Proposition d'approches utilisant les réseaux de neurones profonds et les méthodes géométriques pour la reconstruction d'un visage 3D à partir d'une seule image

Oussema Bouafif

IMT Lille Douai
École Mines-Télécom
IMT-Université de Lille

Université de Lille

Année 2021

Université Lille

THÈSE

pour obtenir le grade de
DOCTEUR,
spécialité Informatique et Application

par

Oussema Bouafif

Doctorat de l'Université de Lille

Délivré par IMT Lille Douai

Proposition d'approches utilisant les réseaux de neurones profonds et les méthodes géométriques pour la reconstruction d'un visage 3D à partir d'une seule image

Soutenue le 25 juin 2021 devant le jury constitué de :

Président du jury :		
	Olivier Colot	Professeur à l'Université de Lille
Directeur de thèse :		
	Mohamed Daoudi	Professeur à l'IMT Lille Douai
Rapporteurs :		
	Kevin Bailly	Maître de conférences à Sorbonne Université
	Rachid Oulad Haj Thami	Professeur à l'ENSIAS
Examinateurs :		
	Bogdan Khomutenko	CTO à MCQ-Scan
	Catherine Soladié	Maîtresse de conférences à CentraleSupélec

Laboratoire d'accueil :
Centre de **R**echerche en **I**nformatique, **S**ignal et **A**utomatique de **L**ille
UMR CRIStAL - Équipe 3D SAM

Ecole Doctorale SPI 072
(Univ.Lille, Univ.Artois, ULCO, Univ.Polyt.Hauts de Fr., Centrale Lille, IMT Lille Douai)

CRIStAL
Centre de Recherche en Informatique, Signal et Automatique de Lille

MCQ-Scan

Titre : Proposition d'approches utilisant les réseaux de neurones profonds et les méthodes géométriques pour la reconstruction d'un visage 3D à partir d'une seule image.

Résumé : La reconstruction 3D d'un visage à partir d'une image 2D est un problème fondamental de la vision par ordinateur qui suscite un intérêt considérable en raison de ses diverses applications telles que la surveillance, la santé, les jeux-vidéo, le cinéma, etc.

Cette thèse présente deux approches hybrides de reconstruction 3D de visage à partir d'une image couleur 2D qui combinent les techniques d'apprentissage profond et de géométrie. Pour faire face au manque de données nécessaires à l'apprentissage des réseaux de neurones, un générateur de têtes humaines synthétiques a été conçu. Ce qui a permis de constituer une base de données d'images faciales avec plusieurs cartes qui contiennent des informations caractéristiques de la géométrie du visage. Les deux approches de reconstruction de visage 3D utilisent des CNN pour produire deux types de cartes à partir d'une image d'un visage humain. La première approche produit une carte de champ des normales et une carte du module de gradient de la carte de profondeur du visage. Par la suite, ces deux sorties sont utilisées dans un processus d'intégration du champ des normales basée sur les moindres carrées pondérées pour générer une surface faciale 3D. Dans la deuxième approche, le réseau de neurones produit une carte de points de repère et une carte de champ des normales similaires à celle produite dans la première approche. Elles sont utilisées dans un processus de régression qui vise à trouver la meilleure combinaison linéaire des bases d'un modèle paramétrique (3DMM) et d'obtenir ainsi le modèle 3D qui s'ajuste au visage présent dans l'image d'entrée.

Mot clés : reconstruction de visage en 3D, méthodes hybrides, apprentissage profond, données synthétiques, normales de surface faciale.

Title: A proposal of approaches using deep neural networks and geometric methods for 3D face reconstruction from a single image.

Abstract: 3D face reconstruction from a 2D image is a fundamental problem in computer vision that is attracting considerable interest owing to its various potential applications such as surveillance, health, video games, cinema, etc.

This thesis presents two hybrid approaches for 3D face reconstruction from a 2D color image that combine deep learning and geometric techniques. To deal with the lack of data needed to train the neural networks, a 3D synthetic human heads generator is designed. It allowed us to provide a facial image database with several maps that contain facial geometry characteristics for each example. Both 3D face reconstruction approaches use CNN to produce two maps from a human face image. The first approach generates a pixel-wise normal map and an image of the magnitude of the depth gradient. Subsequently, using these maps, the 3D facial geometry is recovered by applying a normals integration process based on weighted least squares method. In the second approach, the neural network produces a landmarks map and a pixel-wise normal map similar to the one produced in the first method. Landmarks are used for pose computation and the initialization of an optimization problem, which in turn, reconstructs the 3D head geometry by using a 3D morphable model (3DMM) and normal vector fields.

Keywords: 3D face reconstruction, hybrid methods, deep learning, synthetic data, facial surface normals.

Remerciements

Je tiens à remercier, à travers cette page, toutes les personnes qui ont contribué, de près ou de loin, à la réalisation et à l'aboutissement de cette thèse.

Les travaux présentés dans cette thèse, ont fait l'objet d'une convention CIFRE entre la start-up *MCQ-Scan* et l'École IMT Lille Douai (Laboratoire : Centre de Recherche en Informatique, Signal et Automatique de Lille (*CRIStAL*)).

Je souhaite en tout premier lieu remercier Rachid Oulad Haj Thami, Professeur à L'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS) de Rabat, ainsi que Kevin Bailly, Maître de conférences à Sorbonne Université, de me faire honneur d'avoir accepté la charge de rapporteur de cette thèse. Je remercie également Catherine Soladié, Maîtresse de conférences à CentraleSupélec, campus de Rennes, et Olivier Colot, Professeur à l'Université de Lille, d'avoir accepté de juger mon travail en tant que membre du jury.

Je tiens à remercier Mohamed Daoudi, mon directeur de thèse pour m'avoir donné l'opportunité de commencer et de poursuivre ce travail de recherche au sein de l'équipe 3D SAM. Sa grande disponibilité, son soutien et ses remarques toujours pertinentes ont été précieuses pour la réussite de ce travail.

Je tiens aussi à remercier Bogdan Khomutenko, mon encadrant à *MCQ-Scan*, pour m'avoir guidé et suivi tout au long de ma thèse, pour ses conseils qui m'ont permis de bien mener ce projet et pour avoir participé au jury de soutenance.

Je remercie également Maurad Amara, CEO de *MCQ-Scan*, sans qui cette thèse CIFRE n'aurait sûrement jamais vu le jour, pour m'avoir bien accueilli et m'avoir permis de mener cette thèse dans de bonnes conditions.

Je n'oublie pas de remercier toutes les personnes incluant amis et collègues que j'ai pu rencontrer au laboratoire *CRIStAL* ainsi qu'à *MCQ-Scan* pour la bonne ambiance dans laquelle j'ai pu m'épanouir dans mon travail, leur bienveillance et leur enthousiasme qui m'ont permis de progresser tout en gardant la motivation.

Mes remerciements iront évidemment à ma famille. Je pense tout d'abord à mes parents qui

m'ont toujours épaulée dans mes études, mes décisions et mes choix, même si je n'ai pas pu être à leurs côtés pendant une longue période. Merci à mes grands-parents, à mon frère et ma sœur et leurs petites familles de m'avoir encouragé continuellement pendant ces trois dernières années d'études.

Je termine par un immense merci à mes amis qui ont toujours été présents au cours de ces trois ans, pour m'encourager et me soutenir.

Résumé

La reconstruction 3D d'un visage à partir d'une image 2D est un problème fondamental de la vision par ordinateur qui suscite un intérêt considérable en raison de ses diverses applications potentielles. Bien que le problème soit facile pour les humains grâce à leurs mécanismes de vision, sa résolution est toujours complexe en raison des énormes changements d'apparence sous différentes prises de vues, des conditions d'illuminations non contrôlées et des occultations. Pour relever ces défis, les nombreuses approches qui ont été proposées, explorent les caractéristiques du visage à partir de l'image d'entrée, ce qui permet de restreindre l'espace des solutions. Cependant, les techniques classiques de reconstruction 3D peuvent parfois échouer, car elles sont sensibles à la qualité de l'image d'entrée et à ses conditions d'acquisition. Pour faire face à tout cela, des approches basées sur les réseaux de neurones ont été proposées. Toutefois, les principales limites de ces techniques reposent sur un manque de contrôlabilité, ainsi que la nécessité d'un large ensemble de données, et d'un temps d'apprentissage relativement long.

Dans le cadre de cette thèse CIFRE effectuée au sein de la start-up *MCQ-Scan*, nous proposons de combiner les techniques d'apprentissage profond et de géométrie pour constituer deux nouvelles méthodes hybrides qui reconstruisent un visage humain 3D à partir d'une image 2D. Pour faire face au manque de données nécessaires à l'apprentissage des réseaux de neurones, nous constituons tout d'abord un générateur de têtes humaines synthétiques 3D. Ceci nous a permis de fournir une base de données d'images faciales avec plusieurs cartes qui contiennent des informations caractéristiques de la géométrie faciale. Notre générateur est composé de différents éléments dont un modèle déformable 3D (3DMM) qui définit la géométrie d'une tête humaine. Afin de synthétiser des exemples réalistes, nous proposons de rajouter d'autres éléments, notamment, des textures de haute qualité, des modèles 3D pour les yeux, les cheveux, et pour les lunettes qui sont utilisées comme accessoires. Pour produire un rendu final de haute qualité, tous ces éléments sont combinés puis restitués grâce à un moteur de rendu graphique 3D. Les deux approches de reconstruction de visage 3D que nous proposons, utilisent des CNN pour produire deux types de cartes à partir d'une image d'un visage humain. La première approche produit une carte de champ des normales et une carte du module du gradient de la carte de profondeur du visage. Par la suite, ces deux sorties sont utilisées dans un processus d'intégration du champ des normales basée sur les moindres carrées pondérées pour générer la surface faciale 3D. Dans la deuxième approche, le réseau de neurones produit une carte de points de repère et une carte de champ

des normales similaire à celle produite dans la première approche. Elles sont utilisées dans un processus de régression qui vise à trouver la meilleure combinaison linéaire des bases d'un modèle déformable (3DMM) et d'obtenir ainsi la géométrie qui correspond le mieux à l'image d'entrée. À la fin de chaque approche, nous évaluons la précision grâce à des tests qualitatifs et quantitatives et nous montrons que les résultats que nous obtenons sont compétitifs avec les méthodes récentes de l'état de l'art. Nous montrons aussi que malgré le fait que nos modèles n'aient été formés qu'à partir de données synthétiques, ils parviennent à récupérer des géométries faciales 3D précises pour des images du monde réel.

Mots-clés : reconstruction de visage en 3D, méthodes hybrides, apprentissage profond, données synthétiques, normales de surface faciale.

Abstract

3D face reconstruction from a 2D image is a fundamental problem in computer vision that is attracting considerable interest owing to its various potential applications. Although the problem is easy for humans with their vision mechanisms, its resolution is always complex due to the enormous changes in appearance under view variations, uncontrolled lighting conditions, and occlusions. To meet these challenges, many approaches have been proposed. They explore facial features from the input image to restrict solutions space. However, classical 3D reconstruction techniques can sometimes fail, as they are sensitive to the input image quality and its acquisition conditions. To cope with all this, neural network-based approaches have been proposed. Nevertheless, the main limitations of these techniques lie in a lack of controllability, as well as the need for a large data set, and a long training time.

In the context of this CIFRE thesis carried out within the *MCQ-Scan* start-up, we propose to combine deep learning and geometry techniques to build two methods that reconstruct a 3D human face from a 2D image. To deal with the lack of data needed to learn our neural networks, we first constitute a 3D synthetic human heads generator. It allowed us to provide a facial image database with several maps that contain facial geometry characteristics for each example. Our generator is composed of different elements including a 3D morphable model (3DMM) that defines the human head geometry. To synthesize realistic examples, we propose to add other models with high-quality textures such as eyes, hair, and some examples of eyeglasses that are used as accessories. To produce a plausible final rendering, all elements are combined and then displayed using a 3D graphics rendering engine. The 3D face reconstruction approaches that we propose, use CNN to produce two maps from a human face image. The first approach generates a pixel-wise normal map and an image of the magnitude of the depth gradient. Subsequently, using these maps, we recover the 3D facial geometry by applying a normals integration process based on weighted least squares method. In the second approach, the neural network produces a landmarks map and a pixel-wise normal map similar to the one produced in the first method. Landmarks are used for pose computation and the initialization of an optimization problem, which in turn, reconstructs the 3D head geometry by using a 3D morphable model (3DMM) and normal vector fields. At the end of each approach, we evaluate the accuracy through qualitative and quantitative tests and show that the obtained results are competitive with recent state-of-the-art methods. We also show that despite the fact that our models were formed entirely from

synthetic data, they are able to recover precise 3D facial geometries for real-world images.

Keywords : 3D face reconstruction, hybrid methods, deep learning, synthetic data, facial surface normals.

