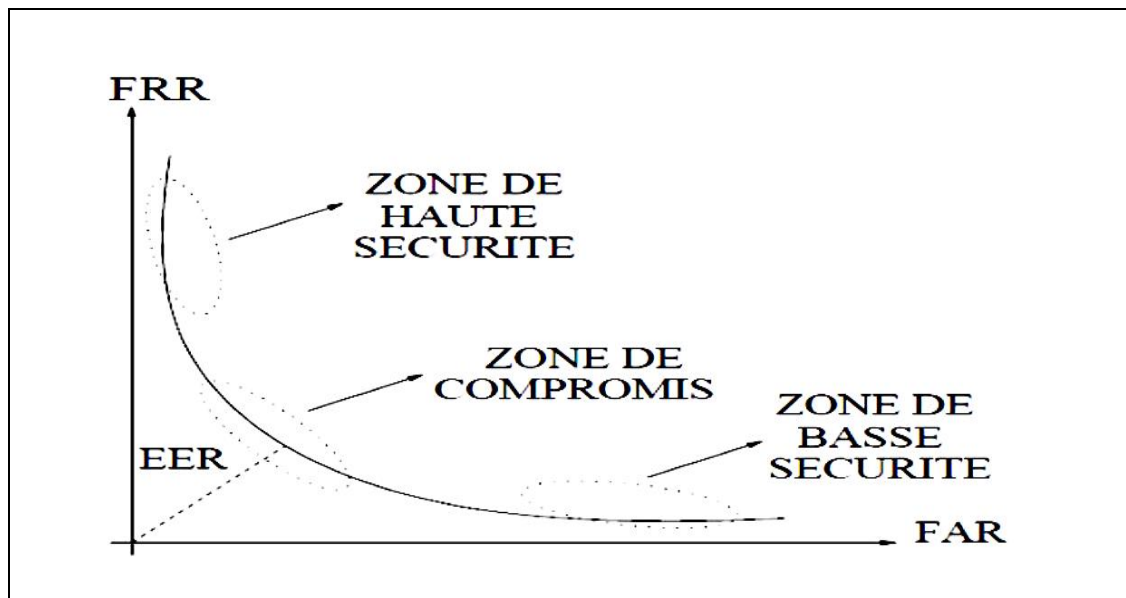


Figure 3.5 : La courbe de Caractéristiques de fonctionnement du récepteur (ROC).

Conclusion

La décision finale est la dernière étape dans le système de vérification de parenté que nous proposons, au cours de laquelle, le système fournit une décision d'existence ou d'inexistence de parenté entre deux images faciales présentées à son entrée.

Au cours de cette étape, le système commence par transformer les paires de vecteurs (parents, enfants) obtenus dans l'étape précédente (extraction de caractéristiques) à un seul vecteur représentatif composé d'un nombre important de caractéristiques, ces dernières ne sont pas dans la majorité des cas significatives, d'où la nécessité de réduire la taille du vecteur représentatif, et ce, en éliminant les caractéristiques redondantes et non significatives. Les méthodes de réduction des caractéristiques, particulièrement la projection et la sélection sont présentées dans ce chapitre. La classification est la dernière opération effectuée par le système permettant d'avoir une décision finale relative à l'existence ou l'inexistence d'une parenté entre deux images faciales. Les principales méthodes de classification sont exposées dans ce chapitre.

Chapitre 4

Résultats et analyse

Sommaire

| | |
|---|-----------|
| 4.1 Introduction | 67 |
| 4.2 Approche Proposé | 67 |
| 4.3 Effet de descripteurs | 69 |
| 4.3.1 Descripteurs de texture | 69 |
| 4.3.2 Descripteurs d'apprentissage en profond | 70 |
| 4.3.3 Fusion des descripteurs | 71 |
| 4.4 Effet de la représentation du visage (Faciale) | 73 |
| 4.5 Effet de la représentation de Caractéristiques | 75 |
| 4.6 Effet de la Réduction des caractéristiques (Projection et Sélection) | 76 |
| 4.6.1 Sélection de caractéristiques | 76 |
| 4.6.2 projections de caractéristiques | 78 |
| 4.6.3 Combinaison entre la sélection et la projection de caractéristiques..... | 79 |
| 4.7 Effet des Classifieurs | 79 |
| 4.8 Temps de calcul (CPU et Nombre des caractéristiques) | 80 |
| 4.9 Performances obtenues pour les différentes relations de parenté | 82 |
| 4.10 Courbes caractéristiques de fonctionnement du récepteur (ROC) | 83 |
| 4.11 Comparaison des méthodes existantes avec Notre Méthode..... | 84 |
| 4.12 Conclusion | 86 |

4.1 Introduction

De toutes les applications biométriques, la vérification automatique de la parenté reste l'un des sujets de recherche les plus actifs.

Jusqu'ici nous avons étudié le sujet de la vérification de la parenté par des images faciales d'un point de vue théorique, or, ce sujet est un problème pratique où les performances de tout système proposé dépendent d'une manière directe ou indirecte de beaucoup de paramètres relatifs à chaque étape de traitement. Par conséquent, nous sommes persuadé que l'étude et l'analyse des différentes alternatives de solutions appliquées sur des bases de données universelles peuvent apporter beaucoup de solutions à des problèmes qui persistent encore et par conséquent, améliorer les performances des approches déjà proposées dans ce domaine.

Ce chapitre est consacré à la présentation de notre approche ainsi que l'analyse et l'interprétation des résultats obtenus. Tout d'abord, nous avons étudié l'influence de la combinaison des descripteurs utilisés pour l'extraction des caractéristiques, en particulier les descripteurs de texture et les descripteurs d'apprentissage en profond, et ceux utilisant plusieurs représentations faciales (MB, ML et PML). De plus, nous avons quantifié les performances en utilisant différentes méthodes de fusion de paires ainsi que différentes méthodes de sélection des caractéristiques. Finalement, dans la dernière étape du système proposé, l'étude de l'influence du module de classification a été également abordée.

Notons toutes les méthodes que nous avons proposées ont été appliquées sur des bases de données universelles et accessibles au public (Cornell KinFace, UB KinFace, Family 101, KinFaceW-I et KinFace-II). L'utilisation de ce type de bases de données donne aux résultats obtenus plus d'objectivité et plus de crédibilité.

4.2 Approche Proposée

L'approche proposée pour la vérification de la parenté consiste à utiliser les visages de deux personnes pour déterminer s'il existe une relation de parenté entre eux. Elle comprend six étapes:

1. Prétraitement de l'image du visage : pour détecter et cadrer le visage à partir de l'image faciale présentée à l'entrée du système.
2. Extraction des caractéristiques: en utilisant les deux types de descripteurs que nous avons présentés dans le deuxième chapitre, à savoir, les descripteurs de texture (LBP, LDP, LTP, LPQ et BSIF), et les descripteurs d'apprentissage en profond (VGG-16, VGG-Face, ResNet).

3. Opter pour une représentation de l'image faciale en se basant sur les méthodes de représentation faciale que nous avons présentées dans le deuxième chapitre, à savoir, la représentation multi bloc (MB), la représentation multi niveaux (ML) et la représentation pyramide multi niveaux (PML).
4. Représentation et normalisation des caractéristiques.
5. Réduction des caractéristiques, et ce, en utilisant la méthode de la sélection des caractéristiques (score de Fisher, T-test, Kullback – Leibler (KL)) ainsi que la méthode de la projection des caractéristiques (l'analyse de composantes principales PCA-, les projections de préservation de la localité –LPP-, l'analyse linéaire discriminante LDA).
6. Décision de la vérification de parenté en utilisant les classifieurs que nous avons présentés dans le chapitre précédent (similarité en Cosinus, méthode des k plus proches voisins (KNN), machine à vecteurs de support (SVM)). **La Figure 4.1** illustre la structure générale de l'approche proposée.

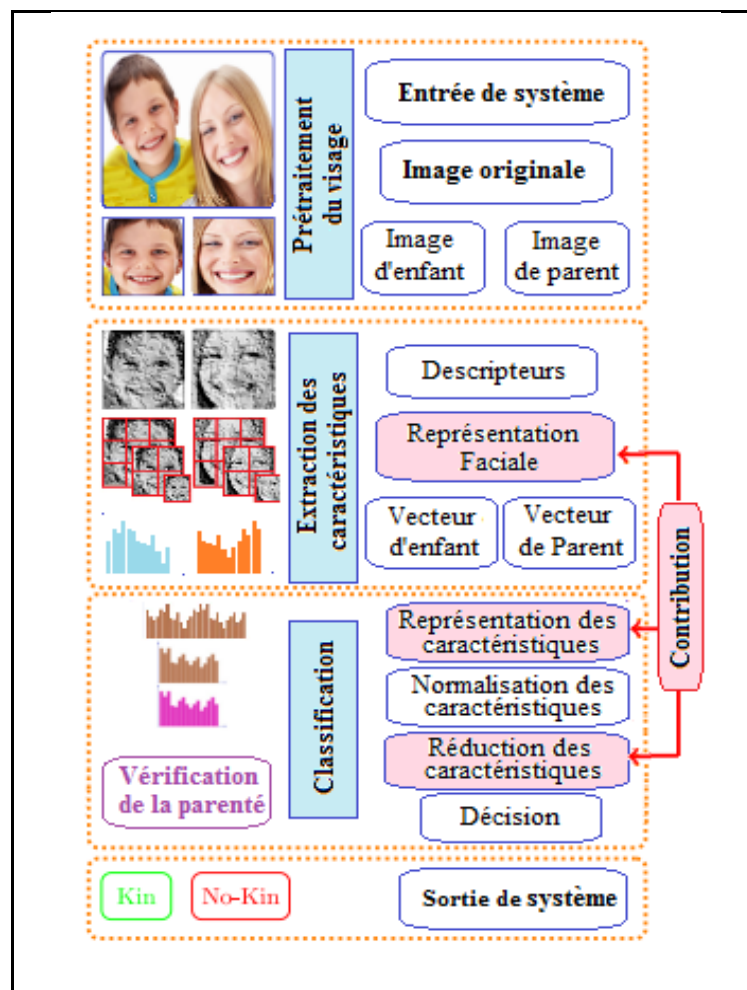


Figure 4.1 : Structure générale de l'approche proposée.

Dans ce travail, nous avons déjà défini ce que l'on entend par les descripteurs d'image, la représentation du visage, la fusion de paires, la réduction des caractéristiques et la classification. Pour chaque facteur, un certain nombre de schémas étaient décrits dans la thèse.

Nous avons mené plusieurs expériences visant à trouver la meilleure combinaison de facteurs pouvant fournir les meilleures performances.

À cette fin, nous avons procédé séquentiellement à l'étude de l'effet de ces facteurs. Nous recherchons la meilleure configuration d'un facteur donné en fixant la configuration des autres.

En procédant de manière séquentielle, nous présentons les résultats expérimentaux dans les sections suivantes.

4.3 Effet des descripteurs

En ce qui concerne l'étape d'extraction des caractéristiques, nous avons étudié l'efficacité de la méthode proposée avec différents descripteurs de caractéristiques (Texture et apprentissage en profondeur).

4.3.1 Descripteurs de texture

Dans cette expérience nous avons utilisé les descripteurs de texture les plus connus dans le traitement des images faciales (LBP, LTP, LDP, LPQ, WLD, BSIF) pour déterminer le plus efficace dans l'étape d'extraction des caractéristiques.