Table des matières

Liste des figures

Liste des tableaux

Liste des acronymes

3DMM Modèle Déformable 3D –ou *3D Morphable Model*–

ACP Analyse en Composantes Principales –ou *Principal component analysis (PCA)*–

BFM *Basel Face Model*–

BU-3DFE *Binghamton University 3D Facial Expression*–

CGAN Réseau Antagoniste Génératif Conditionnel –ou *Conditional Generative Adversarial Network*–

CIFRE Conventions Industrielles de Formation par la REcherche

CNN Réseau de Neurones Convolutif –ou *Convolutional Neural Network (ConvNet)*–

CPD *Coherent Point Drift Algorithm*–

EDP Équation aux dérivées partielles

GAN Réseau Antagoniste Génératif –ou *Generative Adversarial Network*–

ICP Point Itératif le Plus Proche –ou *Iterative Closest Point*–

IHM Interactions Homme-Machine –ou *Human-Machine Interactions*–

LSFM *Large Scale Facial Model*–

LYHM *The Liverpool-York Head Model*–

NICP Point Itératif le Plus Proche (Non Rigide) –ou *Nonrigid Iterative Closest Point*–

PBR Rendu Physique Réaliste –ou *Physically Based Rendering*–

PS Stéréo-Photométrie –ou *Photometric stereo*–

RMSE Erreur Quadratique Moyenne –ou *Root-Mean-Square Error*–

RVB Rouge, Vert, Bleu –ou *Red, Green, Blue*–

SFM *Surrey Face Model*–

SFS Forme à Partir de l'Ombrage –ou *Shape-From-Shading*–

Liste des symboles

$\boldsymbol{a} \times \boldsymbol{b}$ $\quad \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ Produit vectoriel entre deux vecteurs

$\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ $\quad \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ Produit scalaire entre deux vecteurs

$\boldsymbol{a} \in \mathbb{R}^2$ Point 2D

$\boldsymbol{p} \in \mathbb{R}^3$ Sommet (point 3D)

$\boldsymbol{n} \in \mathbb{R}^3$ Normale

$N_X \in \mathbb{N}$ Nombre des sommets d'un modèle 3D

$N_y \in \mathbb{N}$ Nombre des paramètres d'un modèle 3DMM

$W \in \mathbb{R}^{3 \times N_X \times N_y}$ Matrice des composantes principales d'un modèle 3DMM

$X \in \mathbb{R}^{3 \times N_X}$ Représentation 3D d'un modèle 3DMM

$X_0 \in \mathbb{R}^{3 \times N_X}$ Forme moyenne d'un modèle 3DMM

$\boldsymbol{y} \in \mathbb{R}^{N_y}$ Paramètres de forme d'un modèle 3DMM

$\Omega \subset \mathbb{R}^2$ Domaine d'une image

$(u, v) \in \mathbb{R}^2$ Coordonnées d'un point image

$I : \Omega \to \mathbb{R}^3$ Image

$G_p : \Omega \to \mathbb{R}^2$ Gradient de profondeur

$\mathcal{N} : \Omega \to \mathbb{R}^3$ Carte de champ des normales

$h : \Omega \to \mathbb{R}$ Carte de profondeur

$\mathcal{W} : \Omega \to \mathbb{R}$ Module du gradient de la carte de profondeur

$\mathcal{Z} : \Omega \to \mathbb{R}^{24}$ Carte des points de repère du visage

K Matrice de projection, contient des paramètres intrinsèques de la caméra

$R \in SO(3)$ Matrice de rotation

$\boldsymbol{t} \in \mathbb{R}^3$ Vecteur de translation

Publications

[A1] **O.Bouafif**, B. Khomutenko, and M. Daoudi, "Monocular 3d head reconstruction via prediction and integration of normal vector field," in *15th International Conference on Computer Vision, Theory and Applications.*, 2020.

[A2] **O.Bouafif**, B. Khomutenko, and M. Daoudi, "Hybrid approach for 3d head reconstruction: Using neural networks and visual geometry," in *25th International Conference on Pattern Recognition (ICPR2020)*, 2020.

Chapitre

1 Introduction

1.1 Motivations et défis

Depuis le début du XXI^{e} siècle, les caméras numériques ont connu un développement important et sont devenues omniprésentes dans notre vie quotidienne. Ils sont intégrés dans les téléphones intelligents, les ordinateurs portables, les tablettes et dans les appareils photos professionnels. Avec ceci, les avancées technologiques en matière de stockage et de transfert des images de haute qualité ont poussé la recherche scientifique à concevoir des systèmes intelligents qui permettent d'analyser et de traiter ces données afin de les exploiter dans de nombreuses applications.

Plus récemment, la communauté de vision par ordinateur a accordé une importance particulière à la conception des outils informatiques pour analyser les visages humains à partir de données visuelles. Cette analyse vise à répondre à plusieurs questions qui peuvent concerner : le genre du sujet, l'âge, l'identité, l'expression faciale (tristesse, joie, peur, etc.), la pose du visage (direction du regard), reconnaissance des changements de la forme du visage dû à des accidents, etc.

Dans ce contexte, plusieurs études ont été consacrées à l'utilisation des modèles de visage 3D qui peuvent surmonter certaines lacunes propres aux images 2D et peuvent donc atteindre des performances de pointe sur certaines applications. En effet, les résultats issus d'une simple caméra 2D ne permettent pas d'exploiter la géométrie d'un visage humain, qui par nature est tridimensionnelle. Par contre, grâce à l'imagerie 3D, il est possible d'avoir une représentation géométrique qui peut être invariable à l'éclairage qui affecte seulement la texture d'un modèle de visage 3D, mais n'a aucun impact sur sa forme qui reste intacte. Un autre facteur qui donne un avantage à la modélisation 3D par rapport au 2D est la variation de la pose (direction du regard). Cette dernière fait l'objet de différentes études pour être déterminée à l'aide des points de repère du visage 2D qui peuvent être mal positionnés dans certains cas d'occlusions. En revanche, en

possédant les formes de visages 3D, la pose est connue.

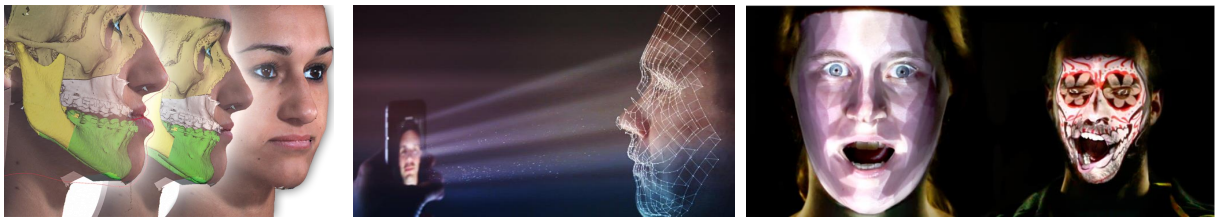

FIGURE 1.1 – Exemples de quelques applications utilisant des modèles 3D de visage : Santé [1], Sécurité [2] et Divertissement [3].

L'analyse du visage humain à l'aide d'un modèle 3D est largement exploitée dans plusieurs domaines différents (voir la figure 1.1), comme : la sécurité [14–17], la santé [18] et le divertissement [13].

— *Sécurité/Surveillance* : Tenant compte des avancés scientifiques, les systèmes de reconnaissance faciale [14–17] sont de plus en plus présents au quotidien alors qu'auparavant, ils étaient considérés comme de la science-fiction. Aujourd'hui, cette technologie est non seulement exploitée par les grandes industries, les aéroports et les services de police, mais aussi elle est présente dans les téléphones, les ordinateurs portables et même dans certaines maisons en vue de prévenir en cas de violence ou de criminalité.

— *Santé physique et mentale* : Plusieurs applications d'analyse faciale 2D/3D ont assisté les professionnelles de la santé dans des problématiques liées aux visages humains pour prendre des décisions ou pour appliquer des traitements. D'une part, ces technologies peuvent prévoir les résultats après des opérations de chirurgie esthétique ou plastique [19]. D'autre part, elles peuvent aussi servir à comprendre l'état psychique d'une personne [20] en effectuant certaines analyses de ces expressions faciales au cours du temps.

— *Divertissement et IHM* : Le besoin d'un modèle 3D de visage humain s'est accru dans plusieurs applications de divertissement ou d'IHM telles que : les jeux vidéo, la réalité virtuelle, le cinéma d'animation et l'impression en 3D. Par exemple, des approches tel que le remplacement de visages (*Morphing*), le transfert d'expressions [21] ou la création d'avatars 3D [22] ont été proposées dans l'industrie du cinéma et des jeux vidéo et même pour la vidéo-conférence. D'autres applications ont enrichi l'expérience d'achat en ligne avec

1. Source : www.dolphinimaging.com
2. Source : www.deccanherald.com
3. C. Siegl, V. Lange, M. Stamminger, F. Bauer, and J. Thies, "Faceforge: Markerless non-rigid face multi-projection mapping," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 11, pp. 2440–2446, 2017

l'essayage virtuel. Dans ce contexte, des techniques de réalité augmenté utilisent un modèle 3D d'un visage pour essayer virtuellement : une monture de lunette [23], une coloration de cheveux, un maquillage [24], etc.

Bien que le modèle 3D faciale présente plusieurs atouts, son obtention est un problème toujours d'actualité. Pour ceci, des solutions matérielles sont de plus en plus disponibles, mais elles sont toujours ignorées par le grand public. Cela s'explique par : leurs complexités d'utilisations dans un environnement de travail typique ou dans un lieu public très fréquenté [25], leurs coûts parfois élevés et aussi la mauvaise qualité de leurs données 3D obtenues qui nécessitent un post-traitement.

Compte tenu de ces facteurs, des techniques de reconstruction de visage 3D à partir d'images 2D ont été proposées et sont devenues de plus en plus répandues. D'abord avec un certain nombre d'approches qui se servent des informations extraites de l'image comme les points de repère faciaux [15, 16, 21], les contours [26], les caractéristiques locales de l'image [27], etc., mais aussi, en utilisant la photométrie [28–34] ou les modèles déformables (3DMM) [35–39]. Quoique ces approches aient permis d'effectuer la reconstruction 3D à partir d'images 2D, certaines difficultés peuvent perturber le processus et produire une forme en 3D inadéquate avec les propriétés du visage présent sur l'image 2D. Ces techniques sont généralement sensibles aux conditions de lumière, réflexions, ombres, qualité de l'image, poses et aussi aux occlusions qui peuvent cacher une partie du visage (cheveux, lunettes, accessoires, etc.).

Récemment, avec l'arrivée des réseaux de neurones et l'apprentissage profond, des alternatives ont été proposées pour résoudre ces problèmes [40–45]. Plusieurs approches qui utilisent ces techniques ont fait leurs preuves pour modéliser implicitement les variations complexes d'éclairage, d'ombres, etc., ce qui conduit à une estimation robuste de la géométrie. En revanche, l'une des difficultés les plus connues dans l'application des réseaux neuronaux est le manque de jeux de données de visages en 3D. Dans certains cas, ce problème rend les systèmes d'apprentissage de bout en bout moins efficaces que les méthodes géométriques. Dans le but de résoudre ce problème, il a été proposé d'utiliser des données synthétiques ou des modèles déformables 3D ajustés, même si ces derniers ne produisent qu'une approximation de la vérité terrain, ce qui peut conduire à une faible précision de reconstruction. Un autre inconvénient qui limite encore plus les approches basées sur l'apprentissage machine de bout en bout est le manque de contrôle sur le processus de reconstruction.

Dans le cadre de cette thèse CIFRE, nous nous intéressons à la problématique de reconstruc-

tion 3D des visages à partir d'image 2D. Pour ceci, nous proposons deux approches hybrides qui sont composées à la fois d'une technique d'apprentissage profond et d'une technique géométrique. Pour compenser le manque des données nécessaires à l'entraînement de nos réseaux de neurones, nous avons mis en œuvre un générateur de tête humaine 3D synthétique composés de différentes parties qui garantit une grande variabilité des données générées. Nous démontrons la capacité de nos systèmes à reconstruire des têtes en 3D à partir d'images du monde réel en réalisant une évaluation quantitative sur une base de données de référence qui possède des images de visages en 2D et leurs géométries correspondantes en 3D ; et une évaluation qualitative en comparant visuellement nos méthodes à d'autres de l'état de l'art où on utilise des images de quelques célébrités. Ces deux évaluations nous permettent aussi d'analyser les lacunes de chacune de nos deux méthodes de reconstruction proposées.

1.2 Cadre de la thèse

Du point de vue applicatif, les travaux de cette thèse s'inscrivent dans un contexte industriel dans le cadre du dispositif national Conventions Industrielles de Formation par la REcherche entre la start-up *MCQ-Scan*[1] et le laboratoire *CRIStAL*[2].

MCQ-Scan est une jeune start-up lilloise spécialisée dans la vision par ordinateur, l'intelligence artificielle et la robotique. Plus spécifiquement, *MCQ-Scan* fournit des solutions innovantes de capture et d'analyse d'images et de vidéo 2D ou 3D. Grâce à son expertise, des informations importantes sont extraites de l'ensemble des données captées et sont par la suite traitées et restituées sous forme de rapport complet qui aide les clients à prendre des décisions appropriées ou à créer des nouveaux services.

1.3 Contributions

La reconstruction faciale 3D à partir d'image 2D est toujours une problématique d'actualité. Dans cette thèse, nous proposons d'autres alternatives à ce problème. Dans ce cadre, nous proposons deux méthodes hybrides qui combinent des approches d'apprentissage profond et des approches géométriques. Le pipeline principal de nos méthodes peut être résumé en deux étapes : 1) Prédiction des caractéristiques géométriques à partir d'une image d'entrée en utilisant un ré-

1. Site-Web : www.mcq-scan.com
2. Centre de Recherche en Informatique, Signal et Automatique de Lille.

seau de neurones entraîné sur des données synthétiques ; 2) Utilisation des sorties du réseau dans une approche géométrique de reconstruction 3D.