Pour les autres facteurs, Nous avons utilisé la configuration suivante : PML pour la représentation faciale, F_1 pour la représentation de caractéristiques, F_2 pour la normalisation, Le Score pour la sélection des caractéristiques et SVM pour la classification.

$$F_1 = \frac{F_{Parent} + F_{Enfant}}{|F_{Parent} - F_{Enfant}|} \quad 4.1$$

$$F_2 = \frac{F - F_{Min}}{F_{Max} - F_{Min}} \quad 4.2$$

Le **Tableau 4.1** présente les résultats obtenus pour chaque descripteur sur les bases de données utilisées.

Tableau 4.1 : Résultats obtenus pour chaque descripteur sur les bases de données utilisées.

| Bases de données | LDP | LTP | LBP | WLD | BSIF | LPQ |
|---------------------|---------|---------|---------|---------|---------|----------------|
| Cornell Kin | 56.10 % | 64.34 % | 88.46 % | 89.32 % | 94.41 % | 95.21 % |
| UB KinFace | 57.88 % | 73.38 % | 90.50 % | 91.86 % | 93.13 % | 94.46 % |
| Family 101 | 60.63 % | 66.94 % | 90.15 % | 91.02 % | 92.21 % | 93.20 % |
| KinFace W-I | 59.16 % | 64.45 % | 87.30 % | 88.81 % | 89.15 % | 91.40 % |
| KinFace W-II | 57.47 % | 61.41 % | 76.24 % | 82.61 % | 84.67 % | 90.38 % |

A partir de ces résultats obtenus, nous pouvons observer que le descripteur LPQ atteint la plus grande précision. Il a été suivi du descripteur BSIF et WLD. Les descripteurs LTP et LDP ont obtenu une faible précision par rapport aux premiers descripteurs (LPQ, BSIF et WLD). Le descripteur LBP a atteint une bonne précision, mais sa précision était encore inférieure à celle des descripteurs LPQ et BSIF et bien meilleure que celle des descripteurs LTP et LDP. Le LPQ était à l'origine lié à l'analyse spectrale du contenu de l'image.

4.3.2 Descripteurs d'apprentissage en profond

Dans cette expérience nous avons utilisé les descripteurs d'apprentissage en profond les plus utilisés dans le traitement des images faciales (VGG-16, VGG-Face) pour déterminer leurs performances dans l'étape d'extraction de caractéristiques.

Pour les autres facteurs, nous avons utilisé la configuration suivante : F_1 pour la représentation des caractéristiques, F_2 pour la normalisation, Score de Fisher pour la sélection des caractéristiques, SVM pour la classification.

$$F_1 = \frac{F_{Parent} + F_{Enfant}}{(F_{Parent} - F_{Enfant})^2} \quad 4.3$$

$$F_2 = \frac{F}{\sqrt{\sum_{j=1}^N F(j)^2}} \quad 4.4$$

Nous avons choisi deux modèles de réseaux de neurones convolutifs (VGG-16, VGG-Face) qui ont la même architecture, mais chaque modèle a été formé par une base de données spéciale.

VGG-16 est un modèle CNN, il a été formé par la base de données ImageNet (c'est une base de données contenant différents types d'images).

VGG-Face est un modèle CNN calculé à l'aide de VGG-16 CNN, il a la même architecture et a été formé par la base de données VGG-Face (c'est une base de données contenant uniquement des images faciales).

Le **Tableau 4.2** présente les Résultats obtenus pour les descripteurs d'apprentissage en profond sur les bases de données utilisées.

Tableau 4.2 : Résultats obtenus pour les descripteurs d'apprentissage en profondeur.

| Bases de données | VGG-16 | VGG-Face |
|---------------------|---------|----------------|
| Cornell Kin | 87.12 % | 92.89 % |
| UB KinFace | 86.35 % | 90.59 % |
| Family 101 | 81.65 % | 84.82 % |
| KinFace W-I | 82.64 % | 86.65 % |
| KinFace W-II | 76.41 % | 81.11 % |

D'après le **Tableau 4.2**, nous avons remarqué que les résultats utilisant VGG-Face étaient meilleurs que VGG-16. Cela confirme l'effet de l'apprentissage des modèles de CNN à travers des bases de données spécifiques. Dans notre expérience, le modèle CNN formé par la base de données VGG-Face est spécialisé dans les images faciales, ce qui a donné un bon résultat à leur utilisation par rapport au VGG-16.

4.3.3 Fusion des descripteurs

Dans cette expérience nous avons choisis deux types de descripteurs (Texture et apprentissage en profond) pour étudier l'influence de la fusion des descripteurs et l'effet d'augmentation du nombre des caractéristiques.

a. Fusion des descripteurs de texture

Dans cette partie, nous avons mené de nombreuses expériences en utilisant un ensemble de descripteurs de texture (LBP, BSIF, LPQ), avec des représentations de visage MB. Nous avons utilisé en plus F_1 pour la représentation des caractéristiques, F_2 pour la normalisation, Score de Fisher pour la sélection des caractéristiques et SVM pour la classification.

$$F_1 = |F_{Parent} - F_{Enfant}| \quad 4.5$$

Nous avons d'abord testé notre approche en utilisant un seul descripteur, puis nous avons fusionné deux descripteurs puis nous avons fusionné tous les descripteurs.

Le **Tableau 4.3** présente les performances du système avec les descripteurs LBP, BSIF et LPQ et la fusion entre eux.

Tableau 4.3 : Performances du système avec les descripteurs LBP, BSIF et LPQ et la fusion entre eux.

| Descripteurs | Cornell Kin | UB Kinface | Family101 | KinFaceW-I | KinFaceW-II |
|--------------------------|----------------|----------------|----------------|----------------|----------------|
| LBP*MB | 68.21 % | 67.36 % | 65.32 % | 62.52 % | 61.96 % |
| BSIF*MB | 75.69 % | 72.17 % | 71.03 % | 68.35 % | 65.51 % |
| LPQ*MB | 78.56 % | 74.87 % | 73.24 % | 72.10 % | 69.35 % |
| (LBP+BSIF)*MB | 79.13 % | 75.54 % | 76.66 % | 75.36 % | 72.98 % |
| (LBP+LPQ)*MB | 80.78 % | 78.14 % | 77.89 % | 77.74 % | 75.63 % |
| (BSIF+LPQ)*MB | 82.68 % | 81.03 % | 79.25 % | 78.32 % | 76.36 % |
| (LBP+BSIF+LPQ)*MB | 84.74 % | 82.89 % | 81.69 % | 80.12 % | 78.16 % |

Le **Tableau 4.3** montre que l'utilisation d'un seul descripteur LBP, BSIF ou LPQ donne des résultats inférieurs à l'utilisation des descripteurs fusionnées (LBP + BSIF), (LBP + LPQ) ou (BSIF + LPQ). Notre approche donne de meilleurs résultats lors de l'utilisation (LBP, BSIF et LPQ) fusionnées. Le but de la fusion de descripteurs c'est l'augmentation de caractéristiques qui nous a permis de choisir les meilleurs de chaque descripteur).

b. Fusion de descripteur d'apprentissage en profond (ResNet)

Dans cette expérience, nous avons utilisé des modèles de réseaux de neurones convolutifs ResNet (ResNet-18, Resnet-50 et ResNet101) pour l'étape d'extraction de caractéristiques.

Nous avons utilisé en plus F_1 (Eq 4.3) pour la représentation de caractéristiques, F_2 (Eq 4.2) pour la normalisation, T-test en sélection de caractéristique, SVM pour la classification).

Le **Tableau 4.4** présente les résultats obtenus pour chaque descripteur sur les bases de données utilisées.

Tableau 4.4 : Performances du système avec les descripteurs ResNet-18, ResNet-50 et ResNet-101 et la fusion entre eux.

| Descripteurs | Cornell Kin | UB Kinface | Family 101 | KinFace W-I | KinFaceW-II |
|-------------------|----------------|----------------|----------------|----------------|----------------|
| ResNet 18 | 72.45 % | 68.74 % | 67.14 % | 65.41 % | 64.46 % |
| ResNet 50 | 74.74 % | 69.24 % | 68.57 % | 66.23 % | 65.74 % |
| ResNet 101 | 76.86 % | 71.83 % | 69.93 % | 68.32 % | 67.25 % |
| ResNet(18+50) | 78.71 % | 72.37 % | 70.54 % | 69.63 % | 70.02 % |
| ResNet(18+101) | 80.32 % | 76.68 % | 75.42 % | 73.86 % | 72.45 % |
| ResNet(50+101) | 83.35 % | 79.36 % | 78.62 % | 76.86 % | 75.47 % |
| ResNet(18+50+101) | 87.16 % | 83.68 % | 82.07 % | 79.76 % | 76.89 % |

Dans cette étude, les caractéristiques du visage sont fournies par les réseaux de neurones convolutionnels profonds préformés ResNet, nous avons d'abord testé notre approche en utilisant un seul modèle de ResNet, puis nous avons fusionné les modèles entre eux par deux modèles et enfin tous les modèles.

Le **Tableau 4.4** montre que l'utilisation d'un seul modèle (ResNet -18, ResNet-50 ou ResNet-101) donne des résultats inférieurs à l'utilisation de ResNet- (18 + 50), ResNet- (18 + 101) ou ResNet (50 + 101). Notre approche donne de meilleurs résultats lors de l'utilisation de ResNet- (18 + 50 +101).

D'après nos expériences, nous observons que le ResNet (18+50+101) améliore les résultats ce qui rend le système plus efficace.

4.4 Effet de la représentation du visage (représentation faciale)

Dans cette expérience nous avons étudié l'effet de la représentation faciale, nous avons comparé les performances obtenues par trois représentations faciales : MB, ML et PML. Chacun a été testé avec plusieurs niveaux de 1 à 10. Nous avons utilisé en plus (Le descripteur LPQ, Eq 4.1 en représentation de caractéristiques, Eq 4.2 pour la normalisation, Score de Fisher en sélection de caractéristiques, SVM pour la classification).

La **Figure 4.2** résume les performances associées aux différentes représentations faciales. Elle présente également le nombre de caractéristiques en fonction du nombre de niveaux de la

représentation faciale. Ces résultats ont été obtenus à partir de la base de données KinFace W-II. Les courbes associées aux autres bases de données ont une forme similaire.

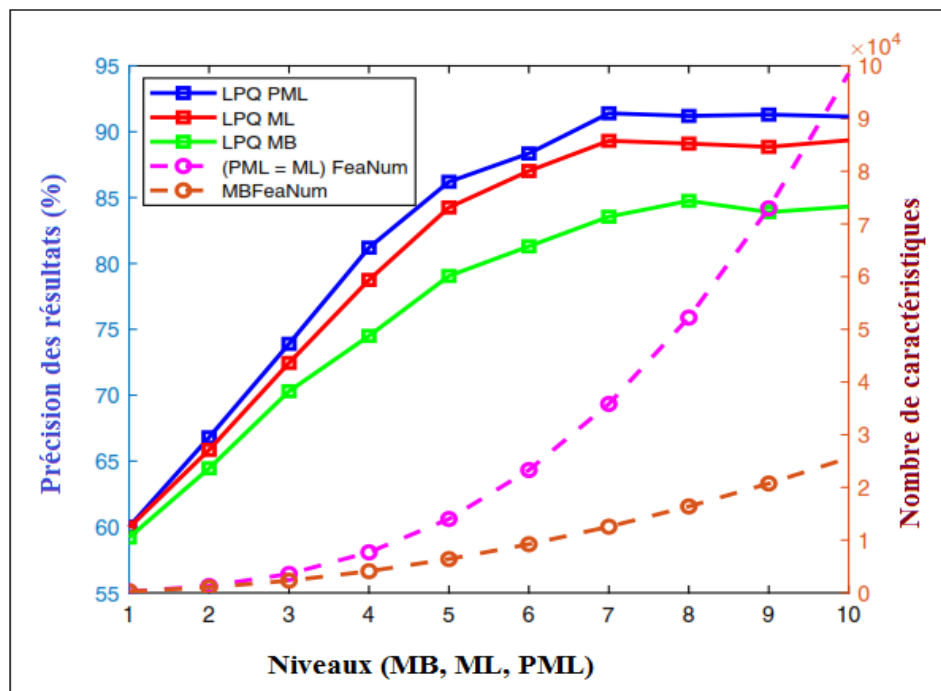


Figure 4.2 : La précision (%) et le nombre de caractéristiques aux différents niveaux de la représentation du visage

D'après nos expériences, nous observons qu'en augmentant le niveau de la représentation faciale, le nombre de caractéristiques des descripteurs augmente. D'autre part, les performances s'améliorent et l'étape d'extraction de caractéristiques prend du temps.

Pour chaque niveau, le nombre de caractéristiques pour les représentations PML et ML est identique et supérieur à celui associé à la représentation MB, ce qui a permis d'obtenir avec MB des performances inférieures à celles de ML et PML. Malgré le fait que les représentations ML et PML aient le même nombre de caractéristiques pour un nombre donné de niveaux, la représentation PML a obtenu de meilleurs résultats que la représentation ML en raison de l'avantage de la PML en termes d'équilibre de poids des caractéristiques. On observe également qu'après le niveau 7, les résultats se sont légèrement améliorés malgré l'augmentation du nombre de caractéristiques. Nous concluons que l'utilisation de PML avec le niveau 7 donne de bons résultats avec un nombre raisonnable de caractéristiques et moins de temps du calcul.

4.5 Effet de la représentation des caractéristiques

Puisque l'entrée dans le système de vérification de la parenté est une paire d'images, dans cette expérience nous avons étudié différents schémas pour fusionner les caractéristiques des deux images à un seul vecteur de caractéristiques utilisant les équations (3.1-3.6) mentionnées dans le chapitre précédent.

$$F_1 = |F_{Parent} - F_{Enfant}|$$

$$F_2 = |F_{Parent} + F_{Enfant}|$$

$$F_3 = F_{Parent} / F_{Enfant}$$

$$F_4 = F_{Parent} * F_{Enfant}$$

$$F_5 = \frac{F_{Parent} * F_{Enfant}}{|F_{Parent} - F_{Enfant}|}$$

$$F_6 = \frac{F_{Parent} + F_{Enfant}}{|F_{Parent} - F_{Enfant}|}$$

Dans les équations ci-dessus, la fusion est effectuée de manière élémentaire (bit par bit).

Nous avons mené plusieurs expériences pour l'étape de représentation de caractéristiques fusionnées en utilisant différentes combinaisons telles que la différence absolue (F_1), somme (F_2), division (F_3), et la multiplication (F_4) et d'autres (F_5 et F_6). Nous avons testé les descripteurs (LDP, LTP, LBP, WLD, BSIF et LPQ) utilisant les schémas de fusion de paires mentionnés ci-dessus.

Le but de cette expérience est de montrer l'influence des schémas de fusion sur les différents descripteurs. Les résultats ont été obtenus sur la base de données KinFace-WII comme exemple.

Le tableau 4.5 résume les performances de différents schémas de fusion de paires obtenus.

D'après le tableau ci-dessus, nous pouvons voir que le schéma de fusion qui a utilisé la différence absolue (F_1) a obtenu les meilleurs résultats par rapport aux schémas de fusion : somme, division et multiplication. Pour cette raison, nous avons inclus cette distance dans les schémas de fusion proposés F_5 et F_6 .

Les performances sont considérablement augmentées lors de l'utilisation du schéma F_6 avec les descripteurs LBP, BSIF, WLD et LPQ. C'était le meilleur schéma de fusion pour le descripteur LPQ qui est utilisé dans les expériences restantes.

Tableau 4.5 : Performances des différents schémas de fusion de paires pour chaque descripteur

| Descripteurs | F ₁ | F ₂ | F ₃ | F ₄ | F ₅ | F ₆ |
|--------------|----------------|----------------|----------------|----------------|----------------|----------------|
| LDP | 66.10 % | 51.86 % | 64.05 % | 59.10 % | 59.60 % | 56.16 % |
| LTP | 72.35 % | 52.64 % | 71.25 % | 70.15 % | 71.35 % | 62.45 % |
| LBP | 80.40 % | 55.15 % | 76.65 % | 69.90 % | 71.55 % | 87.30 % |
| WLD | 79.65 % | 60.32 % | 75.68 % | 68.61 % | 67.32 % | 88.01 % |
| BSIF | 78.30 % | 57.98 % | 73.30 % | 73.30 % | 65.40 % | 89.15 % |
| LPQ | 74.30 % | 58.69 % | 77.50 % | 65.85 % | 68.70 % | 91.40 % |

4.6 Effet de la Réduction des caractéristiques (Projection et Sélection)

Dans cette expérience, nous étudions l'effet de la réduction de caractéristiques à travers des expériences sur la projection et la sélection de caractéristiques.

4.6.1 Sélection des caractéristiques

Dans cette expérience, nous avons utilisé et comparé différentes fonctions (score de Fisher, score KL, score t-test et le score proposé) pour l'étape de sélection de caractéristiques. Après sélection des caractéristiques, les caractéristiques ont été classées par ordre décroissant. Nous avons utilisé en plus (Le descripteur LPQ – PML-7, Eq 4.1 en représentation de caractéristiques, Eq 4.2 pour la normalisation, SVM pour la classification). Le tableau 4.6 résume les performances obtenues en utilisant la sélection de caractéristiques pour les bases de données utilisées.

Notre approche de sélection des caractéristiques proposée utilise la différence entre le

T-test et KL. Ceci est donné par l'équation suivant :

$$W_i = W_{T-test}(i) - W_{KL}(i) \quad 4.12$$

où W_i est le poids de la caractéristique i .

Ce choix était basé sur un ensemble d'expériences avec différentes relations mathématiques. Nous avons constaté que les meilleurs poids caractéristiques étaient obtenus lorsque le poids du T-test (Eq 3.25) était très élevé et le poids KL (Eq 3.26) était très faible.

La sélection de caractéristiques proposée a surpassé la fonction de Fisher, la fonction KL et la fonction du t-test. Pour avoir une bonne idée du choix de la sélection de caractéristiques proposée, nous avons mené l'expérience suivante :

Tableau 4.6 : Performances obtenues en utilisant la sélection des caractéristiques

| Bases de données | KL | T-test | Fisher | Proposé |
|------------------|---------|---------|---------|----------------|
| Cornell Kin | 62.01 % | 92.36 % | 94.19 % | 95.21 % |
| UB KinFace | 60.41 % | 88.47 % | 93.46 % | 94.46 % |
| Family 101 | 59.78 % | 87.71 % | 92.13 % | 93.20 % |
| KinFace W-I | 58.92 % | 82.61 % | 89.70 % | 91.40 % |
| KinFace W-II | 56.35 % | 76.24 % | 88.97 % | 90.38 % |

Tout d'abord, nous avons classé les caractéristiques du KinFace W-II en fonction du poids de Fisher. Deuxièmement, nous avons conservé les premières 1000 caractéristiques sélectionnées. À ce stade, nous avons 1000 caractéristiques pertinentes. Troisièmement, pour chacune de ces caractéristiques, nous avons calculé son poids selon quatre fonctions différentes : t-test, KL, t-test + KL et notre test t de poids proposé t-test - KL.

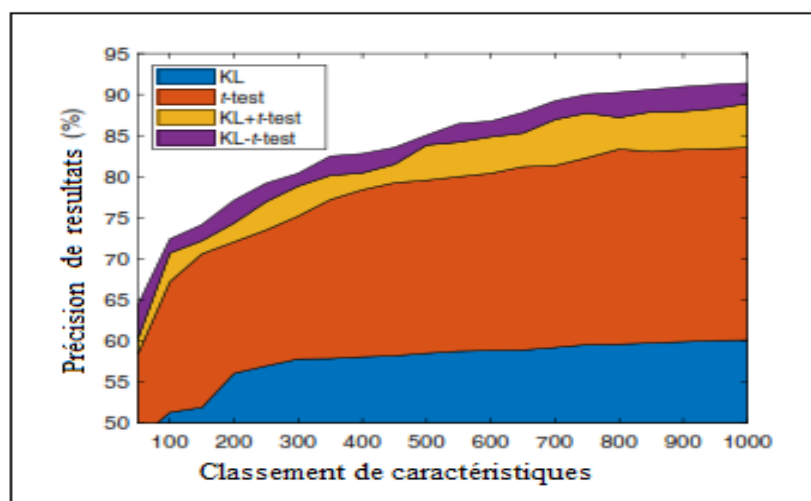


Figure 4.3 : Précision obtenue (%) en utilisant différents schémas de sélection de caractéristiques

La **Figure 4.3** illustre les quatre poids pour les 1000 caractéristiques retenues. Nous pouvons observer que le score de Fisher et le t-test ont obtenu de bonnes performances malgré la légère supériorité du score de Fisher, qui est une technique de sélection de caractéristiques bien connue. KL a réalisé une faible performance par rapport à celle des autres schémas de sélection de caractéristiques. Notre idée était de générer une fonction de poids caractéristique plus sensible qui était dérivée du poids du t-test et du poids KL, comme indiqué dans l'équation (4.7). Les performances sont considérablement améliorées lors de l'utilisation de la fonction de sélection de caractéristiques proposée.

D'après le **Tableau 4.6** et la **Figure 4.3** nous avons observé la supériorité de notre sélection de caractéristiques proposée en termes de précision par rapport aux autres schémas de sélection de caractéristiques.

4.6.2 Projections des caractéristiques

Dans cette expérience, nous avons utilisé et comparé différentes fonctions (LDA, LPP, PCA) pour l'étape de la projection de caractéristiques. Après cette étape, les caractéristiques ont été classées par ordre décroissant. Nous avons utilisé en plus (Le descripteur HOG, Eq 4.3 en représentation de caractéristiques, Eq 4.4 pour la normalisation, SVM pour la classification).

Le **Tableau 4.7** résume les performances obtenues en utilisant la projection de caractéristiques pour les bases de données utilisées.

Tableau 4.7 : Performances obtenues en utilisant la projection des caractéristiques

| Bases de données | LDA | LPP | PCA |
|---------------------|---------|---------|----------------|
| Cornell Kin | 77.65 % | 82.74 % | 84.36 % |
| UB KinFace | 76.36 % | 81.04 % | 83.20 % |
| Family 101 | 74.89 % | 79.55 % | 81.03 % |
| KinFace W-I | 73.27 % | 78.24 % | 79.61 % |
| KinFace W-II | 71.87 % | 77.51 % | 78.74 % |

D'après le tableau ci-dessus, nous pouvons voir que la fonction de la projection de caractéristiques PCA a obtenu les meilleurs résultats par rapport aux schémas (LPP et LDA).

4.6.2 Combinaisons entre la sélection et la projection des caractéristiques

Dans cette expérience nous avons choisis deux meilleurs types de la réduction de caractéristiques d'après les expériences précédentes [(PCA) pour la projection et (la fonction proposée) pour la sélection], pour démontrer le phénomène de la combinaison entre la projection et la sélection de caractéristiques pour bien réduire le nombre et choisir les meilleures caractéristiques. Nous avons utilisé en plus (Le descripteur HOG, Eq 4.3 en représentation de caractéristiques, Eq 4.4 pour la normalisation, SVM pour la classification).

Le **Tableau 4.8** résume les performances obtenues en utilisant combinaisons entre la sélection et la projection pour les bases de données utilisées.