Nous résumons les principales contributions de cette thèse comme suit :

Générateur de données synthétiques : Pour former nos réseaux de neurones, nous mettons en œuvre un générateur de têtes humaines synthétiques 3D. Pour le constituer, nous utilisons principalement un modèle paramétrique (3DMM) qui synthétise aléatoirement une géométrie 3D de tête humaine. Avec ceci, nous proposons de rajouter d'autres composants comme des modèles 3D de yeux, de cheveux et de lunettes de vue qui contribuent pour la création d'exemples assez réalistes. En ce qui concerne l'apparence, nous utilisons plusieurs textures de haute qualité qui permettent de donner une allure naturelle et cohérente aux différents composants 3D. En dernier lieu, après alignement des différents éléments, nous utilisons un moteur de rendu graphique 3D pour former des images contenant des visages synthétiques de qualité supérieure. De même, nous générons pour chaque exemple, un ensemble de cartes riches en caractéristiques géométriques qui vont servir au processus de reconstruction 3D de visage. Elles sont aussi utilisées comme vérités de terrain pour la formation de nos réseaux de neurones.

Reconstruction à base d'intégration de champ des normales : Nous proposons une première méthode hybride de reconstruction de surface faciale à partir d'une seule image d'entrée. Au début, nous présentons un réseau neuronal profond basé sur l'architecture proposée par [46] et que nous formons en utilisant des données synthétiques produites par notre générateur de visages humains. Ensuite, ce réseau prédit deux différentes cartes à partir de l'image d'entrée. La première est une carte de champ des normales de la surface du visage et la deuxième contient le module du gradient de la carte de profondeur du visage. Puis, en utilisant ces deux sorties, nous récupérons la géométrie faciale 3D en utilisant une technique d'intégration des normales basée sur les moindres carrés pondérés et qui est à la fois robuste et rapide. Dans ce sens, la deuxième carte estimée fait l'objet de pondération et elle contrôle donc le processus d'intégration afin d'éviter l'apparition d'un biais à proximité des discontinuités de profondeur.

Reconstruction à base d'ajustement de modèle déformable : Contrairement à notre première approche, notre deuxième cadre de reconstruction hybride permet de reconstruire une tête 3D à partir d'une ou de plusieurs images d'entrée. Tout d'abord, nous utilisons

un réseau d'encodeur-décodeur basé sur l'architecture U-net [47]. Tout comme pour le réseau de l'approche précédente, ce réseau est formé avec des données synthétiques. Il nous permet ainsi de prédire à partir d'une image d'entrée, une carte de champ des normales de la surface faciale et une carte de points de repère. Cette dernière est utilisée par la suite pour une estimation grossière de la pose et pour l'initialisation du problème d'optimisation qui, à son tour, reconstruit la géométrie 3D de la tête en utilisant un modèle déformable (3DMM) et la carte de champ des normales estimée. Il s'agit d'un processus d'ajustement du modèle (3DMM) qui permet d'approximer la forme 3D du visage à partir de l'image 2D. Les paramètres d'identités résultant de la convergence du processus d'optimisation permettent d'expliquer la forme faciale obtenue.

Évaluation des méthodes proposées : Grâce à des tests d'évaluation qualitative sur des photos de quelques célébrités et quantitative sur la base de données BU-3DFE [10], nous révélons l'efficacité et la performance de nos deux méthodes proposées. Nous démontrons ainsi leurs compétitivités avec des approches de pointe de l'état de l'art. D'une part, nous évaluons notre première méthode qui prend uniquement une seule image d'entrée. D'autre part, nous analysons la performance de la deuxième méthode pour ses deux configurations à base d'une et de plusieurs images d'entrée. Alors que nos réseaux de neurones ont été formés uniquement sur des données synthétiques, nous montrons comment nos modèles d'apprentissage proposés se généralisent bien aux images du monde réel.

1.4 Organisation du manuscrit

Dans la figure 1.2, nous montrons la structure de notre thèse qui est décrite comme suit :

Chapitre 2 : Dans ce chapitre, nous commençons tout d'abord par une présentation générale de la structure d'un modèle déformable (3DMM) et nous présentons dans ce contexte les modèles les plus connus de l'état de l'art. Ce modèle permet de reconstruire un visage 3D à partir d'une seule image. Ensuite, nous passons en revue les différentes solutions existantes pour résoudre le problème de reconstruction 3D de visage à partir d'image 2D. Nous concentrons notre intérêt sur trois techniques de reconstruction de visage 3D qui sont en relation avec nos méthodes proposées. Plus précisément, nous nous intéressons aux techniques à base de : photométrie, ajustement d'un modèle déformable (3DMM) et réseaux de neurones profonds. Diverses méthodes de reconstruction 3D de visages sont analysées afin de démonter les différences entre eux.

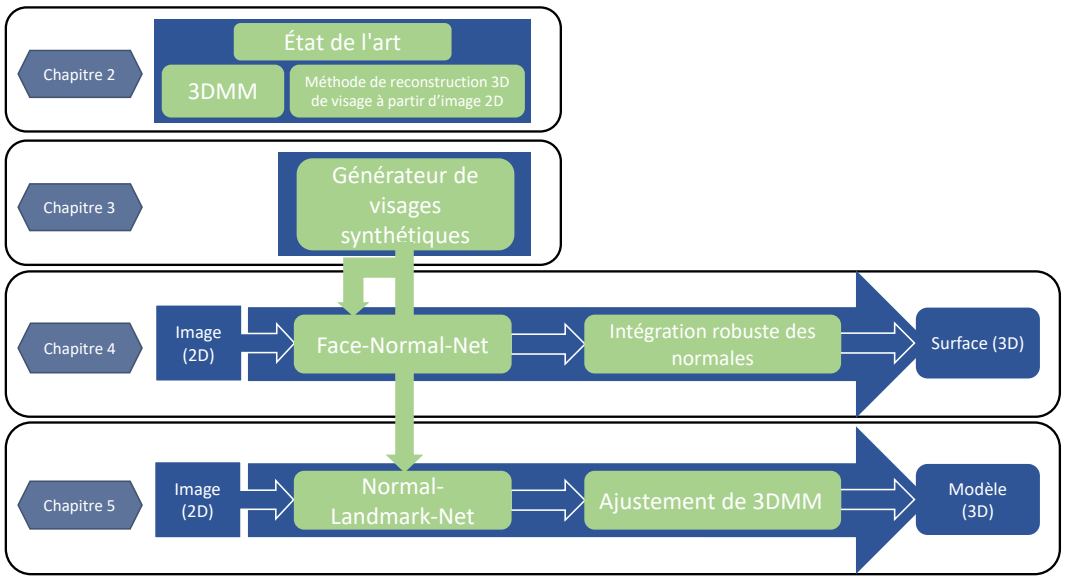

FIGURE 1.2 – Strucutre du manuscrit.

Chapitre 3 : À ce stade, nous présentons l'outil principal que nous avons constitué pour générer les bases de données utilisées pour former nos réseaux de neurones. Après un état de l'art des méthodes qui utilisent des données synthétiques pour l'apprentissage profond, nous décrivons la structure de notre générateur de données synthétiques. Dans ce sens, nous exposons les différents éléments (géométries et textures) qui, une fois traités et alignés, permettent de synthétiser un modèle complet de tête humaine 3D. Pour terminer, nous présentons le processus de rendu final où nous utilisons plusieurs techniques d'affichage dans un moteur graphique 3D qui, à son tour, restitue une quantité illimitée d'images faciales avec différentes cartes riches en informations géométriques.

Chapitre 4 : Ce chapitre est consacré à notre première méthode hybride de reconstruction de visage 3D. Nous décrivons en détail les différentes étapes qui nous permettent d'avoir une surface faciale 3D à partir d'une seule image d'entrée. D'abord, nous présentons l'ensemble d'images utilisées pour former notre réseau de neurones, puis nous décrivons l'architecture de notre modèle d'apprentissage. Ce dernier prend en entrée une image d'un visage humain et produit deux cartes qui contiennent respectivement le champ des normales de la surface du visage et le module du gradient de la carte de profondeur du visage. Par la suite, nous présentons les détails de notre méthode de reconstruction qui utilise une technique d'intégration des normales qui incluent les deux cartes estimées dans une méthode à base des moindres carrés pondérés. Avant de conclure, nous analysons la qualité des images produites par notre réseau, l'efficacité de l'introduction de la deuxième carte pour guider et assurer la robustesse du processus de reconstruction. Enfin,

nous examinons la performance globale de notre méthode de reconstruction 3D de visage face à quelques approches de la littérature.

Chapitre 5 : Dans ce chapitre, la deuxième méthode hybride de reconstruction de visage 3D est décrite en détail. La structure de ce chapitre ressemble à celui de la première méthode présentée dans le chapitre précédent. Dans cette perspective, les différentes phases qui ont constitué notre approche sont abordées, particulièrement, l'ensemble d'apprentissage, le réseau de neurones et notre méthode de reconstruction à base d'ajustement de modèle déformable qui produit une tête 3D avec une morphologie crâniofacial. Par la suite, contrairement à la première méthode, notre analyse porte principalement sur la reconstruction 3D des visages où nous montrons l'efficacité de notre méthode pour les deux configurations à base d'une seule ou de plusieurs images. Pour ce faire, nous effectuons une évaluation expérimentale avec quelques méthodes de pointe de l'état de l'art.

Chapitre 6 : En dernier lieu, la conclusion, les limitations de nos contributions ainsi que les éventuels travaux futurs sont examinés.

Chapitre

2 État de l'art : Reconstruction faciale 3D à partir d'image(s) 2D

2.1 Introduction

La reconstruction de formes 3D à partir d'images 2D est un domaine complexe et problématique. Contrairement au système visuel humain qui peut percevoir la forme 3D à partir d'image 2D simplement en l'observant, la simulation de ces formes dans les machines nécessite l'exploration des méthodes efficaces pour déterminer des caractéristiques de la forme à reconstruire.

Dans le monde numérique, ils existent divers systèmes de reconstruction faciale 3D sophistiqués qui ont à la fois des prix élevés et des portées restreintes. Les plus connues de ces systèmes sont : Les scanners laser 3D [48], les caméras accompagnées de système infrarouge (Kinect [49] par exemple), et finalement les structures stéréovision [25]. Malgré les avancés biens connus dans ce domaine, l'usage répandu des systèmes de capture des visages 3D est encore limité. Ceci est dû aux inconvénients liés à l'état des technologies de numérisation 3D. Parmi ces inconvénients, nous pouvons citer : l'inexactitude des données issues des zones non réfléchissantes du visage, la nécessité d'avoir un environnement contrôlé et un grand investissement pour des machines coûteuses (dans le cas des structures stéréovision et des scanners laser 3D) et l'exigence d'une intervention spéciale pendant le processus de capture et pour une étape de post-traitement.

D'autres techniques de reconstruction 3D ont apparu au cours de la dernière décennie, où de grands efforts ont été investis dans la génération de modèles 3D à partir d'images 2D [42, 50, 51]. Dans ce cadre, plusieurs études approfondies ont comparé et discuté les différentes approches les plus pertinentes dans ce domaine [52–54]. L'avantage de ces techniques est le fait qu'elles soient accessibles à tout le monde grâce à des technologies peu coûteuses et non encombrantes pour

capturer les images 2D. Cependant, et de manière générale, ces techniques rencontrent plusieurs défis liés à l'éclairage, la pose de la tête, la complexité de la géométrie faciale à capturer, etc. Malgré ceci, des progrès remarquables ont été accomplis dans ce sens et ont permis de récupérer même des détails bien spécifiques du visage (par exemple les rides [55]), ou de reconstruire un visage contenant des occlusions ou de capturer les sujets dans des positions difficiles [56]. Les trois grandes stratégies qui ont réussi à reconstruire un visage 3D à partir d'image(s) 2D ou qui ont permis d'extraire des informations préalables à cette reconstruction sont : la photométrie, l'ajustement du modèle déformable et les réseaux de neurones profonds. Dans la première, l'utilisation des méthodes photogrammétriques permet l'estimation des cartes du champ des normales de la surface faciale qui par la suite sont utilisées pour la reconstruction 3D. Pour la deuxième catégorie, plusieurs visages 3D scannées ont permis de construire un modèle statistique qui permettra de produire un visage 3D à partir d'un ajustement à une image d'entrée 2D. En dernier lieu, les systèmes d'apprentissage profond ont permis d'encoder des informations géométriques des visages 3D après un processus d'apprentissage qui utilise une collection de paires faciales 2D-3D.

Dans cette thèse, nous concentrons notre intérêt sur les trois techniques de reconstruction de visage 3D à partir d'une ou de plusieurs images 2D citées ci-dessus. Dans ce contexte, nous proposons deux différentes techniques où nous combinons dans la première les réseaux de neurones avec une technique d'intégration des champs de normales issues des approches photométriques et dans la deuxième les réseaux de neurones et l'ajustement des modèles déformables.