Tableau 4.8 : Performances obtenues en utilisant la combinaison entre la sélection et la projection

| Bases de données | Sans | PCA | Proposé | PCA + Proposé |
|---------------------|---------|---------|---------|----------------|
| Cornell Kin | 81.39 % | 84.36 % | 86.37 % | 88.94 % |
| UB KinFace | 80.14 % | 83.20 % | 84.98 % | 87.32 % |
| Family 101 | 79.35 % | 81.03 % | 82.71 % | 85.68 % |
| KinFace W-I | 77.87 % | 79.61 % | 81.36 % | 83.28 % |
| KinFace W-II | 75.22 % | 78.74 % | 80.11 % | 82.03 % |

A partir des résultats présentés au **Tableau 4.8**, nous pouvons observer que l'utilisation du schéma proposé pour la sélection atteint une bonne précision. Il a été suivi par la projection utilisant PCA. Sans l'utilisation des deux, nous avons obtenu une faible précision et la plus grande précision obtenu lors de la combinaison entre la projection et la sélection.

L'idée de cette proposition est de réduire le nombre de caractéristiques, dans un premier temps la projection de caractéristiques dans une autre référence, ensuite trie les caractéristiques en fonction du poids selon le schéma de sélection proposé et le choix de nombre nécessaire pour l'étape de classification.

4.7 Effet des Classifieurs

Pour donner le résultat final (Kin ou Non-Kin), un classifieur efficace doit être utilisé. Pour cette raison, nous avons comparé des classifieurs connus pour prouver leurs efficacités.

Nous avons testé notre approche en utilisant Cosin, KNN (k-Nearest Neighbors) et SVM pour voir l'impact sur notre approche par rapport aux autres. Nous avons utilisé en plus (Le descripteur LPQ, PML-7 pour la représentation faciale, F_1 en représentation de caractéristiques, F_2 pour la normalisation).

Le **Tableau 4.9** montre les performances obtenues en utilisant les trois classifieurs pour les bases de données utilisées.

Tableau 4.9 : les performances obtenues en utilisant les trois classifieurs (Cosin, KNN, SVM)

| Bases de données | Cosin | KNN | SVM |
|---------------------|---------|---------|----------------|
| Cornell Kin | 73.12 % | 84.32 % | 95.21 % |
| UB KinFace | 71.92 % | 82.98 % | 94.46 % |
| Family 101 | 70.34 % | 80.32 % | 93.20 % |
| KinFace W-I | 69.11 % | 78.54 % | 91.40 % |
| KinFace W-II | 68.03 % | 75.36 % | 90.38 % |

Dans l'évaluation des performances de notre approche, nous pouvons observer que le classifieur SVM atteint la plus grande précision. Il a été suivi du classifieur KNN, il a atteint une moyenne précision, et encore inférieure à celle de SVM. Le classifieur Cosin a obtenu une faible précision par rapport aux premiers classifieurs. Pour ces raisons, nous avons choisi d'utiliser SVM comme classifieur principal.

D'après nos expériences, nous concluons que dans les problèmes binaires, SVM est efficace par rapport à Cosin et KNN. De plus, le SVM produit une précision significative avec moins de puissance de calcul.

4.8 Temps de calcul (CPU et Nombre des caractéristiques)

Le but de cette expérience est de calculer le temps d'apprentissage et de test dans la phase de classification avec et sans l'utilisation de la réduction de caractéristiques.

Dans un premier temps, nous avons utilisé le descripteur LPQ avec une représentation PML a niveau 7. Le descripteur correspondant comportait 38500 caractéristiques. Nous avons utilisé les données totales de (Nombre de paires \times 38500) pour les cinq bases de données sans

utiliser de fonction de sélection de caractéristiques. Les résultats obtenus (précision, temps d'Apprentissage CPU A, temps de Test CPU T) sont présentés dans le **Tableau 4.10**.

Tableau 4.10 : Performances obtenues et leurs CPU A et CPU T.

| Bases de données | Nbr de pairs | LPQ-PML-7 | Temps CPU A | Temps CPU T | Précision |
|------------------|--------------|-----------|-------------|-------------|-----------|
| Cornell Kin | 286 | 38500 | 43.2561 s | 01.3745 s | 94.32 % |
| UB KinFace | 600 | 38500 | 91.9812 s | 03.6322 s | 93.68 % |
| Family 101 | 2000 | 38500 | 386.3212 s | 13.3652 s | 92.93 % |
| KinFace W-I | 1066 | 38500 | 169.3658 s | 06.3361 s | 90.66 % |
| KinFace W-II | 2000 | 38500 | 320.6213 s | 11.2164 s | 90.12 % |

Ensuite, nous avons utilisé la fonction de sélection proposée pour trier les caractéristiques en fonction de leur poids. Enfin, nous avons testé plusieurs dimensions de caractéristiques pour sélectionner le meilleur nombre. La **Figure 4.14** montre les courbes (la précision, le temps CPU A et le temps CPU T) en fonction du nombre de caractéristiques. Ces résultats ont été obtenus à partir de la base de données KinFace W-II. Les courbes associées aux autres bases de données ont une forme similaire.

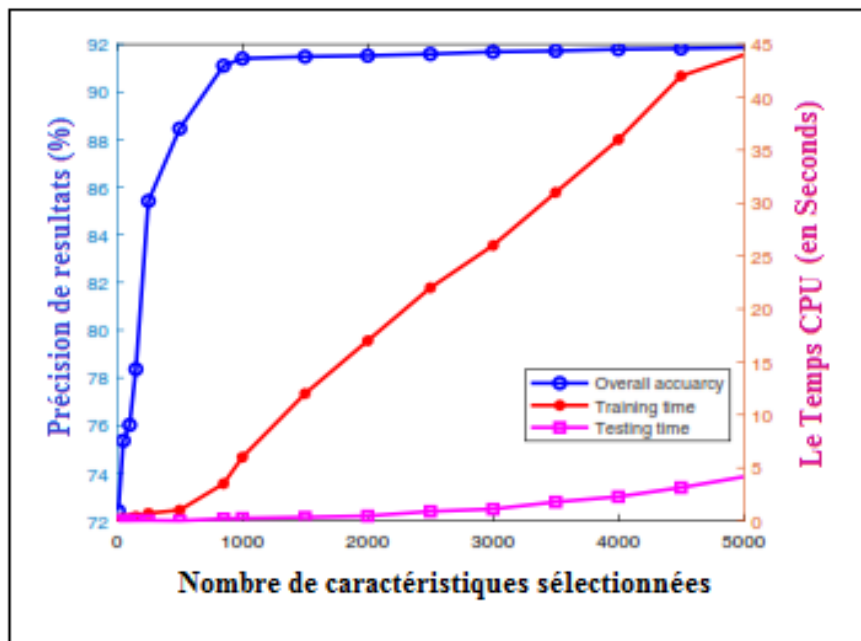


Figure 4.4 : La précision (%) et le temps CPU (formation / test) en fonction de nombre de caractéristiques

La **Figure 4.4** montre à la fois la précision de la vérification de parenté et le temps CPU (en secondes), qui était nécessaire pour l'apprentissage du modèle et les tests en fonction du nombre de caractéristiques conservées qui ont été obtenues après avoir utilisé notre schéma de sélection des caractéristiques proposé. Le temps CPU (Apprentissage/Test) augmente linéairement avec l'augmentation du nombre de caractéristiques.

De plus, la précision a augmenté de façon exponentielle jusqu'à (850 - 1100) caractéristiques, puis a varié en tant que fonction logarithmique, et après 1100 malgré que le nombre de caractéristiques augmente la précision augmente légèrement, par contre le temps CPU est très important. Sur cette base, le choix de 1000 caractéristiques peut être considéré comme un bon compromis entre précision et le temps de calcul.

Les expériences ont été effectuées sur un ordinateur portable DELL 7510 Precision (processeur Xeon E3-1535M v5, 8 Mo de cache, 2,90 GHz, 64 Go de RAM, GPU NVIDIA Quadro M2000M, Windows 10 en utilisant MATLAB R2018b).

4.9 Performances obtenues pour les différentes relations de parenté

Chaque base de données contient les quatre relations de parenté Père-fils (P-Fs), Père-Fille (P-Fe), Mère-Fils (M-Fs) et Mère-fille (M-Fe). Dans cette expérience nous avons présenté les précisions obtenues de notre approche pour chaque relation de parenté de toutes les bases de données utilisées. Le tableau (4.11) montre les performances de l'approche proposée avec différentes relations (P-Fs), (P-Fe), (M-Fs) et (M-Fe).

Tableau 4.11 : Performances obtenues utilisant les différentes relations de parenté.

| Bases de données | P – Fs | P – Fe | M – Fs | M – Fe | Moyenne |
|---------------------|---------|---------|---------|---------|----------------|
| Cornell Kin | 97.56 % | 96.97 % | 96.97 % | 89.36 | 95.21 % |
| UB KinFace | 95.84 % | 95.58 % | 93.84 % | 92.59 % | 94.46 % |
| Family 101 | 93.91 % | 93.30 % | 92.91 % | 92.67 % | 93.20 % |
| KinFace W-I | 92.00 % | 92.00 % | 91.60 % | 90.00 % | 91.40 % |
| KinFace W-II | 91.81 % | 90.29 % | 89.85 % | 89.85 % | 90.38 % |

D'après le **Tableau 4.11**, nous observons que :

1. La relation la plus facile à classer est la relation dans laquelle il y a un homme, qu'il s'agisse d'un enfant ou d'un père. Ainsi, la relation (P-Fs) est plus facile que (P-Fe) et (M-Fs) est plus facile que (M-Fe).
2. La relation la plus difficile à classer est la relation dans laquelle il y a une femme, fille ou mère. Ainsi (M-Fs) est plus difficile que (P-Fs) et (M-Fe) est plus difficile que (M-Fs).

Donc, la relation la plus simple à classer est (P-Fs) et la plus difficile à classer est (M-Fe).

4.10 Courbes caractéristiques de fonctionnement du récepteur (ROC)

D'après toutes les expériences précédentes, nous avons conclu les meilleurs résultats obtenus en utilisant les paramètres suivants :

- Le descripteur LPQ pour l'extraction des caractéristiques.
- PML niveau 7 pour la représentation faciale.
- La représentation des caractéristiques proposée.
- Le schéma de sélection de caractéristiques proposée.
- Le SVM pour la classification.

Pour mieux visualiser les performances de notre approche et les résultats obtenus, les courbes des caractéristiques de fonctionnement du récepteur (ROC) des différents effets (Descripteurs, Représentation faciales, Représentation des caractéristiques, Réduction des caractéristiques, Classifieurs) sont illustrées à la **Figure 4.5**. Ces résultats ont été obtenus à partir de la base de données KinFace W-II. Les courbes associées aux autres bases de données ont une forme similaire.