Pour ceci, nous consacrons ce chapitre à l'état de l'art des trois différentes techniques citées ci-dessus. Nous commençons tout d'abord par une présentation générale des modèles déformables de visage 3D et nous présentons dans ce contexte les modèles les plus connues (Section 2.2). Ensuite, nous décrivons les méthodes basées sur la photométrie (Section 2.3.1), puis les méthodes basées sur l'ajustement d'un modèle déformable (Section 2.3.2) et enfin les méthodes à base d'apprentissage en profondeur (Section 2.3.3). Finalement, nous finissons ce chapitre par une conclusion (Section 2.4). Nous illustrons un aperçu des travaux considérées et de leur catégorisation dans la figure 2.3.

2.2 Modèle déformable de visage 3D

L'apparition des modèles de visage déformables 3D a provoqué un développement rapide et remarquable pour la reconstruction des visages 3D à partir d'image(s) 2D. Ces modèles intègrent

des informations préalables présentant sous forme de variations géométriques du visage, et aussi de l'apparence (texture). Ils se constituent d'un visage moyen avec des modes de variation de la géométrie et de la texture. Obtenir un visage 3D en ajustant un modèle déformable 3D à une image 2D est effectué en estimant, en plus des paramètres de la forme et de la texture, les paramètres de la pose et de l'illumination de sorte que la projection dans le plan image du visage 3D résultant produit une image aussi similaire que possible à l'image donnée. Un exemple du processus d'ajustement est illustré dans la figure 2.1.

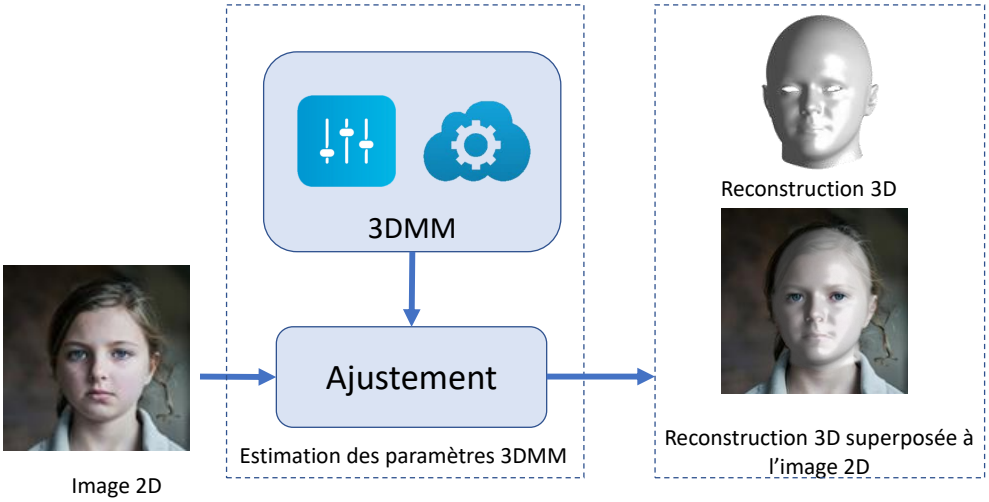

FIGURE 2.1 – Visualisation d'un exemple du processus de reconstruction 3D d'un visage à partir d'une image 2D en utilisant la technique d'ajustement 3DMM.

2.2.1 Terminologie

Généralement connu sous l'abréviation (3DMM) pour Modèle Déformable 3D –ou *3D Morphable Model*–, il a été présenté pour la première fois par Blanz et Vetter [1]. Il est construit en utilisant la méthode d'analyse en composantes principales (ACP) qui est souvent connue pour être utilisée dans la réduction de dimensionnalité. Mais dans ce contexte, elle permet de créer un modèle génératif paramétrable par un ensemble de coefficients non corrélés.

Afin de créer un 3DMM, il faut tout d'abord, un ensemble d'exemples de visages 3D alignés, où tous les sommets sont en correspondance point à point densément. Puis en utilisant l'ACP, une compression des données de l'ensemble d'apprentissage est réalisée grâce à une transformation de base qui produit un système de coordonnées orthogonales défini par des vecteurs propres et des valeurs propres. Au final, ceci permet d'avoir deux modèles : un pour la forme géométrique et un pour la texture (aussi appelé albédo).

Un visage 3D est représenté par un maillage constitué de N_X sommets, où pour chaque sommet $\boldsymbol{p}_i = [x_i, y_i, z_i]^\intercal \in \mathbb{R}^3$, une couleur $\boldsymbol{c}_i = [r_i, v_i, b_i] \in [0,1]^3$ qui contient les valeurs r_i (rouge), v_i (vert) et b_i (bleu) est associée. Les deux vecteurs qui représentent la géométrie et la texture de ce modèle sont :

$$\begin{aligned} X &= [\boldsymbol{p}_1^\intercal, \boldsymbol{p}_2^\intercal, ..., \boldsymbol{p}_{N_X}^\intercal]^\intercal \\ C &= [\boldsymbol{c}_1^\intercal, \boldsymbol{c}_2^\intercal, ..., \boldsymbol{c}_{N_X}^\intercal]^\intercal \end{aligned} \tag{2.1}$$

Cependant, en supposant que le modèle déformable est constitué d'une collection de visages 3D assez variés. Nous appliquons l'ACP pour obtenir un système de coordonnées orthogonales défini par les vecteurs propres W_i de la matrice de covariance calculé sur les formes de l'ensemble d'apprentissage. Ainsi, nous pouvons produire la géométrie d'un nouveau sujet, tout en combinant linéairement les composantes de cet ensemble. La formulation de cette combinaison est décrite comme ceci :

$$X = X_0 + \sum_{i=1}^{N_y} \boldsymbol{y}_i W_i = X_0 + W\boldsymbol{y} \tag{2.2}$$

où $X \in \mathbb{R}^{3\times N_X}$ représente les sommets générés pour un visage 3D ; $X_0 \in \mathbb{R}^{3\times N_X}$ est la forme moyenne du modèle (3DMM) calculé sur l'ensemble des scans faciaux alignés et représentées par les sommets 3D des nuages de points ; $\boldsymbol{y} = [\boldsymbol{y}_1, .., \boldsymbol{y}_{N_y}] \in \mathbb{R}^{N_y}$ est le vecteur des paramètres du modèle et enfin, $W = [W_1, .., W_{N_y}] \in \mathbb{R}^{3\times N_X \times N_y}$ est la matrice qui représente les composantes principales du modèle. Nous soulignons que pour constituer un maillage à partir d'un modèle 3DMM, un ensemble de triangles (*faces*) est obtenu après l'étape d'alignement des exemples de la base d'apprentissage. Ces triangles permettent de connecter les différents sommets entre eux.

2.2.2 État de l'art

Durant les dernières décennies, plusieurs modèles déformables sont apparus. Le premier était celui de Blanz et Vetter [1], construit avec 200 modèles de visages scannés (100 hommes et 100 femmes) avec une expression faciale neutre. Durant cette construction, l'algorithme de flux optique a été adopté pour établir une correspondance pixel par pixel dans l'espace UV de l'ensemble d'apprentissage. Pour ceci, une cartographie bijective associe chaque visage 3D à un paramétrage cylindrique et par conséquent une correspondance dense entre deux représentations UV produit ainsi une correspondance dense entre les deux nuages de points 3D.

Ensuite, Paysan *et al.* [2] ont proposé le modèle connu sous le nom de *Basel Face Model–* (BFM). Ce modèle est aussi construit avec 200 sujets (100 hommes et 100 femmes), mais un scanner plus performant a été utilisé pendant le processus de scan. Il s'agit d'un système d'éclairage codé qui a permis de reconstruire la forme d'un visage à l'aide d'une séquence de motifs lumineux. Tout cela a permis de produire des modèles de haute précision avec un temps de capture plus court. Le modèle BFM est le premier modèle mis à disposition pour l'ensemble de la communauté. En vue de rajouter plus de flexibilité à l'ensemble du modèle déformable, *FaceWarehouse* été proposé par [6]. Outre les variations géométriques de la forme globale du maillage, ce modèle comprend aussi des variations d'expressions faciales. Le processus de construction de ce modèle est comme suit : un système de scan de type *Kinect* a été utilisé pour capturer 20 expressions de 150 sujets. Ensuite, un ensemble de points repère sur l'image de chaque sujet a été localisé. Cet ensemble a servi pour ajuster le modèle déformable [1] à la carte de profondeur, pour obtenir une correspondance entre l'ensemble des maillages des sujets disponibles. Finalement, à partir de ces maillages, des formes d'expression ont été générés, résultant en 47 expressions faciales obtenues pour chaque sujet, et qui sont capables de représenter la plupart des expressions faciales humaines.

Surrey Face Model– (SFM) est un autre modèle déformable qui a été proposé par [3], où un ensemble de 169 sujets ont été scannées et qui ont une grande diversité en ethnie et en âge. Pour le recalage des sujets, une méthode itérative [57] composée de trois étapes [55] a été adoptée. Elle permet d'aligner deux visages 3D de manière grossière à fine. Afin de créer des modèles 3D texturées, l'ensemble de photos prises pendant la capture de chaque visage ont été mappées à chaque scan 3D enregistrée. Ce modèle déformable est disponible en multi-résolution accompagnée d'un logiciel d'ajustement qui permet d'adapter le SFM à des nouvelles images et des vidéos.

Plus récemment, le plus grand modèle déformable (3DMM) connue sous le nom de *Large Scale Facial Model–* (LSFM) a été proposé par [4]. Le modèle est construit à partir de 9663 personnes couvrant une grande variété d'ethnie et d'âge. Sa construction a été établie selon un pipeline automatisé, où on détecte tout d'abord des points de repères 2D à partir de chaque image rendue pour tous les maillages de l'ensemble de la base, puis ces points de repères 2D sont mappées sur le modèle 3D. Ensuite, une correspondance dense entre les sujets est réalisée en utilisant l'algorithme NICP [58]. Ainsi, le modèle final est obtenu en appliquant l'ACP sur l'ensemble des sujets de la base d'apprentissage après avoir exclu les valeurs aberrantes.

Un pipeline similaire à celui de [4], a été proposé par [5] pour présenter le *The Liverpool-York Head Model*– (LYHM) qui comporte 1212 sujets équilibrés en genre et diversifié en âge. Les deux plus grandes différences dans le pipeline de construction de ce modèle sont : la détection de points de repère 2D qui s'est fait directement sur les images issues du scanner utilisé pendant l'étape de capture des sujets, plutôt que d'utiliser les images rendues à partir de chaque maillage. Ensuite, au lieu d'utiliser la méthode NICP [58] pour établir la correspondance dense entre les modèles, une autre approche basée sur l'algorithme CPD [59] a été utilisé. Une spécificité importante de ce modèle est le fait que sa géométrie comporte non seulement la partie faciale comme les modèles cités précédemment, mais aussi la partie du crâne ce qui lui permet d'être un modèle déformable qui représente une tête humaine entière. Plus de détails qui concernent les modèles faciales statistiques ont été décrites dans [52].

Nous illustrons dans la figure 2.2 les différents modèles décrits précédemment.

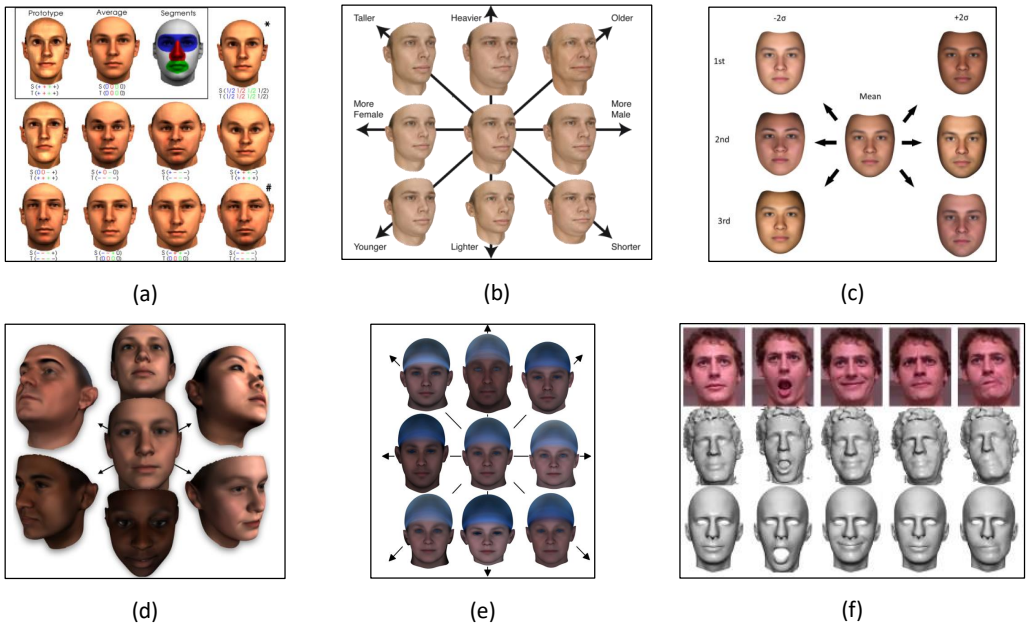

FIGURE 2.2 – Exemples de modèles déformables 3D : (a) Modèle de Blanz et Vetter [1] (b) BFM [2] (c) SFM [3] (d) LSFM [4] (e) LYHM [5] (f) FaceWarehouse [6]

2.3 Méthodes de reconstruction faciale monoculaire 3D

2.3.1 Méthodes à base de photométrie

La récupération de la forme 3D en utilisant les méthodes photométriques est un thème classique de la vision par ordinateur. L'idée principale derrière cette classe de méthodes repose sur

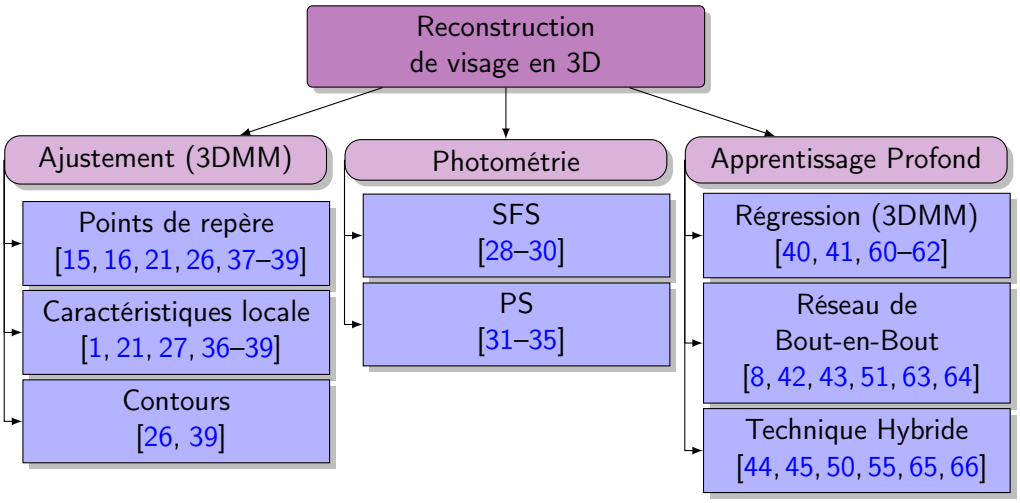

FIGURE 2.3 – État de l'art des méthodes de reconstruction 3D de visage selon notre classification. La récupération de la géométrie faciale 3D avec image 2D peut être faite en utilisant : l'ajustement d'un modèle déformable 3DMM, les techniques photométriques ou les réseaux de neurones profonds.

l'analyse de la composition d'une ou de plusieurs images. L'objectif derrière ces techniques est de reconstruire la forme tridimensionnelle d'un objet en se basant sur l'hypothèse de formation d'image (en général : l'hypothèse de réflectance lambertienne). Les approches de reconstruction à base de méthode photométrique peuvent être divisées en deux catégories selon le nombre d'images utilisées. La première technique est appelée Shape from Shading (SFS), et elle prend en compte une seule image d'entrée dans le processus de reconstruction [67,68]. Plus tard, une extension de cette technique est apparue sous le nom de Photometric Stereo (PS) [69,70]. Elle utilise plusieurs images du même sujet, prises sous le même angle, mais avec des sources d'éclairages différentes.

La formulation mathématique de ce problème est basée sur l'utilisation de l'équation d'irradiance d'image dans le cas d'une seule image d'entrée pour les méthodes SFS ou dans le cas de plusieurs images pour les méthodes PS. Cette formulation peut être décrite comme suit :

$$I^i = \mathfrak{R}(\mathcal{N}, s^i, \rho), \qquad i \in \{1, ..., m\} \tag{2.3}$$

Avec I est l'image d'entrée prise par l'appareil photo ; $\mathfrak{R}$ est la carte de réflectance [69] ; ρ représente les paramètres de réflectance (l'albédo par exemple) ; s représente le vecteur de la source lumineuse et $\mathcal{N}$ est la carte de champ des normales. Comme nous l'avons expliqué ci-dessus, le problème fait référence à la technique de PS si, $i \geq 2$.

Dans la plupart des cas, l'idée derrière cette formulation est de retrouver la carte de champ des normales $\mathcal{N}$ qui par la suite doit être intégré dans une seconde étape afin de retrouver la

carte de profondeur du sujet présent dans l'image. Plusieurs techniques d'intégration des champs de normales ont été discutées dans [71, 72]. Dans la suite de cette section, nous allons passer en revue les différentes techniques de photométrie qui vise à reconstruire un visage en 3D. À cet égard, nous nous focalisons au début sur les techniques qui utilisent une seule image d'entrée (SFS) puis sur les méthodes à base de plusieurs images (PS).

2.3.1.1 Méthodes monoculaires

Étant donné que la reconstruction du visage en 3D à partir d'une seule image 2D est un problème mal posé, des connaissances préalables sont nécessaires pour limiter l'espace des solutions.

Les auteurs de [28] ont utilisé une forme 3D d'un visage comme connaissance préalable pour effectuer la reconstruction donnée à partir d'une seule image. Le processus de reconstruction consiste tout d'abord à récupérer les coefficients d'éclairages en ajustant la forme du visage de référence à l'image d'entrée en utilisant l'équation (2.3). Ensuite, les auteurs ont estimé la carte de profondeur grâce au champ des normales et enfin, ils ont récupéré l'albédo en utilisant les coefficients d'éclairage et la carte de profondeur déjà estimée.

Dans [29], une combinaison réussie de deux approches a été proposée pour résoudre le problème de reconstruction en 3D à partir d'une seule image d'un visage humain capturé sous un éclairage directionnel connu. La première approche utilise la symétrie de visage [73] et la deuxième utilise les statistiques des visages humains [74]. L'algorithme que les auteurs ont proposé, fournit une solution de forme fermée qui satisfait à la fois la symétrie et les contraintes statistiques en résolvant un système d'équations des moindres carrés. Bien que cette configuration tienne compte de l'albédo non uniforme du visage et possède une faible complexité de calcul, les résultats produits sont inexacts pour certains visages et ceci est dû à la dissymétrie faciale de ces exemples.

Différemment à ce qui a été proposé dans les approches citées ci-dessus, les auteurs de [30] ont utilisé les méthodes de SFS pour affiner un visage grossier qui était estimé par ajustement d'un modèle déformable 3DMM (Section 2.3.2). Pendant une première étape d'initialisation, les paramètres de forme 3DMM, les coefficients d'expression et de textures, les paramètres de caméra et d'illumination sont estimés au niveau grossier. Puis, pendant une deuxième étape de correction, la géométrie du visage est corrigée pour avoir une plus grande ressemblance avec l'image d'entrée. Cette correction est effectuée en appliquant la méthode de déformation de [75] pour obtenir des positions optimales des sommets du maillage avec les points de repère détectés. Finalement, une

technique de SFS est appliquée en utilisant une décomposition de l'image d'entrée similaire à celle décrite dans l'équation (2.3). Ainsi, ils obtiennent une carte de profondeur qui est utilisée pour relever des détails fins du visage reconstruit.

2.3.1.2 Méthodes multi-images

Le problème de reconstruction à partir de plusieurs images a été introduit à l'origine par Woodham dans [69]. L'idée derrière cette approche est d'estimer le champ des normales de la surface à partir de plusieurs images prises sous différentes conditions d'éclairage et pour une pose de caméra fixe.

Les auteurs de [31] ont proposé une méthode de reconstruction de visage en 3D à partir de plusieurs images. Leur idée principale est de décomposer une matrice contenant les images (sous forme de vecteurs) en une matrice des coefficients de l'éclairage et une matrice qui contient à la fois les normales de la surface et l'albédo. L'estimation initiale de ces matrices a été faite en utilisant une décomposition en valeurs singulières. Enfin, dans un processus itératif, ces matrices sont optimisées et la forme 3D est reconstruite grâce à l'intégration des normales.

Suwajanakorn *et al.* [32] ont proposé de reconstruire une forme en 3D pour chacune des images issues d'une séquence vidéo. Pour créer une forme 3D de référence, ils ont utilisé la technique proposée par [31]. Puis, ils ont déformé la forme obtenue pour qu'elle corresponde à chaque image de la vidéo en utilisant un algorithme de flux optique 3D qui calcule des correspondances entre la forme 3D personnalisée et chaque image de la vidéo. Ainsi, le maillage est déformé pour correspondre à l'image donnée avant d'être affiné en minimisant une erreur photométrique.

Une méthode de reconstruction 3D de visage sans contrainte a été proposée dans [33]. La méthode est appelée sans contrainte parce que les images d'entrée peuvent être prises sous différentes conditions d'éclairages et sous différentes poses de caméra. Au départ, une déformation de la forme 3D est réalisée en utilisant des points de repère 2D détectés sur chaque image. Puis, à l'aide de la même méthode décrite dans [31], la matrice qui regroupe les images est décomposée en deux matrices pour les coefficients d'éclairage et pour les normales de surface et l'albédo. Une amélioration de cette méthode a été proposée dans [35] en proposant de rajouter une étape d'ajustement en utilisant un modèle déformable 3DMM (Section 2.3.2).

Dernièrement, Cao *et al.* [34] ont proposé aussi d'utiliser une technique d'ajustement d'un modèle déformable comme étape initiale. Pour ce faire, un ensemble d'images prises sous différentes sources d'éclairages a été utilisé. Suite à cet ajustement, ils ont obtenu, un visage en 3D

accompagné de son expression et de sa pose. À partir de ce modèle, ils déduisent la carte des normales avant d'estimer la position et la quantité de lumière de toutes les sources d'éclairages dans une procédure de calibrage. Ensuite, une carte des normales à haute résolution est calculée en sélectionnant trois lumières incidentes fiables. La géométrie faciale est enfin obtenue après avoir intégré la carte de champ des normales.

2.3.2 Ajustement d'un modèle déformable

La reconstruction de la géométrie 3D d'un visage humain à partir d'une ou de plusieurs images a été abordée par plusieurs chercheurs qui utilisent des techniques d'ajustement de modèle déformable (3DMM). Plusieurs caractéristiques qui sont extraites des images faciales d'entrées peuvent servir au processus d'estimation des paramètres (3DMM). Parmi ses caractéristiques, nous pouvons citer : les points de repère [15, 16, 21] qui décrivent la forme du visage et les caractéristiques locales de l'image tel que les valeurs brutes des pixels [27] ou les contours [26]. Le processus d'ajustement qui utilise ses caractéristiques est souvent intensif en termes de calcul. Il repose sur une procédure d'optimisation qui peut donner un minimum local plutôt que global et qui dépend d'une bonne initialisation.

Bien que la plupart des approches utilisent une seule image d'entrée pour reconstruire la géométrie 3D, certains chercheurs ont considéré que l'utilisation de plusieurs images peut améliorer la précision de la reconstruction d'où le fait d'observer un visage à partir de plusieurs vues ou dans différentes conditions d'illumination.

Parmi les approches qui utilisent plusieurs images d'entrées, les auteurs de [35] ont proposé un processus sur trois étapes. D'abord, une détection des points de repère est effectuée sur l'ensemble des photos collectées. Puis, un ajustement grossier du modèle déformable est effectué en utilisant ces points de repère. Ensuite, dans un processus itératif, deux étapes de reconstruction des détails fins sont alternées pour chaque image de l'ensemble jusqu'à un seuil bien déterminé. Au début, une estimation des champs de normales est effectuée en utilisant une approche photométrique. Puis, les détails sont reconstruits en utilisant les champs de normales estimés.

Dans [36], l'idée de base pour relever la géométrie finale est d'effectuer des reconstructions séparées à partir de chaque image d'entrée pour un seul sujet et puis de combiner les meilleurs de toutes les reconstructions dans une forme finale. L'élément clé de cet algorithme est la mesure de la qualité des reconstructions 3D, basée sur la distance entre les normales des surfaces du visage reconstruit et ceux du visage moyen. Les auteurs ont montré que cette mesure surpasse d'autres

critères comme la distance de Mahalanobis ou la distance euclidienne.

Contrairement à ce qui a été proposé dans [36], les auteurs de [21] ont proposé des approches qui estiment les paramètres à partir de toutes les images d'entrée d'un seul sujet en même temps. La minimisation d'une fonction coût est effectuée à partir de toutes les images en gardant un vecteur unique pour les paramètres de forme, mais en estimant les autres paramètres séparément (de projection par exemple) pour chaque image.

La première technique d'ajustement d'un modèle déformable à partir d'une seule image d'entrée a été proposée par Blanz et Vetter [1]. L'idée de leur technique est de minimiser l'erreur entre l'image d'entrée et l'image rendue simulée avec le modèle de réflectance de Phong [76]. Le terme de minimisation comprend la géométrie, la texture, les paramètres de projection, les paramètres intrinsèques de la caméra et les paramètres d'éclairage. Ces paramètres sont ajustés de manière itérative jusqu'à ce qu'une correspondance optimale soit obtenue.

La localisation automatique des points de repère du visage en 2D a permis à des chercheurs de développer des approches de reconstruction 3D de visage dans un processus complètement automatisé. L'utilisation de ces points de repère rend ces algorithmes plus robustes aux changements d'éclairage et de texture, car ils ne dépendent pas directement de l'image. Cependant, la précision de détection des points est essentielle.

Par exemple, dans [15], les auteurs se servent de la reconstruction 3D de visage pour une approche de reconnaissance faciale. Pour ce faire, ils utilisent la même procédure décrite dans [1], mais il rajoute à la fonction coût un terme à minimiser. Il s'agit de la distance entre l'ensemble des points de repère de la forme faciale reconstruite projetée et l'ensemble des points de repère 2D détectés directement à partir de l'image d'entrée.