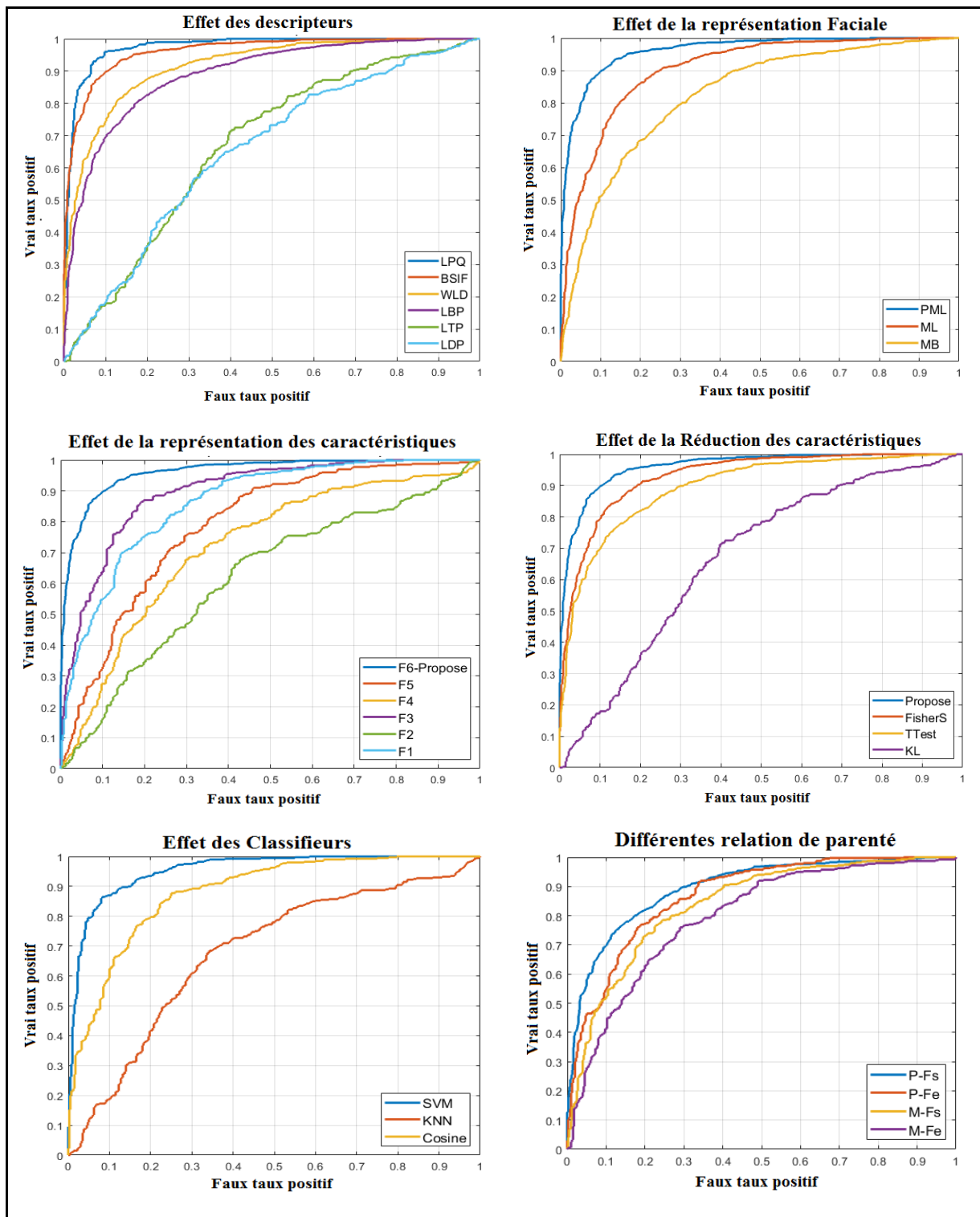


Figure 4.5 : ROC des différents effets dans la base de données KinFace W-II

4.11 Comparaison des approches existantes avec notre approche

L'évaluation des performances de plusieurs bases de données de référence accessibles au public confirme que notre approche surpasse les approches de pointe existantes. Le **Tableau 4.12** illustre les performances de notre approche de la vérification de parenté proposée ainsi que celle de certaines approches existantes. Comme on peut le voir dans ce tableau, nous

obtenons 95,21%, 94,46%, 93,20%, 91,40% et 90,38% de précision de vérification de parenté sur Cornell, UB KinFace, KinFace-I, KinFace-II et Family 101, respectivement. Cela confirme la robustesse et la stabilité de notre approche sur toutes les bases de données utilisées.

Tableau 4.12: Comparaison de l'approche proposée avec d'autres approches existantes.

| Approches | Cornell Kin | UB kinFace | KinFace W-I | KinFace W-II | Family 101 | |
|----------------------|-------------|----------------|----------------|----------------|-----------------|----------------|
| PSM 2010 [28] | 70.67 % | - | - | - | - | |
| TSL 2011 [30] | - | 69.67 % | - | - | - | |
| SSRWF 2012 [31] | - | 69.67 % | - | - | - | |
| MNRML 2014 [37] | - | - | 69.90 % | 76.50 % | - | |
| DMML 2014 [39] | 73.50 % | 74.50 % | 72.00 % | 78.00 % | - | |
| RSBM 2015 [51] | - | - | - | - | 69.6 % | |
| ESL 2015 [40] | - | - | 78.60 % | 75.70 % | - | |
| PDFL 2015 [42] | 71.90 % | 67.30 % | 70.10 % | 77.00 % | - | |
| CNN-pnt 2015 [44] | - | - | 77.50 % | 88.40 % | - | |
| SSML 2016 [123] | - | - | 79.55 % | 80.15 % | - | |
| S.scoring 2016 [124] | 81.40 % | 52.20 % | 71.40 % | 80.10 % | - | |
| EHRMFS 2017 [125] | - | - | 80.20 % | 80.16 % | - | |
| NRML 2017 [126] | - | - | 66.30 % | 78.70 % | - | |
| DDMML 2017 [128] | - | - | 83.50 % | 84.30 | - | |
| DCTNet 2018 [129] | - | - | 84.75 % | 89.25 % | - | |
| PML-COV 2018 [32] | - | 84.50 % | 88.20 % | 88.20 % | - | |
| WLD*RGB 2018 [43] | 92.06 % | 90.39 % | - | - | 83.20 % | |
| WGEML2019 [41] | - | - | 81.90 % | 82.80 % | - | |
| KML 2019 [130] | 81.40 % | 75.50 % | 82.80 % | 85.70 % | - | |
| Humans 2020 [131] | - | - | 81.10 % | 81.10 % | - | |
| GKR 2020 [132] | - | - | 79.20 % | 90.6% | - | |
| Proposé 2020 | [86] | 92.89 % | 90.59 % | 86.65 % | 81.11. % | 84.82 % |
| | [93] | 95.21 % | 94.46 % | 93.20 % | 91.40 % | 90.38 % |

4.12 Conclusion

Dans ce chapitre, nous avons proposé un système de vérification de la parenté où toutes les étapes du traitement ont été présentées et étudiées, en particulier l'influence de certains paramètres de chaque étape sur l'influence des performances du système, et ce, dans le but d'obtenir la bonne configuration nous permettant d'avoir les meilleurs résultats.

Notons que dans le but d'améliorer les performances du système proposé, nous avons proposé plusieurs techniques et méthodes pour atteindre cet objectif. Parmi ces méthodes on trouve : la représentation faciale, la représentation des caractéristiques et la sélection et la réduction des caractéristiques.

En procédant d'une manière séquentielle, les expériences menées sur des bases de données universelles et publiques ont montré que la meilleure combinaison des différents paramètres de chaque étape donnant la meilleure configuration en termes de performances est composée comme suit :

(1) Le descripteur d'image est le LPQ, (2) La représentation du visage est le PML niveau 7, (3) La représentation de paires de caractéristiques est notre schéma que nous avons proposé, (4) La sélection des caractéristiques est également notre schéma que nous avons proposé, et finalement (5) le classifieur est le SVM.

La comparaison des résultats obtenus par notre approche avec ceux obtenus par d'autres approches montre clairement que l'approche que nous avons proposée est nettement meilleure et efficace par rapport à la plupart des approches existantes.



Bibliographie

Bibliographie

- [1] Dehousse Franklin and Annabelle Littoz-Monnet (2004). l'union européenne face au défi du terrorisme. *Studia Diplomatica*, 57(4/5): 5–95. www.jstor.org/stable/44838352.
- [2] Perronnin Florent and Jean-Luc Dugelay (2002). Introduction à la biométrie-Authentification des individus par traitement audio-vidéo. *Traitement du signal*, 19(4), pp. 253-265. <http://hdl.handle.net/2042/2211>
- [3] Anil Kumar Jain, Ruud Bolle and Sharath Pankanti (2006). Biometrics: personal identification in networked society. Vol. 479. Springer Science & Business Media. DOI: 10.1007/978-0-387-32659-7
- [4] Simon Liu and Mark Silverman (2001). A practical guide to biometric security technology. *IT Professional*, 3(1), pp. 27-32. DOI: 10.1109/6294.899930
- [5] Lieberman Debra, John Tooby and Leda Cosmides (2007). The architecture of human kin detection. *Nature*, 445(7129), pp. 727-31. DOI: 10.1038/nature05510.
- [6] Trupin Éric (2005). La reconnaissance d'images de documents: Un panorama. *Traitement du signal*, 22(3), pp. 159-189. <http://hdl.handle.net/2042/4390>
- [7] Marée Raphaël. Classification automatique d'images par arbres de décision (2005). Doctoral dissertation, University of Liege - Electrical Engineering and Computer Science. <http://hdl.handle.net/2268/12203>.
- [8] Schmittbuhl Matthieu. Différentes techniques d'analyse d'images appliquées à la morphologie faciale des primates hominoïdes (1996). Doctoral dissertation in medical sciences, Strasbourg 1. <http://www.theses.fr/1996STR13205>
- [9] Tiago Figueiredo Vieira (2013). Identifying kinship cues from facial images. Doctoral dissertation in Electrical Engineering, Federal University of Pernambuco. <https://repositorio.ufpe.br/handle/123456789/13315>
- [10] David D. Zhang (2012). Biometric solutions: For authentication in an e-world. Vol. 697. Springer Science & Business Media. DOI: 10.1007/978-1-4615-1053-6
- [11] Anil K. Jain, Lin Hong and Sharath Pankanti (2000). Biometric identification. *Communications of the ACM*, 43(2), pp. 90-98. DOI: 10.1145/328236.328110
- [12] Anil K.Jain, Arun A. Ross, and Karthik Nandakumar, (2011). Introduction to biometrics. Springer Science & Business Media. DOI: 10.1007/978-0-387-77326-1.
- [13] Bertrand Leroy, Ahlame Chouakria, Isabelle Herlin and Edwin Diday (1996). Approche géométrique et classification pour la reconnaissance de visage. Congrès de Reconnaissance des Formes et Intelligence Artificielle, INRIA and IRISA and CNRS, France, pp.548-557. <https://hal.inria.fr/inria-00532693/document>
- [14] Stan Z. Li and Anil K. Jain, (2011). Handbook of face recognition. New York : springer. DOI: 10.1007/978-0-85729-932-1