Dans [16], les auteurs ont remarqué que les paramètres de projection et les points de repère sont corrélés, pour cela, ils ont proposé un processus itératif pour estimer ces paramètres et mettre à jour la position des points de repère sur le modèle 3D à chaque itération.

Dans d'autres approches, la reconstruction 3D d'un visage est faite sur différentes étapes. Dans [37], l'estimation des paramètres de forme, d'expression et de projection a été le cœur d'une première étape globale. Ensuite, une étape de régularisation qui adopte le modèle de réflectance Lambertien a été proposée pour ajouter des détails (rides) au visage résultant.

Pour Jin *et al.* [38], une phase de pré-alignement est proposée pour estimer les paramètres de projection à l'aide d'un modèle de régression. Ensuite, une première étape globale d'optimisation est effectuée en prenant en compte le terme de minimisation des points de repère et un terme

de régularisation. Enfin, pour accentuer les détails, ils ont minimisé la différence entre l'image d'entrée et l'image reconstruite, en ajoutant un terme de contrôle qui stabilise les paramètres de forme pour qu'ils soient similaires à ceux estimés dans la phase globale et pour que la forme obtenue ne dérive pas de la forme naturelle d'un visage humain.

Bas *et al.* [26] ont proposé une méthode entièrement automatique qui estime d'abord les paramètres de forme et de projection avec des solutions analytiques. Puis une deuxième partie qui raffine les paramètres de forme est effectuée de façon non linéaire, similairement aux autres approches. Cette approche inclut non seulement les points de repère fourni par le détecteur, mais aussi des points qui se situent sur le bord du visage du côté de l'image et qui correspondent aux sommets projetés qui se trouvent sur la limite du modèle pré-aligné.

D'un autre côté, des techniques pour reconstruire un modèle facial 3D en plusieurs étapes qui consistent à séparer la géométrie et l'apparence ont été étudiées. Par exemple, Hu *et al.* [39] ont commencé la reconstruction par une étape géométrique où ils estiment les paramètres de projection puis les paramètres de forme et enfin, ils ont rajouté des points de contour du visage pour affiner les paramètres déjà estimés dans une autre itération. Dans la seconde phase d'estimation de la texture, ils ont utilisé le modèle de réflectance de Phong [76]. Cette estimation comprend la direction et l'intensité pour la lumière et les paramètres de texture du modèle déformable (3DMM).

2.3.3 Méthodes à base d'apprentissage profond

Au cours des dernières années, les réseaux de neurones profonds ont démontré de meilleures performances pour la résolution de plusieurs tâches de vision par ordinateur. Parmi ces tâches, plusieurs approches ont été proposées pour résoudre le problème de reconstruction 3D de visage à partir d'image 2D. Le but principal de ces techniques est de fournir la géométrie 3D la plus fidèle possible au sujet présent dans une image d'entrée 2D en encodant les connaissances préalables dans les poids d'un réseau formé. Dans ce contexte, la combinaison de plusieurs critères pourrait être envisagée pour former un réseau complet, notamment l'architecture, la fonction coût et la technique établie pour former le réseau.

Dans cette section, nous présentons les travaux les plus pertinents en matière de reconstruction de visages en 3D qui utilisent l'apprentissage approfondi comme outil principal. Pour ceci, nous séparons les techniques d'apprentissage en trois catégories : tout d'abord, nous commençons par les techniques qui régressent directement les paramètres d'identités 3DMM à partir d'une

image 2D, ensuite nous présentons les techniques d'apprentissage de bout en bout et enfin nous finissons par les techniques hybrides qui sont composées de réseaux de neurones et de techniques géométriques.

2.3.3.1 Régression des paramètres 3DMM

L'idée de produire directement les paramètres de forme 3DMM à partir d'une image 2D en utilisant l'apprentissage profond a été adressée par plusieurs chercheurs. Cette application est représentée par $f_\theta : I \rightarrow \boldsymbol{y}$ où $I \in \mathbb{R}^{H \times W \times 3}$ est l'image faciale d'entrée (H et W représentent respectivement l'hauteur et la largeur de l'image) et $\boldsymbol{y} \in \mathbb{R}^{N_y}$ représente les paramètres du modèle 3DMM obtenus. Ces paramètres sont par la suite utilisés pour reconstruire un modèle 3D de visage en utilisant (2.2).

Dans ce contexte, Tran *et al.* [40] ont utilisé un ensemble de données constitué d'images faciales et de ces paramètres 3DMM correspondants dans le but d'entraîner un réseau qui régresse les paramètres de forme et de texture à partir d'une seule photo d'entrée. L'architecture adoptée est un ResNet [77] avec 101 couches qui a déjà été utilisée pour la reconnaissance faciale [78]. Afin de l'adapter à la tâche de régression des paramètres du modèle déformable, les auteurs ont modifié la dernière couche pour produire le vecteur de caractéristiques 3DMM.

En suivant la même direction, Dou *et al.* [41] ont proposé *UH-E2FAR*, une méthode de reconstruction faciale 3D basée sur les réseaux neuronaux profonds. Ils ont introduit deux éléments clés à leur architecture, à savoir un réseau neuronal convolutif (CNN) de fusion et une fonction coût multitâche pour améliorer la reconstruction des expressions faciales. Avec ces deux composants, ils ont divisé la reconstruction faciale 3D en deux sous-tâches — la prédiction de la forme neutre du visage en 3D et les paramètres d'expression d'un 3DMM — en utilisant une seule image frontale pour chaque sujet.

Dans [60], Yi *et al.* ont proposé *MMFace*, qui est un réseau de régression multimétrique composé de deux sous-réseaux : un sous-réseau volumétrique pour estimer une géométrie de visage intermédiaire et un sous-réseau paramétrique pour déduire les paramètres 3DMM correspondants.

Dans [61], la même technique a été aussi utilisée pour régresser les paramètres 3DMM à partir de trois images du même sujet prises dans différentes vues dans. L'idée principale derrière leur réseau est d'apprendre les caractéristiques de chaque image d'entrée. Elles sont ensuite traitées par deux couches séparées et entièrement connectées : l'une d'elles estime les paramètres de projection de chaque image et l'autre estime les paramètres de forme et d'expression à partir de

la concaténation de toutes les caractéristiques extraites par le réseau. Enfin, un maillage coloré est obtenu en récupérant la texture du visage à partir des images d'entrée.

Sanyal *et al.* [7] ont proposé RingNet un réseau de neurones qui estime les paramètres d'un modèle déformable à partir d'une image 2D. Pendant l'apprentissage, ils ont appliqué une fonction coût qui encourage la forme d'un visage à être similaire à un autre lorsque l'identité est la même et différente s'il s'agit d'une autre personne. Plus précisément, cette technique renforce explicitement la cohérence de l'identité entre les paramètres de forme estimés à partir de plusieurs images de la même personne et vice-versa.

Deng *et al.* [9] ont utilisé une approche avec un cadre d'apprentissage multi-image qui ressemble à ce qui a été proposé dans [7]. En plus de la régression des paramètres de forme, les auteurs estiment aussi les coefficients de texture, de pose, d'expressions et d'illumination. Pour la formation de ce réseau, ils ont proposé un schéma non supervisé qui intègre deux fonctions à différents niveaux : image et perception. De plus, ils ont entrainé un réseau auxiliaire simple qui prédit des « scores de confiance » pour les coefficients du modèle 3D régressé portant l'identité et ils obtiennent les coefficients d'identité finales via une agrégation basée sur la confiance. Bien qu'aucune étiquette de confiance explicite ne soit utilisée comme vérité terrain, cette méthode apprend automatiquement à favoriser les photos de haute qualité.

Au cours des dernières années, les réseaux antagonistes génératifs (GAN) ont suscité un grand intérêt et plusieurs chercheurs ont souligné leur potentiel. Plusieurs avancées ont été réalisées grâce à ce type de réseau dans différentes applications telles que la synthèse d'images [79], la transformation d'image en images [80] et la super-résolution [81]. Une architecture GAN est constituée de deux réseaux de neurones qui sont mis en compétition. Le premier, appelé générateur G, vise à générer des échantillons, tandis que le deuxième appelé discriminateur D, tente de détecter si cet échantillon provient des données réelles ou s'il s'agit d'une création de son « adversaire » le générateur. En d'autres termes, G capture la distribution de données en prenant un vecteur latent z échantillonné à partir d'une distribution antérieure $p_z(z)$ et transforme ce vecteur en une image $\hat{x} = G(z)$. Cette image $\hat{x}$ est considérée comme un échantillon de la distribution générative apprise p_G. Juste après, le discriminateur D essai de faire la distinction entre $\hat{x}$ qui fait partie des fausses données et x qui est échantillonné à partir de la distribution de données réelles $p_{data}(x)$. En conséquence, le mode d'entraînement est antagoniste, c'est-à-dire que D et G vont entrer dans un jeu de minimax. D est entraîné à maximiser la probabilité d'attribuer l'étiquette correcte à la fois aux échantillons réels et à ceux générés alors que G est

formé pour créer des images inédites. La fonction objective d'un GAN est comme ceci :

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[log\ D(x)] + \mathbb{E}_{z\sim p_z}[log\ (1 - D(G(z)))] \tag{2.4}$$

ou $V(D,G)$ désigne la fonction coût antagoniste pour G et D, et $D(x)$ désigne la probabilité que x soit réel. Une version améliorée de cette fonction objective a été proposée par [82]. Elle consiste à maximiser $log\ (D(G(z)))$ au lieu de minimiser $log\ (1 - D(G(z)))$ pour assurer une meilleure formation de G. L'idée de cette amélioration vient du fait qu'au début de l'apprentissage, lorsque G ne génère que des images de mauvaise qualité, D a tendance à attribuer une faible probabilité, ce qui sature $log\ (1 - D(G(z)))$ et ne peut fournir un gradient suffisant pour mettre à jour G et provoque donc la disparition du gradient (*gradient vanishing*). Plus tard, plusieurs chercheurs ont aussi proposé d'autres approches pour améliorer la formation des GAN. Parmi ces approches, une variante appelée Wasserstein GAN (WGAN) [83] s'est montrée plus stable pendant l'entraînement. L'idée principale de WGAN est le fait qu'il optimise la distance de Wasserstein en utilisant la dualité Kantorovich Rubinstein, au lieu d'optimiser la divergence Jensen-Shannon dans le GAN standard. Une description plus détaillée des variantes des modèles génératifs GAN est décrite dans [84–86]. Avec cette nouvelle approche, Tu *et al.* [62] ont utilisé

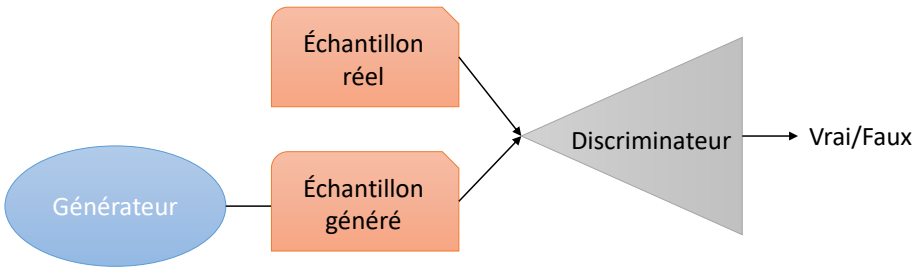

FIGURE 2.4 – Cadre général de l'architecture d'un réseau GAN. Le générateur G produit des échantillons à partir d'un vecteur latent. Puis, le discriminateur D fait la distinction entre les échantillons générés et réels.

un réseau génératif GAN pour régresser des reconstructions plus réalistes. Leur réseau GAN conditionnel est composé d'un générateur ResNet [77] à 50 couches qui estime les paramètres de forme 3DMM, d'expression et celles de projection. Avec ceci, un discriminateur détermine si les paramètres estimés suivent la distribution apprise des images avec leurs paramètres de vérité terrain.

2.3.3.2 Techniques d'apprentissage de bout-en-bout

En utilisant d'autres architectures de réseaux de neurones profonds, la reconstruction 3D à partir d'une image 2D peut être effectuée de bout en bout. Cette application est représentée par $f_\theta : I \rightarrow \mathcal{V}$ où $I \in \mathbb{R}^{H \times W \times 3}$ est l'image faciale d'entrée (H et W désignent respectivement l'hauteur et la largeur de l'image) et $\mathcal{V} \in \mathbb{R}^{3 \times N_X}$ représente la reconstruction 3D finale (N_X est le nombre des sommets du modèle).

Dans les travaux de [42], les auteurs ont présenté une technique de bout en bout pour produire une reconstruction détaillée du visage à partir d'une image. L'architecture est composée de deux réseaux connectés (*CoarseNet* et *FineNet*). Le premier vise à récupérer la géométrie grossière du visage tandis que le deuxième affine les traits du visage de la géométrie déjà produite.

Une autre approche directe a été proposée dans [43] qui effectue une régression directe d'une représentation volumétrique de la géométrie faciale 3D à partir d'une seule image 2D. L'architecture du réseau proposé est constituée de deux modules empilés sans supervision intermédiaire et qui sont basés sur le réseau *hourglass* de [87]. Ce réseau a une structure d'encodeur-décodeur dont laquelle la représentation des caractéristiques estimée par l'encodeur est ramenée au domaine spatial en utilisant des modules résiduels et en combinant les caractéristiques de différentes résolutions à l'aide des connexions de saut.

Dans [8], une structure d'encodeur-décodeur a été utilisée pour apprendre une fonction de transfert entre une image RVB d'entrée et la carte de position UV, qui est une représentation 2D conçue pour enregistrer la forme 3D d'un visage complet dans l'espace UV. Afin de pénaliser les erreurs de reconstruction sur des régions bien spécifiques du visage, les auteurs ont proposé une fonction de coût à base de distance euclidienne avec un masque de pondération qui attribue un poids à chaque pixel de la carte de position UV.

Dans une autre technique, le réseau DF^2Net [51] utilise trois modules combinés, qui sont chacun formé sur un type de données distinct avec des stratégies de formation différentes. Le premier réseau exploite l'architecture U-Net [47] pour transformer l'image d'entrée RVB en une image de profondeur dense. Le deuxième réseau affine l'image de sortie en intégrant des caractéristiques des domaines de profondeur et RVB. Enfin, le dernier réseau affine encore plus les résultats en utilisant une architecture multi-résolution.

L'architecture U-Net [47] a été proposée par Ronneberger *et al.* en 2015 et s'est montrée performante pour la tâche de segmentation des cellules dans un cadre d'images biomédicales.

L'idée principale derrière cette architecture est de rajouter des connexions raccourcies entre les couches d'encodeur et du décodeur afin d'améliorer l'efficacité du modèle. Ainsi, contrairement à l'architecture classique d'un auto-encodeur où les deux parties sont découplées, les connexions par sauts sont utilisées pour transférer des informations à détails fins des couches de bas niveau à partir de l'entrée du modèle vers les couches de haut niveau de la partie décodeur tout en concaténant les couches symétriques de convolution et de déconvolution.

Afin de pouvoir utiliser une quantité illimitée de données pour l'entraînement, des apprentissages de types non supervisés ont été proposés.

Par exemple, un moteur de rendu différenciable a été utilisé dans [63] pour restituer le maillage d'un visage après avoir appris ses paramètres 3DMM. Ces dernières encodent l'identité d'une personne à partir de son image d'entrée en utilisant une architecture à base d'un réseau de reconnaissance faciale qui vise à préserver cette identité dans des conditions d'expression, de pose et d'éclairages variables. L'entraînement du réseau est effectué en utilisant trois fonctions coûts. La première encourage la ressemblance entre les distributions de sortie et du modèle déformable utilisé et la deuxième compare les caractéristiques du visage prédit à celles de l'image d'entrée et assure leurs similarités. Finalement, la dernière garantit que le réseau peut correctement réinterpréter sa propre sortie.

L'apprentissage géométrique approfondi est aussi un axe de recherche qui a motivé les auteurs de [64] pour proposer une nouvelle technique de reconstruction de visage 3D. L'architecture de cette technique est composée d'un encodeur qui extrait les caractéristiques d'une image 2D et un autre qui extrait les caractéristiques d'un maillage facial 3D en couleur. Par conséquent, l'architecture globale consiste en un flux de maillage à maillage et un flux d'image à maillage, qui partagent le même décodeur. L'avantage d'utiliser telle architecture est le fait de former conjointement le décodeur d'une façon auto-surveillé d'une part pour le flux maille à maille et d'autre part pour le flux image à maille en ajoutant une couche de rendu final qui crée une image synthétique à partir du visage 3D résultant.

2.3.3.3 Techniques de reconstruction hybrides

D'autres techniques de reconstruction reposent sur des approches hybrides qui combinent les réseaux de neurones avec des techniques purement géométriques.

Par exemple, Sela *et al.* [44] ont présenté un réseau de neurones basé sur l'architecture U-Net [47]. Il permet de produire directement à partir d'une photo d'entrée, deux types d'images :

une carte de profondeur alignée avec l'image d'entrée et une carte de correspondance de chaque pixel de l'image à un sommet du maillage de référence. Ensuite, les auteurs utilisent une procédure de déformation non rigide afin d'adapter les cartes produites à un modèle de visage en 3D. Enfin, un algorithme de raffinement des détails fins guidé par l'image d'entrée récupère la structure géométrique subtile du visage.

Une autre méthode qui se rapproche des techniques [40, 41] a été proposée par Richardson *et al.* [45]. Elle combine une technique d'apprentissage pour régresser les paramètres 3DMM et une technique à base de SFS qui permet d'affiner le résultat obtenu par le réseau. Tout d'abord, l'image d'entrée est masquée en fonction de l'alignement d'un modèle déformable 3D avec l'image d'entrée. Ensuite, cette image masquée est propagée plusieurs fois à travers le réseau de neurones qui est exploité de manière itérative. Le résultat de ce réseau est enfin affiné par un algorithme de SFS pour avoir une reconstruction 3D détaillée.

Une approche similaire à [45] a été proposée dans [55], dont laquelle la méthode de régression des paramètres 3DMM de [40] a été utilisée pour avoir la forme initiale d'un visage 3D. Ensuite, au lieu d'utiliser une technique de SFS pour relever les détails du visage, les auteurs ont implémenté une technique inspirée de [88] qui saisit les rides du visage et d'autres caractéristiques non définies dans le modèle déformable. Enfin, ils ont complété les détails faciaux manquants dus aux occlusions pour produire la dernière forme 3D en utilisant [89].

Feng *et al.* [66] ont proposé une approche sans modèle. Premièrement, un réseau CNN densément connecté est conçu pour régresser les courbes faciales 3D à partir d'images du plan épipolaire horizontal et vertical. La base d'apprentissage utilisée pour former le réseau a été constituée à partir de scans faciaux de la base BU-3DFE [10]. Deuxièmement, ces courbes sont transformées en nuage de points 3D et l'algorithme d'ajustement [90] est utilisé pour obtenir une surface faciale finale.

Dans une approche [65] similaire à ce que nous présentons dans le chapitre 4, les auteurs ont entraîné un réseau convolutionnel profond qui produit des cartes de champ des normales à partir d'une image d'un visage. Ainsi, cette carte prédite est ensuite utilisée pour reconstruire la surface facile en utilisant la méthode de Frankot-Chellappa [91].

Plus récemment, Wang *et al.* [50] ont utilisé un réseau neuronal de reconnaissance faciale standard formé sur des données de haute qualité pour reconstruire entièrement les géométries faciales à partir d'un seul selfie. Pendant la procédure de test, l'image d'entrée est d'abord utilisée pour régresser directement les coordonnées des sommets d'un modèle de visage 3D avec

une topologie bien définie. Ce modèle est ensuite affiné pour ajuster l'image d'entrée avec une approche de déformation non rigide. Une fois l'ajustement précis effectué, l'image capturée est projetée dans l'espace UV pour en déduire une carte de texture complète.

2.4 Conclusion

Dans ce chapitre, nous avons passé en revue une multitude de stratégies de l'état de l'art pour reconstruire des visages en 3D à partir d'image(s) 2D. Compte tenu de la complexité de résolution de ce problème, des informations supplémentaires sont nécessaires pour effectuer une reconstruction de la géométrie faciale sans ambiguïté. Pour ceci, plusieurs techniques ont été explorées pour rajouter de telles contraintes afin de répondre à cette problématique.

Dans ce contexte, le premier groupe d'approches à base de photométrie (Section 2.3.1), utilise certains a priori comme un modèle de visage de référence [28], la symétrie faciale [73], ou l'utilisation de plusieurs images prises sous différentes sources d'éclairage [31–33]. Bien que ces méthodes soient capables de reconstruire des détails géométriques fins, les visages obtenus contiennent généralement des défauts. Ensuite, le deuxième groupe d'approches utilise les modèles déformables 3DMM (Section 2.2) tels que [1–6] comme contrainte statistique dans un processus d'ajustement (Section 2.3.2) à travers certains éléments extraits de l'image d'entrée comme les caractéristiques locales [1, 21, 27, 36–39], les bords [26, 39], et les points de repère [15, 16, 21, 26, 37–39]. Ce genre de contrainte permet de limiter l'espace des solutions et d'assurer le réalisme des visages 3D reconstruits. Par contre, et contrairement à ce que les méthodes photométriques peuvent offrir, les visages reconstruits grâce aux 3DMM manquent de détails fins et ont des formes grossières. Enfin, le troisième groupe d'approches se focalise sur les techniques à base de réseaux de neurones profonds (Section 2.3.3) qui sont apparus récemment. Quoique ces techniques aient montré de meilleures performances dans de nombreux domaines, leur principal inconvénient est la nécessité d'avoir une quantité de données importantes pour l'étape d'entraînement et pour atteindre les performances attendues.

Dans cette thèse, nous tirons parti des avantages de chacune des techniques décrites précédemment pour constituer deux approches hybrides de reconstruction 3D de visage à partir d'image(s) 2D. Au début, nous allons décrire la constitution de notre générateur de visages synthétiques entièrement contrôlé, qui nous a servis pour préparer l'ensemble de données d'apprentissage utilisé pour entraîner les réseaux de neurones proposés dans les méthodes des chapitres 4 et 5.

Nous présentons dans le chapitre suivant notre première méthode de reconstruction de visage en 3D à partir d'une seule image 2D. Cette approche est le résultat de la combinaison d'une approche à base de réseaux de neurones et d'une approche d'intégration de champ des normales. Cette dernière fait partie des solutions proposées dans les méthodes photométriques et elle est considérée comme une approche variationnelle présentée pour la première fois dans [92] et dans laquelle une certaine fonction doit être minimisée. Après, nous proposons une deuxième méthode pour reconstruire un visage en 3D qui peut utiliser une ou plusieurs images du même sujet dans un seul processus. Cette méthode combine aussi un réseau de neurones et une technique d'ajustement de modèle déformable dont laquelle nous utilisons le modèle LYHM [5] qui permet d'avoir un maillage 3D sémantique possédant les caractéristiques physiques d'un visage humain.

Chapitre

3 Générateur de tête humaine synthétique : la géométrie faciale, les cheveux, les yeux et les lunettes

3.1 Introduction

Au cours des dernières années, l'apprentissage profond a donné de nombreux résultats prometteurs dans différents domaines parmi lesquels la vision par ordinateur. Cependant, afin de former correctement un réseau neuronal, des données sont nécessaires. Elles sont utilisées pour entraîner le réseau à faire des prédictions sur des exemples qu'il n'a jamais vus auparavant. À cet égard, de nombreux efforts ont été réalisés récemment pour améliorer la disponibilité, le volume et la qualité des données. Pour ce faire, la génération des données synthétiques est l'une des méthodes qui a été proposée pour répondre à cette problématique. Ce type de données est important parce qu'elles peuvent être générées pour répondre à des besoins ou des conditions spécifiques qui ne sont pas disponibles dans les données (réelles) existantes. Par exemple, leurs importances pour certaines applications d'apprentissage où la collection de données réelles est coûteuse (exemple : les applications spatiales) et dans d'autres contextes, les exigences en matière de protection de la vie privée qui limitent la disponibilité des données réelles ou la manière dont elles peuvent être utilisées.

L'utilisation des réseaux de neurones pour récupérer la géométrie faciale intégrée dans une image donnée est devenue en quelques années l'outil le plus précieux devant les approches de reconstruction classiques [15, 26–28, 31]. En ce sens, aucun grand ensemble de données appropriées n'est actuellement disponible afin de former un réseau qui se généralise bien. À cette fin, nous

proposons de développer notre propre générateur de données synthétiques qui répond à cette problématique. Notre principale contribution dans ce chapitre consiste à introduire un nouvel outil qui produit des têtes humaines synthétiques 3D photo-réalistes composées de différents éléments avec des géométries connues. En se basant sur des techniques d'infographie, nous intégrons plusieurs modules dans un moteur de rendu 3D qui a pour objectif de créer plusieurs modèles de têtes humaines 3D entièrement synthétiques. À ce propos, nous tirons au hasard plusieurs paramètres de notre générateur pour former des têtes synthétiques qui varient considérablement dans leurs géométries, leurs textures, leurs races, leurs genres, leurs poses, leurs conditions d'éclairage, etc.

En premier lieu, nous présentons dans ce chapitre, un état de l'art détaillé des méthodes de reconstruction 3D de visages à partir d'image(s) qui utilisent des données synthétiques dans la section 3.2. Ensuite, dans la section 3.3, nous décrivons la structure de notre générateur de données synthétiques ainsi que ces différentes composantes. Puis, nous présentons le processus de rendu final de l'image 2D à partir du modèle 3D de la tête humaine synthétique dans la section 3.4. Enfin, nous terminons par une conclusion dans la section 3.5.

3.2 État de l'art : Générateur de visages synthétiques

Le manque des données pour former les réseaux de neurones est le plus grand obstacle à l'application de l'apprentissage profond pour la reconstruction des visages 3D à partir d'images 2D. Cet obstacle est dû à la complexité d'obtenir un grand nombre de modèles faciaux 3D et ces images 2D correspondantes. Face à cette contrainte, plusieurs techniques ont été développées pour construire des ensembles de données synthétiques rendues par les moteurs graphiques et qui utilisent les modèles déformables (3DMM) pour obtenir des visages en 3D de façon plus accessible. L'un des principaux intérêts d'utiliser cette technique est qu'on peut produire un nombre d'exemples pratiquement illimité, disponibles avec des vérités terrain 3D connues. L'efficacité de l'utilisation des images synthétiques a été aussi démontrée avec d'autres types d'applications telles que la détection de visage [93], l'estimation de la pose de la tête [94], la reconnaissance faciale [95], et bien d'autres applications [96, 97].