- [15] Ennaama Faouzia, Khalid Benhida and Ahmed Boulahoual (2019). Comparative and analysis study of biometric systems. *Journal of Theoretical and Applied Information Technology*, 97(12), pp. 3466-3476. <http://www.jatit.org/volumes/Vol97No12/20Vol97No12.pdf>
- [16] Salah Eddine Bekhouche, Abdelkrim Ouafi, Abdelmalik Taleb-Ahmed, Abdenour Hadid and Azeddine Benlamoudi (2016). Facial age estimation using BSIF and LBP. In *Proceedings of the First International Conference on Electrical Engineering (ICEEB'14)*, Biskra, Algeria. DOI: 10.13140/RG.2.1.1933.6483/1
- [17] Salah Eddine Bekhouche, Fadi Dornaika, Abdelkrim Ouafi, Abdelmalik Taleb-Ahmed (2017). Personality traits and job candidate screening via analyzing facial videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, Honolulu, HI, USA, pp.10-13. DOI: 10.1109/CVPRW.2017.211
- [18] Azeddine Benlamoudi, Kamal Eddine Aiadi, Abdelkrim Ouafi, Djamel Samai, and Mourad Oussalah (2017). Face antispoofing based on frame difference and multilevel representation. *Journal of Electronic Imaging*, 26(4), pp. 043007(1-14). DOI: 10.1117/1.JEI.26.4.043007
- [19] Fares Bougourzi, Fadi Dornaika, Karim Mokrani, Abdelmalik Taleb-Ahmed and Yassine Ruichek (2020). Fusion Transformed Deep and Shallow features (FTDS) for Image-Based Facial Expression Recognition. *Expert Systems with Application*, 156: 113459. DOI: 10.1016/j.eswa.2020.113459.
- [20] Weiyang Liu, Yandong Wen, Zhiding Yu Ming Li, Bhiksha Raj and Le Song (2017). SpheroFace: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, HI, USA, pp. 212-220. DOI: 10.1109/CVPR.2017.713.
- [21] Yuanhao Guo, Hamdi Dibeklioglu and Laurens van der Maaten (2014). Graph-based kinship recognition. *22nd International Conference on Pattern Recognition*, Stockholm, pp. 4287-4292. DOI: 10.1109/ICPR.2014.735.
- [22] Guha, Martin (2009). *Psychology of Classroom Learning: An Encyclopedia*. Reference Reviews, 23(8): 11-14. DOI: 10.1108/09504120911003203.
- [23] David L. Sills (1968). *International encyclopedia of the social sciences*. New York: Macmillan Co. and Free Press, 1968. 17 vols, Social Work, Volume 14, Issue 2, Page 109. DOI: 10.1093/sw/14.2.109.
- [24] N., Pam M.S., (2013). KINSHIP. In *PsychologyDictionary.org*, May 11, 2013, <https://psychologydictionary.org/kinship/> (accessed July22,2018).
- [25] Salil Prabhakar, Sharath Pankanti and Anil K. Jain (2003). Biometric recognition: security and privacy concerns. *IEEE Security & Privacy*, 1(2), pp. 33-42. DOI: 10.1109/MSECP.2003.1193209.
- [26] Hamdi Dibeklioglu (2017). Visual Transformation Aided Contrastive Learning for Video-Based Kinship Verification. *IEEE International Conference on Computer Vision (ICCV)*, Venice, pp. 2478-2487. DOI: 10.1109/ICCV.2017.269.

- [27] Xiaoting Wu, Eric Granger, Xiaoyi Feng (2019). Audio-Visual Kinship Verification. arXiv preprint arXiv:1906.10096. <https://arxiv.org/abs/1906.10096>
- [28] Ruogu Fang, Kevin D. Tang, Noah Snavely, Tsuhan Chen (2010). Towards computational models of kinship verification. In Proceedings of the International Conference on Image Processing, ICIP 2010, Hong Kong, China, pp. 1577-1580. DOI: 10.1109/ICIP.2010.5652590
- [29] Xiuzhuang Zhou, Junlin Hu, Jiwen Lu, Yuanyuan Shang and Yong Guan (2011). Kinship verification from facial images under uncontrolled conditions. In Proceedings of the 19th ACM international conference on Multimedia, Scottsdale Arizona, USA, pp. 953-956. DOI: 10.1145/2072298.2071911
- [30] Ming Shao, Siyu Xia and Yun Fu (2011). Genealogical face recognition based on UB KinFace database. CVPR 2011 WORKSHOPS, Colorado Springs, CO, pp. 60-65. DOI: 10.1109/CVPRW.2011.5981801
- [31] Naman Kohli, Richa Singh and Mayank Vatsa (2012). Self-similarity representation of weber faces for kinship classification. IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), Arlington, VA, pp. 245-250. DOI: 10.1109/BTAS.2012.6374584
- [32] Moujahid, Abdelmalik, and Fadi Dornaika (2019). A pyramid multi-level face descriptor: application to kinship verification. *Multimedia Tools and Applications*, 78(7), pp. 9335-9354. DOI: 10.1007/s11042-018-6517-0
- [33] Chergui Abdelhakim, Ouchtati Salim, Telli Hichem, Fares Bougourzi, Salah Eddine Bekhouche (2018). LPQ and LDP descriptors with ml representation for kinship verification. In: The second edition of the International Workshop on Signal Processing Applied to Rotating Machinery Diagnostics (SIGPROMD'2018), El Djelfa, Algeria, pp. 1-10.
- [34] Chergui, Abdelhakim, Ouchtati Salim, Sequeira Jean, Bekhouche Salah Eddine and Bougourzi Fares (2018). Kinship verification using BSIF and LBP. International Conference on Signal, Image, Vision and their Applications (SIVA), Guelma, Algeria, pp. 1-5. DOI: 10.1109/SIVA.2018.8661085
- [35] Chergui Abdelhakim, Ouchtati Salim, Sequeira Jean, Benlamoudi Azeddine, Telli Hichem and Bekhouche Salah Eddine (2018). Robust kinship verification using local descriptors. Proceeding of Third International Conference On Advanced Technologies And Electrical Engineering ICTAEE18, Skikda, Algeria.
- [36] Siyu Xia, Ming Shao, and Yun Fu (2011). Kinship verification through transfer learning. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, Barcelona, Catalonia, Spain, pp. 2539–2544. DOI: 10.5591/978-1-57735-516-8/IJCAI11-422
- [37] Jiwen Lu, Xiuzhuang Zhou, Yap-Pen Tan, Yuanyuan Shang, Jie Zhou (2013). Neighborhood repulsed metric learning for kinship verification. *IEEE transactions on pattern analysis and machine intelligence*, 36(2), pp.331-345. DOI: 10.1109/TPAMI.2013.134

- [38] Junlin Hu, Jiwen Lu, Junsong Yuan and Yap-Peng Tan (2015). Large Margin Multimetric Learning for Face and Kinship Verification in the Wild. In: Cremers D., Reid I., Saito H., Yang MH. (eds) Computer Vision -- ACCV 2014. ACCV 2014. Lecture Notes in Computer Science, vol 9005, p. 252-267. Springer, Cham. DOI: 10.1007/978-3-319-16811-1_17
- [39] Haibin Yan, Jiwen Lu, Weihong Deng and Xiuzhuang Zhou (2014). Discriminative Multimetric Learning for Kinship Verification. IEEE Transactions on Information Forensics and Security, 9(7), pp. 1169-1178. DOI: 10.1109/TIFS.2014.2327757
- [40] Xiuzhuang Zhou, Yuanyuan Shang, Haibin Yan and Guodong Guo (2016). Ensemble similarity learning for kinship verification from facial images in the wild. Information Fusion, Volume 32, Part B, pp. 40-48, DOI : 10.1016/j.inffus.2015.08.006
- [41] Jianqing Liang, Qinghua Hu, Chuangyin Dang and Wangmeng Zuo (2019). Weighted Graph Embedding-Based Metric Learning for Kinship Verification. IEEE Transactions on Image Processing, 28(3), pp. 1149-1162. DOI: 10.1109/TIP.2018.2875346
- [42] Haibin Yan , Jiwen Lu and Xiuzhuang Zhou (2015). Prototype-Based Discriminative Feature Learning for Kinship Verification. IEEE Transactions on Cybernetics, 45(11), pp. 2535-2545. DOI: 10.1109/TCYB.2014.2376934
- [43] Chergui Abdelhakim, Ouchtati Salim, Sequeira, Jean, Bekhouche Salah Edine, Bougourzi Fares and Benlamoudi, Azeddine (2018). Discriminant analysis for facial verification using colorimages. In Proceeding of the Second International Conference on Electrical Engineering ICEEB'2018, Biskra, Algeria.
- [44] Yuting Zhang, Kihyuk Sohn, Ruben Villegas, Gang Pan and Honglak Lee (2015). Improving object detection with deep convolutional networks via Bayesian optimization and structured prediction. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, pp. 249-258. DOI: 10.1109/CVPR.2015.7298621
- [45] Naman Kohli, Mayank Vatsa, Richa Singh, Afzel Noore and Angshul Majumdar (2017). Hierarchical Representation Learning for Kinship Verification. IEEE Transactions on Image Processing, 26(1), pp. 289-302. DOI: 10.1109/TIP.2016.2609811
- [46] Afshin Dehghan, Enrique G. Ortiz, Ruben Villegas and Mubarak Shah (2014). Who Do I Look Like? Determining Parent-Offspring Resemblance via Gated Autoencoders. 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, pp. 1757-1764. DOI: 10.1109/CVPR.2014.227
- [47] Mengyin Wang, Zechao Li, Xiangbo Shu, Jingdong and Jinhui Tang (2015). Deep kinship verification. 2015 IEEE 17th International Workshop on Multimedia Signal Processing (MMSP), Xiamen, pp. 1-6. DOI: 10.1109/MMSP.2015.7340820
- [48] Chergui Abdelhakim, Ouchtati Salim, Sequeira Jean, Bekhouche Salah Eddine, Bougourzi Fares and Telli Hichem (2019). Deep features for kinship verification from facial images. 2019 International Conference on Advanced Systems and Emergent

- Technologies (IC_ASET), Hammamet, Tunisia, pp. 64-67. DOI: 10.1109/ASET.2019.8871011
- [49] Chergui Abdelhakim, Ouchtati Salim, Mavromatis Sébastien, Bekhouche Salah Eddine and Sequeira Jean (2019). Investigating Deep CNNs Models Applied in Kinship Verification through Facial Images. 2019 5th International Conference on Frontiers of Signal Processing (ICFSP), Marseille, France, pp. 82-87. DOI: 10.1109/ICFSP48124.2019.8938055
- [50] Ruogu Fang, Andrew C. Gallagher, Tsuhan Chen and Alexander Loui (2013). Kinship classification by modeling facial feature heredity. 2013 IEEE International Conference on Image Processing, Melbourne, VIC, pp. 2983-2987. DOI: 10.1109/ICIP.2013.6738614
- [51] Xiaoqian Qin, Xiaoyang Tan and Songcan Chen (2015). Tri-subjects kinship verification: Understanding the core of a family. 2015 14th IAPR International Conference on Machine Vision Applications (MVA), Tokyo, pp. 580-583. DOI: 10.1109/MVA.2015.7153258 1867
- [52] Hamdi Dibeklioglu, Albert Ali Salah and Theo Gevers (2012). Are You Really Smiling at Me? Spontaneous versus Posed Enjoyment Smiles. In: Fitzgibbon A., Lazebnik S., Perona P., Sato Y., Schmid C. (eds) Computer Vision – ECCV 2012. ECCV 2012. Lecture Notes in Computer Science, vol 7574. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-33712-3_38
- [53] Haibin Yan and Junlin Hu (2018). Video-based kinship verification using distance metric learning. *Pattern Recognition*, 75, pp.15-24. DOI: 10.1016/j.patcog.2017.03.001
- [54] Naman Kohli, Daksha Yadav, Mayank Vatsa, Richa Singh and Afzel Noore (2019). Supervised Mixed Norm Autoencoder for Kinship Verification in Unconstrained Videos. *IEEE Transactions on Image Processing*, 28(3), pp. 1329-1341. DOI: 10.1109/TIP.2018.2840880
- [55] Li Huiqi, and Opas Chutatape (2000). Fundus image features extraction. Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Cat. No.00CH37143), Chicago, IL, vol.4, pp. 3071-3073. DOI: 10.1109/IEMBS.2000.901530
- [56] Bekhouche Salah Eddine (2017). Facial Soft Biometrics: Extracting demographic traits. Doctoral dissertation in Signals and Communications, Faculty of Science and Technology, Mohamed Khider University - Biskra. <http://thesis.univ-biskra.dz/3011/>
- [57] W. M. K. Wan Mohd Khairrosfaizal and A. J. Nor'aini (2009). Eyes detection in facial images using Circular Hough Transform. 2009 5th International Colloquium on Signal Processing & Its Applications, Kuala Lumpur, pp. 238-242. DOI: 10.1109/CSPA.2009.5069224
- [58] Paul Viola and Michael Jones (2001). Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on

- Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, pp. I-I. DOI: 10.1109/CVPR.2001.990517
- [59] Rein-Lien Hsu, M. Abdel-Mottaleb and A. K. Jain (2002). Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), pp. 696-706. DOI: 10.1109/34.1000242
- [60] Vahid Kazemi and Josephine Sullivan (2014). One millisecond face alignment with an ensemble of regression trees. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, pp. 1867-1874. DOI: 10.1109/CVPR.2014.241
- [61] Luigi Stringa (1993). Eyes detection for face recognition. *Applied Artificial Intelligence an International Journal*, 7(4), pp. 365-382. DOI: 10.1080/08839519308949995.
- [62] Sung Eun Choi, Youn Joo Lee, Sung Joo Lee, Kang Ryoung Park and Jaihie Kim (2010). A comparative study of local feature extraction for age estimation. In *2010 11th International Conference on Control Automation Robotics & Vision* (pp. 1280-1284), Singapore, Singapore. DOI: 10.1109/ICARCV.2010.5707432
- [63] Salah Eddine Bekhouche, Abdelkrim Ouafi, Abdelmalik Taleb-Ahmed, Abdenour Hadid and Azeddine Benlamoudi (2016). Facial age estimation using BSIF and LBP. *arXiv preprint arXiv:1601.01876*. DOI: 10.13140/RG.2.1.1933.6483/1
- [64] Bangalore S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan and Akio Yamada (2001). Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6), pp. 703-715. DOI: 10.1109/76.927424
- [65] Ping Wu, Bangalore S. Manjunath, Shawn D. Newsam and Hyundoo D. Shin (2000). A texture descriptor for browsing and similarity retrieval. *Signal processing: Image communication*, 16(1-2), pp.33-43. DOI : 10.1016/S0923-5965(00)00016-3
- [66] Timo Ojala, Matti Pietikäinen, and David Harwood (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition* 29(1), pp. 51-59. DOI: 10.1016/0031-3203(95)00067-4
- [67] Timo Ahonen, Abdenour Hadid and Matti Pietikäinen (2006). Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), pp. 2037-2041. DOI: 10.1109/TPAMI.2006.244
- [68] Timo Ojala, Matti Pietikäinen and Topi Mäenpää (2000). Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *Computer Vision - ECCV 2000. ECCV 2000. Lecture Notes in Computer Science*, vol 1842. Springer, Berlin, Heidelberg. DOI: 10.1007/3-540-45054-8_27
- [69] Timo Ojala, Matti Pietikainen and Topi Maenpaa (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), pp. 971-987. DOI: 10.1109/TPAMI.2002.1017623

- [70] Wen-Hung Liao (2010). Region Description Using Extended Local Ternary Patterns. 2010 20th International Conference on Pattern Recognition, Istanbul, pp. 1003-1006. DOI: 10.1109/ICPR.2010.251
- [71] Wen-Hung Liao and Ting-Jung Young (2010). Texture Classification Using Uniform Extended Local Ternary Patterns," 2010 IEEE International Symposium on Multimedia, Taichung, pp. 191-195. DOI: 10.1109/ISM.2010.35
- [72] Taskeed Jabid, Md Hasanul Kabir and Oksam Chae (2010). Local Directional Pattern (LDP) – A Robust Image Descriptor for Object Recognition," 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance, Boston, MA, pp. 482-487. DOI: 10.1109/AVSS.2010.17
- [73] Taskeed Jabid, Md Hasanul Kabir and Oksam Chae (2010). Local Directional Pattern (LDP) for face recognition. 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE), Las Vegas, NV, pp. 329-330. DOI: 10.1109/ICCE.2010.5418801
- [74] Ville Ojansivu and Janne Heikkilä (2008). Blur Insensitive Texture Classification Using Local Phase Quantization. In: Elmoataz A., Lezoray O., Nouboud F., Mammass D. (eds) Image and Signal Processing. ICISP 2008. Lecture Notes in Computer Science, vol 5099. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-540-69905-7_27
- [75] Juho Kannala and Esa Rahtu (2012). BSIF: Binarized Statistical Image Features," Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, pp. 1363-1366. <https://ieeexplore.ieee.org/document/6460393>
- [76] Sahar Qasem Saleh, Muhammad Hussain, Ghulam Muhammad and George Bebis (2013). Evaluation of Image Forgery Detection Using Multi-scale Weber Local Descriptors. In: Bebis G. et al. (eds) Advances in Visual Computing. ISVC 2013. Lecture Notes in Computer Science, vol 8034. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-41939-3_40
- [77] Muhammad Hussain, Sahar Qasem Saleh, George Bebis, Ghulam Muhammad, Hatim Aboalsamh, Hassan Mathkour (2015). Evaluation of image forgery detection using multi-scale Weber local descriptors. International Journal on Artificial Intelligence Tools, 24(4), pp. 1540016/1-1540016/28. DOI : 10.1142/s0218213015400163
- [78] Navneet Dalal and Bill Triggs (2005). Histograms of oriented gradients for human detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, vol. 1, pp. 886-893. DOI: 10.1109/CVPR.2005.177
- [79] Yann LeCun (2015). LeNet-5, convolutional neural networks. URL: <http://yann.lecun.com/exdb/lenet>
- [80] Kim Yoon (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*. DOI: 10.3115/v1/D14-1181

- [81] Alex Krizhevsky, Alex, Ilya Sutskever and Geoffrey E. Hinton (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), pp. 84-90. DOI : 10.1145/3065386
- [82] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon and Ali Farhadi (2016). XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks. In: Leibe B., Matas J., Sebe N., Welling M. (eds) *Computer Vision – ECCV 2016*. ECCV 2016. *Lecture Notes in Computer Science*, vol 9908. Springer, Cham. DOI: 10.1007/978-3-319-46493-0_32
- [83] Omkar M. Parkhi, Andrea Vedaldi and Andrew Zisserman (2015). Deep Face Recognition. In Xianghua Xie, Mark W. Jones, and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 41.1-41.12. BMVA Press. DOI: 10.5244/C.29.41
- [84] Qawaqneh, Zakariya, Arafat Abu Mallouh, and Buket D. Barkana (2017). Deep convolutional neural network for age estimation based on VGG-face model. arXiv preprint arXiv:1709.01664. <https://arxiv.org/abs/1709.01664>
- [85] El Khiyari Hachim, and Harry Wechsler (2016). Face recognition across time lapse using convolutional neural networks. *Journal of Information Security* 7(3), pp.141-151. DOI: 10.4236/jis.2016.73010
- [86] Abdelhakim Chergui, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche, Mohamed Lashab and Jean Sequeira (2020). Kinship Verification Through Facial Images Using CNN-Based Features. *Traitement du Signal*, 37(1), pp.1-8. DOI: 10.18280/ts.370101
- [87] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun (2016). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, pp. 770-778. DOI: 10.1109/CVPR.2016.90
- [88] Lun Zhang, Rufeng Chu, Shiming Xiang, Shengcai Liao and Stan Z. Li (2007). Face Detection Based on Multi-Block LBP Representation. In: Lee SW., Li S.Z. (eds) *Advances in Biometrics. ICB 2007*. *Lecture Notes in Computer Science*, vol 4642. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-540-74549-5_2
- [89] Abdelhakim Chergui, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche, Jean Sequeira and Houssein Zerrari (2020). Kinship Verification using Mixed Descriptors and Multi Block Face Representation. 2019 International Conference on Networking and Advanced Systems (ICNAS), Annaba, Algeria, pp. 1-6. DOI: 10.1109/ICNAS.2019.8807875
- [90] Dat Tien Nguyen, So Ra Cho, Kwang Yong Shin, Jae Won Bang and Kang Ryoung Park (2014). Comparative study of human age estimation with or without preclassification of gender and facial expression. *The Scientific World Journal*, Volume 2014, Article ID 905269, 15 pages. DOI: 10.1155/2014/905269

- [91] Salah Eddine Bekhouche, Abdelkrim Ouafi, Azeddine Benlamoudi, Abdelmalik Taleb-Ahmed and Abdenour Hadid (2015). Facial age estimation and gender classification using multi level local phase quantization. 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT), Tlemcen, Algeria, pp. 1-4. DOI: 10.1109/CEIT.2015.7233141
- [92] Salah Eddine Bekhouche, Abdelkrim Ouafi, Fadi Dornaika, Abdelmalik Taleb-Ahmed and Abdenour Hadid (2017). Pyramid multi-level features for facial demographic estimation. *Expert Systems with Applications*, 80, 297-310. DOI: 10.1016/j.eswa.2017.03.030
- [93] Abdelhakim Chergui, Salim Ouchtati, Sébastien Mavromatis, Salah Eddine Bekhouche, Jean Sequeira and Fadi Dornaika (2020). Kinship verification through facial images using multiscale and multilevel handcrafted features. *Journal of Electronic Imaging* 29(2), 023017. DOI: 10.1117/1.JEI.29.2.023017
- [94] Dale R. Setla (1999). Fingerprint sensor having spoof reduction features and related methods. U.S. Patent Number 5,953,441. <https://patents.google.com/patent/US5953441>
- [95] Michael L. Raymer, William F. Punch, Erik D. Goodman, Leslie A. Kuhn and Anil K. Jain (2000). Dimensionality reduction using genetic algorithms. *IEEE Transactions on Evolutionary Computation*, 4(2), pp. 164-171. DOI: 10.1109/4235.850656
- [96] Irena Botwinik-Rotem (2004). The category P: Features, projections, interpretation. Thesis submitted for the degree “Doctor of Philosophy”, Tel Aviv University, 2004. <https://www.tau.ac.il/~botwinik/TheCategoryP.pdf>
- [97] Hervé Abdi, Lynne J. Williams (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4), pp. 433-459. DOI: 10.1002/wics.101
- [98] Balakrishnama Suresh and Aravind Ganapathiraju (1998). Linear discriminant analysis-a brief tutorial. Institute for Signal and information Processing, pp.1-8. https://www.isip.piconepress.com/publications/reports/1998/isip/lda/lda_theory.pdf
- [99] Zhonglong Zheng, Fan Yang, Wenan Tan, Jiong Jia and Jie Yang (2007). Gabor feature-based face recognition using supervised locality preserving projection. *Signal Processing*, 87(10), pp.2473-2483. DOI : 10.1016/j.sigpro.2007.03.006
- [100] Robnik-Šikonja, Marko and Igor Kononenko (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine learning*, 53, pp.23-69. DOI: 10.1023/A:1025667309714
- [101] Hans Lohninger (1999). Teach/Me - Data Analysis: Single User. Editeur : Springer-Verlag Berlin and Heidelberg GmbH & Co. K. <https://www.amazon.fr/Teach-Me-Data-Analysis-Single/dp/3540147438>
- [102] Abdi, Hervé (2007). Z scores. In N. J. Salkind (Ed.), *Encyclopedia of measurement and statistics* (Vol. 3, pp. 1057-1058). Thousand Oaks, CA: SAGE Publications, Inc. DOI: 10.4135/9781412952644.n487

- [103] Richard O. Duda, Peter E. Hart, and David G. Stork. (2001). Pattern Classification, 2nd Edition. John Wiley & Sons. New York 58 (2001): 16.
- [104] Guyon Isabelle and André Elisseeff (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(7-8), pp. 1157-1182. DOI: 10.1162/153244303322753616
- [105] Kullback Solomon and Richard A. Leibler (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1), pp. 79-86. DOI:10.1214/aoms/1177729694
- [106] Chouaib Hassan (2011). Sélection de caractéristiques: méthodes et applications. Doctoral dissertation in Computer Sciences ? Paris Descartes University: Paris, France. http://helios.mi.parisdescartes.fr/~vincent/siten/en/Publications/Conf_inter/pdf/chouaib.pdf
- [107] Haliche Zohra (2010). Classification et segmentation d'images texturées basée sur la théorie des ensembles. Doctoral dissertation in automatic, Mouloud Mammeri University, Algeria. <https://dl.ummo.dz/handle/ummo/728>
- [108] Hieu V. Nguyen and Li Bai (2010). Cosine Similarity Metric Learning for Face Verification. In: Kimmel R., Klette R., Sugimoto A. (eds) *Computer Vision – ACCV 2010*. ACCV 2010. Lecture Notes in Computer Science, vol 6493. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-19309-5_55
- [109] Piotr, Piotr and Rajeev Motwani (1998). Approximate nearest neighbors: towards removing the curse of dimensionality. *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pp. 604-613. DOI: 10.1145/276698.276876
- [110] Khan Maleq, Qin Ding and William Perrizo (2002). k-nearest neighbor classification on spatial data streams using P-trees. *k-nearest Neighbor Classification on Spatial Data Streams Using P-trees*. In: Chen MS., Yu P.S., Liu B. (eds) *Advances in Knowledge Discovery and Data Mining*. PAKDD 2002. Lecture Notes in Computer Science, vol 2336, pp. 517-528, Springer, Berlin, Heidelberg. DOI: 10.1007/3-540-47887-6_51.
- [111] Min-Ling Zhang and Zhi-Hua Zhou (2005). A k-nearest neighbor based algorithm for multi-label classification. *2005 IEEE International Conference on Granular Computing*, Beijing, Vol. 2, pp. 718-721. DOI: 10.1109/GRC.2005.1547385
- [112] Kulis Brian (2012). Metric learning: A survey. *Foundations and trends in machine learning*, 5(4), pp. 287-364. DOI: 10.1561/22000000019
- [113] Vezzetti Enrico and Federica Marcolin (2015). *Similarity measures for face recognition*. Bentham Science Publishers. DOI: 10.2174/97816810804441150101
- [114] Khamar Khushbu (2013). Short text classification using Knn based on distance function. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(4), pp. 1916-1919. <https://ijarccce.com/recent-issue-april-2013/#>
- [115] Cortes Corinna and Vladimir Vapnik (1995). Support-vector networks. *Machine learning*, 20(3), pp 273-297. DOI : 10.1007/BF00994018

- [116] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang and Chih-Jen Lin (2008). "LIBLINEAR: A library for large linear classification. *Journal of machine learning research*, 9(Aug), pp. 1871-1874. <https://jmlr.csail.mit.edu/papers/v9/fan08a.html>
- [117] M. Arumugam, P. Thangaraj, P. Sivakumar and P. Pradeepkumar (2010). Implementation of two class classifiers for hybrid intrusion detection. 2010 International Conference on Communication and Computational Intelligence (INCOCCI), Erode, pp. 486-490. <https://ieeexplore.ieee.org/document/5738779>
- [118] Dredze Mark, Koby Crammer and Fernando Pereira (2008). Confidence-weighted linear classification. *Proceedings of the 25 th International Conference on Machine Learning*, Helsinki, Finland, pp. 264–271. DOI: 10.1145/1390156.1390190
- [119] Schaffer Cullen (1993). Selecting a classification method by cross-validation. *Machine Learning* 13, pp. 135-143. DOI: 10.1007/BF00993106
- [120] Witten, Ian H., Frank Eibe, Trigg Leonard E., Hall Mark A., Holmes Geoffrey, and Sally Jo Cunningham (1999). *Weka: Practical machine learning tools and techniques with Java implementations*. (Working paper 99/11). Hamilton, New Zealand: University of Waikato, Department of Computer Science. <https://hdl.handle.net/10289/1040>
- [121] Jain Anil K., Flynn Patrick and Arun A. Ross (2007). eds. (eds) *Handbook of Biometrics*. Springer, Boston, MA. DOI : 10.1007/978-0-387-71041-9
- [122] Mark H. Zweig and Gregory Campbell (1993). Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. *Clinical chemistry*, 39(4), pp. 561-577. DOI:10.1093/clinchem/39.4.561
- [123] Yuan Fang, Yan Yan, Si Chen, Hanzi Wang and Chang Shu (2016). Sparse similarity metric learning for kinship verification. 2016 Visual Communications and Image Processing (VCIP), Chengdu, 2016, pp. 1-4, DOI: 10.1109/VCIP.2016.7805462
- [124] Miguel Bordallo López, Elhocine Boutellaa and Abdenour Hadid (2016). Comments on the “Kinship Face in the Wild” Data Sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11), pp. 2342-2344. DOI: 10.1109/TPAMI.2016.2522416
- [125] Ravi Kumar Y B and C N Ravi Kumar (2016). Harmonic rule for measuring the facial similarities among relatives. *Transactions on Machine Learning and Artificial Intelligence*, 4(6), 29-49. DOI:10.14738/TMLAI.46.2221
- [126] Haibin Yan (2017). Kinship verification using neighborhood repulsed correlation metric learning. *Image and Vision Computing*, 60, pp. 91-97. DOI: 10.1016/j.imavis.2016.08.009
- [127] Jiwen Lu, Junlin Hu and Yap-Peng Tan (2017). Discriminative Deep Metric Learning for Face and Kinship Verification. *IEEE Transactions on Image Processing*, 26(9), pp. 4269-4282, DOI: 10.1109/TIP.2017.2717505
- [128] Rachid Aliradi, Abdelkader Belkhir, Abdelmalik Ouamane and Adel S. Elmaghraby (2018). DIEDA: discriminative information based on exponential discriminant analysis

- combined with local features representation for face and kinship verification. *Multimed Tools Appl* (2018). DOI: 10.1007/s11042-017-5572-2
- [129] Amina Tidjani, Abdelmalik Taleb-Ahmed, Djamel Samai, Aiadi Kamal Eddine (2018). Deep learning features for robust facial kinship verification. *IET Image Processing*, 12(12), pp. 2336-2345. DOI: 10.1049/iet-ipr.2018.5552
- [130] Xiuzhuang Zhou, Kai Jin, Min Xu and Guodong Guo (2019). Learning deep compact similarity metric for kinship verification from face images. *Information Fusion*, 48, pp. 84-94. DOI : 10.1016/j.inffus.2018.07.011
- [131] Danula Hettiachchi, Niels van Berkel, Simo Hosio, Miguel Bordallo López, Vassilis Kostakos and Jorge Goncalves (2020). Augmenting Automated Kinship Verification with Targeted Human Input *PACIS 2020 Proceedings*. 141. <https://aisel.aisnet.org/pacis2020/141>
- [132] Wanhua Li, Yingqiang Zhang, Kangchen Lv, Jiwen Lu, Jianjiang Feng and Jie Zhou (2020). Graph-based Kinship Reasoning Network. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, London, United Kingdom, pp. 1-6. DOI: 10.1109/ICME46284.2020.9102823
- [133] Salah Eddine Bekhouche, Abdelhakim Chergui, Abdenour Hadid, and Yassine Ruichek (2020). Kinship verification from gait?. Paper accepted for an oral presentation in the *IEEE International Conference on Image Processing (ICIP)*. <https://cmsworkshops.com/ICIP2020/Papers/ViewPaper.asp?PaperNum=2167>



Abdelhakim Chergui est né le 22/10/1983 à Biskra, Algérie. Il a obtenu le diplôme d'ingénieur d'état en électronique, option Communication de l'Université de Biskra, Algérie, en 2007. Il a entamé le domaine professionnel au sein de l'entreprise nationale de la navigation aérienne ENNA depuis 2008 à ce jour. Après sa réintégration dans le système LMD, il a obtenu en 2015 un diplôme de Master en électronique, option signaux et communication de l'Université de Biskra, Algérie. Ensuite, Il a commencé en 2016 sa formation doctorale, et ce, au niveau de l'université 20 août 1955 – Skikda, cette formation s'est couronnée par la préparation de cette thèse pour l'obtention d'un diplôme de doctorat en électronique option Communication et Traitement de l'Information. Aujourd'hui, il est membre du Laboratoire de Recherche en Electronique de Skikda (LRES) et membre dans le projet de recherche de type P.R.F.U intitulé: Utilisation des Techniques de Traitement d'Images pour la Vérification de la Parenté et la Reconnaissance des Panneaux de Signalisation Routiers (Code : A10N01UN210120180002). Il est le premier auteur de plusieurs publications et communications dans différents Proceedings et revues internationales citées au début de cette thèse. Ses principaux domaines de recherche s'articulent principalement sur le traitement des images faciales et la biométrie, l'intelligence artificielle, la vision par ordinateur.



Salim Ouchtati est né le 08/07/1970 à Azzaba W Skikda Algérie. Il a obtenu ses diplômes d'ingénieur d'état et de Magister en électronique de l'Université d'Annaba, Algérie, respectivement en 1994 et 1999. Il a obtenu son doctorat en automatique en 2007 et son diplôme HDR (Habilitation à Diriger la Recherche) en électronique en 2010 de l'Université d'Annaba. Il est actuellement professeur à l'université de Skikda, membre du Laboratoire de Recherche en Electronique de Skikda (LRES) et responsable du projet de recherche de type P.R.F.U intitulé: Utilisation des Techniques de Traitement d'Images pour la Vérification de la Parenté et la Reconnaissance des Panneaux de Signalisation Routiers (Code : A10N01UN210120180002). Il est l'auteur de plusieurs publications et communications dans différents Proceedings et revues internationales. Ses principaux domaines de recherche s'articulent principalement sur la reconnaissance de l'écriture manuscrite, l'intelligence artificielle, le traitement des images médicales et la biométrie.



Jean Sequeira est diplômé de l'École Polytechnique (1977) et de l'École Nationale Supérieure des Télécommunications (1979), il a obtenu le grade de Docteur Ingénieur (1982) puis celui de Docteur ès Sciences (Doctorat d'État - 1987). Il a été enseignant-chercheur à l'École Nationale Supérieure des Travaux Publics de Yamoussoukro (Côte d'Ivoire –1979-1981). De 1981 à 1991, il a été chercheur, puis chef de projet de recherche, au Centre Scientifique d'IBM France. Au cours de cette période, il a été « détaché » six mois (en 1983) à IBM États-Unis, pour travailler à la conception de la station graphique IBM 5080 (Kingston, État de New-York). En 1991, il a été recruté comme « Professeur des Universités » à l'Université de la Méditerranée (actuellement Aix-Marseille Université). Enfin, il a mis en place un ensemble de collaborations avec institutions internationales (États-Unis, Inde, Portugal, Canada, Pérou, Chine, Algérie) dont certaines sont devenues pérennes. Il a aussi participé à la création (en 2006) et fait partie pendant dix ans du Comité Exécutif d'une organisation internationale, ISDE (International Society for Digital Earth). En 2010, il a été promu « Professeur Classe Exceptionnelle 1 » par le CNU, sur la base de son dossier scientifique. Il est également « Professeur Invité » à l'Académie Chinoise des Sciences et membre « Sénior » de l'association IEEE. Ses thèmes de recherche sont toujours axés sur « l'imagerie numérique », avec une implication dans des domaines fortement connexes tels que l'Intelligence Artificielle, le « Big Data » mais aussi la Réalité Virtuelle, Augmentée et Mixte.