Dans ce contexte, trois grandes techniques peuvent être identifiées. La première consiste à utiliser les modèles déformables (3DMM) pour générer au hasard des visages 3D et puis d'effectuer le rendu des images synthétiques en utilisant ces mêmes visages générés. La deuxième est tout

simplement d'utiliser les techniques d'ajustement des modèles déformables pour les adapter à des images réelles puis de rendre les images synthétiques en utilisant ces modèles de visages 3D estimés. Pour la troisième technique, il s'agit tout simplement de la combinaison des deux techniques citées ci-dessus et d'obtenir donc les deux formes de données : ajustées et générées.

Nous présentons dans ce qui suit les travaux qui utilisent respectivement les données synthétiques, les données semi-synthétiques et les deux catégories de données pour former un ensemble d'apprentissage pour leurs réseaux de neurones.

3.2.1 Approches à base de données synthétiques

L'idée d'utiliser un ensemble de données complètement synthétiques a été abordée par plusieurs chercheurs. Par exemple, Richardson *et al.* [45] ont utilisé le modèle déformable (3DMM) de [1] avec le modèle de réflectance Phong [76] pour modéliser les ombres, ce qui rend les résultats produits plus réalistes. En suivant la même approche, dans les travaux de [44] et [42], les auteurs ont réalisé une base de données similaire. Par contre, dans [41], Dou *et al.* ont utilisé des données réelles comme BU-3DFE [10] pour initialiser le réseau de neurones profond et pour une seconde étape d'affinement, ils ont utilisé des données synthétiques générées de la même façon décrite auparavant. Piao *et al.* [98] ont aussi généré une base de données synthétique pour former leur réseau, mais ils ont ajouté quelques déformations au niveau de la géométrie du nez et du menton ce qui permet d'éviter le sur-apprentissage du modèle déformable utilisé et rend donc les visages plus réalistes. Dans d'autres approches [63, 99], les réseaux de neurones ont été formés en deux étapes. En premier lieu, des données synthétiques ont été utilisées pour former les réseaux des deux approches. En second lieu, les auteurs de [99] ont appliqué le réseau entraîné sur des données réelles afin d'obtenir des estimations des champs de normales, de l'albédo et de l'éclairage. Ces données produites sont par la suite utilisées comme données de pseudo-supervision pour les données réelles. Par contre, Genova *et al.* [63] ont utilisé des données sans vérité terrain pour effectuer un entraînement non supervisé de leur réseau.

3.2.2 Approches à base de données semi-synthétiques

Cette approche a été proposée en 2016 par Zhu *et al.* [100] où ils ont ajusté le modèle déformable (3DMM) [1] à des images faciales venant de plusieurs bases de données en utilisant les techniques de [101] et [16]. Les résultats obtenus de cet ajustement ont été soumis à des rotations et des projections pour générer des nouvelles images synthétiques ayant les mêmes

paramètres d'identités, mais avec des poses plus larges. Ainsi, en utilisant cette technique, ils ont créé la base de données 300W-LP qui a servi pour de nombreux travaux [8, 43, 60] vu qu'elle comprend des images de visages réalistes avec les paramètres de forme de projection. Dans une autre approche, Tran *et al.* [40, 55] ont utilisé une technique similaire pour calculer les paramètres de formes et de textures à partir de plusieurs images pour une même personne, puis une moyenne pondérée a été appliquée sur les paramètres obtenus pour avoir un seul vecteur de paramètres qui est considéré comme vérité terrain pour toutes les images de la même personne. En dernier ressort, ces données ont servi à l'apprentissage d'un réseau qui régresse les paramètres 3DMM à partir d'une image d'entrée. Dans [17], Liu *et al.* ont proposé une autre méthode qui consiste à ajuster un modèle déformable (3DMM) à plusieurs images d'une seule personne, mais en estimant tout simplement la déformation de forme due aux expressions pour chaque exemple.

3.2.3 Approches à base de données semi-synthétiques et synthétiques

La combinaison de ces deux types de données a été proposée par [65] pour entraîner un réseau de neurones profond qui apprend des cartes du champ des normales à partir des images faciales. La tâche effectuée par ce réseau est similaire à ce que nous proposons dans nos approches décrites dans les chapitres qui suivent. Pour préparer leurs données d'apprentissage, ils ont associé le modèle déformable LSFM [4] au modèle d'expression FaceWarehouse [6] pour générer des visages synthétiques en 3D qui sont par la suite alignés avec images de visages réels. Leur idée est de fournir plus de réalisme au rendu final. Ce même modèle déformable a été aussi utilisé pour être ajusté avec des images de visages réels afin de générer un ensemble de données semi-synthétiques. Pour finaliser les données d'entraînement, deux bases de visages scannés ont été rajoutées pour constituer une seule base de données finale. La même approche de préparation de données a aussi été utilisée dans [51]. Par contre, ils ont rajouté une base de données de visages réels sans vérité terrain 3D pour pouvoir former le réseau de façon auto-contrôlé afin de reconstruire les détails fins en utilisant la technique SFS (*Shape-From-Shading*) qui récupère des informations géométriques à haute fréquence.

3.3 Composition du générateur

Nous suivons les approches présentées dans la section 3.2.1 pour construire notre générateur de têtes synthétiques 3D. Au lieu de synthétiser tout simplement une image 2D en utilisant

le modèle BFM (*Basel Face Model*) comme la plupart des autres approches. Nous regroupons plusieurs modules pour former une seule structure qui produit une quantité illimitée de têtes humaines photo-réalistes. De ce fait, nous allons présenter dans ce qui suit, les différentes parties qui constituent notre générateur, notamment la tête, les yeux, les cheveux et les lunettes comme accessoire.

3.3.1 La tête

La géométrie : Généralement, les méthodes qui génèrent des visages synthétiques [41, 42, 44, 45] utilisent le modèle déformable BFM [1]. Au contraire, nous choisissons de contourner ce choix pour diverses raisons. Premièrement, ce modèle est constitué de 200 sujets, principalement Caucasiens, ce qui limitait l'étendue de la diversité des exemples générés. Deuxièmement, bien que les visages produits par ce modèle soient représentés par un nombre élevé de sommets, la géométrie globale de chaque maillage est entièrement concentrée sur la partie frontale, ce qui affecte son utilisation et nous empêche de synthétiser des exemples ayant des poses larges et avoir un rendu réaliste.

Pour surmonter ces limites, notre choix s'est porté sur le modèle déformable LYHM (*The Liverpool-York Head Model*) proposé par [5] comme élément de base pour produire la géométrie faciale de nos données synthétiques. Ce modèle est composé de 1212 sujets équitablement répartis entre hommes et femmes et qui ont différents âges et ethnicités. Il permet de constituer un maillage sémantique dense avec $N_X = 11510$ sommets et 22392 triangles qui relient les sommets. En plus, il possède une géométrie qui couvre la tête entière (modèle crâniofacial) ce qui lui permet d'être unique par rapport aux autres modèles déformables 3DMM de visage disponible au grand public. L'avantage d'avoir une géométrie complète de la tête nous permet de produire des visages plus réalistes, et surtout nous facilite la tâche de lui rajouter des modèles de cheveux en 3D. Avec le modèle global LYHM, les chercheurs [5] ont fourni d'autres modèles qui utilisent des sous-populations de l'ensemble d'apprentissage, notamment, des modèles spécifiques au genre (LYHM-*male* et LYHM-*female*) que nous utilisons dans notre générateur pour pouvoir contrôler la géométrie de la tête tout en choisissant la texture faciale qui lui correspond..

Comme nous l'avons montré dans la section 2.2, en utilisant un modèle déformable 3DMM, nous produisons un nombre infini de visages synthétiques. La génération d'une tête synthétique est effectuée en choisissant aléatoirement les coefficients du vecteur des paramètres de l'équation (2.2) selon une distribution gaussienne $\boldsymbol{y} \sim N(0, 1)$. Une fois la forme géométrique X est obtenue,

nous l'utilisons avec la liste de triangles prédéfinie pour former un maillage qui définit la structure de la tête.

La texture : La plupart des visages synthétiques générés dans les approches telles que [41, 42, 44, 45] possèdent des textures qui ont été créées en utilisant le modèle paramétrique de texture fourni avec le modèle BFM. Comme il a été proposé pour la géométrie, ce modèle est constitué d'une texture moyenne, d'une matrice des composantes principales et d'un vecteur de coefficients. Ainsi, une nouvelle texture peut-être produite en utilisant une formulation similaire à celle décrite dans l'équation (2.2). Dans ce sens, bien que nous puissions générer les textures en utilisant le modèle LYHM, ce dernier a un inconvénient majeur qui nous mène à choisir une autre méthode pour constituer la texture de la tête. Il s'agit d'une coloration bleue qui est omniprésente sur toute la partie du crâne du modèle (Voir (e) de la figure 2.2). Cette couleur vient du fait que tous les sujets qui ont constitué le modèle déformable LYHM ont porté des capuchons bleus en latex bien ajustés lors du processus de scan. L'utilité de ces capuchons est que la géométrie des cheveux ne nuit pas à la reconstruction finale, ainsi le rendu obtenu se rapproche de la forme réelle de la géométrie d'une tête humaine.

Pour pallier cet inconvénient, nous proposons d'appliquer la technique de plaquage de carte *UV* (ou *UV-Mapping*) [102, 103] qui est largement utilisé en infographie et en synthèse d'images pour améliorer la qualité du rendu.

Considérant le modèle 3D de la tête X de $\mathbb{R}^3$, cette technique consiste à trouver la paramétrisation $f(u, v)$ de X qui est une fonction bijective qui met un sous-ensemble Ω de $\mathbb{R}^2$ (appelé domaine paramétrique) en correspondance avec le modèle X.

$$\begin{aligned} f &: \mathbb{R}^2 \rightarrow \mathbb{R}^3 \\ (u, v) &\rightarrow f(u, v) = [x(u, v),\ y(u, v),\ z(u, v)]^\intercal \end{aligned} \tag{3.1}$$

Pour ce faire, il faut tout d'abord déplier l'objet 3D en une carte plane 2D qui est représentée sur un plan bidimensionnel de coordonnées u et v. En utilisant cette carte, nous plaçons plusieurs types de textures qui seront déformées par la suite selon les lignes du dépliage de la carte UV, pour être appliquée sur le modèle 3D. Ainsi, chaque sommet $\boldsymbol{p}$ de notre modèle 3DMM est alors transféré à un pixel a de coordonnées (u, v) de la carte de texture en utilisant la paramétrisation inverse définie par $f^{-1} : (x, y, z) \rightarrow (u, v)$. Cette information est par la suite stockée comme un couple (u_i, v_i) associé à chaque sommet $\boldsymbol{p}_i$ avec $i \in [1, ..., N_X]$. Avec ce procédé, nous pouvons

obtenir des visages 3D riches en détails visuels en appliquant différents types de textures.

De ce fait, dans notre contexte, cette technique de *UV-mapping* est appliquée en utilisant le maillage de notre tête déformable 3DMM avec les coordonnées 3D du modèle moyen X_0. Étant donné que le modèle 3DMM que nous utilisons, possède une structure sémantique fixe, nous réalisons cette opération une seule fois et nous stockons les coordonnées de la carte UV obtenue pour l'utiliser par la suite avec chaque nouvelle tête générée. Pour pouvoir appliquer cette technique, nous utilisons le logiciel de modélisation 3D Blender [104] et dans la figure 3.1 nous montrons la carte UV que nous avons obtenue.

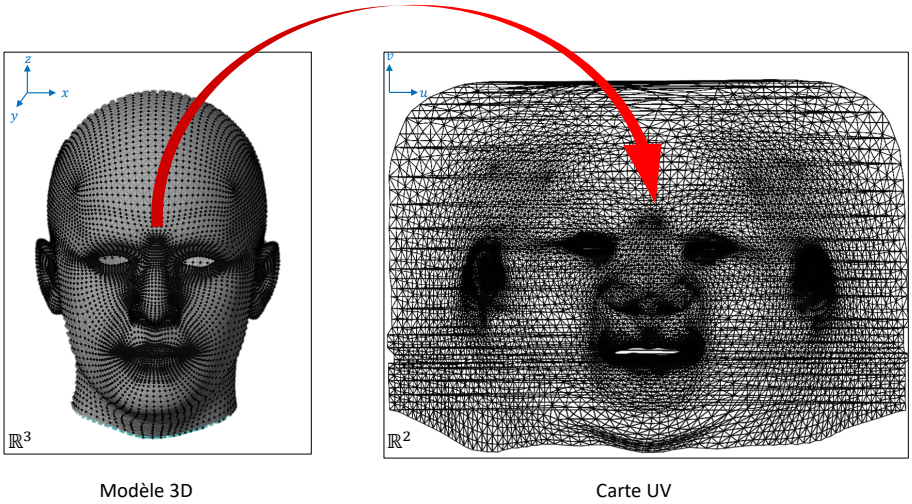

FIGURE 3.1 – Création de la carte UV à partir d'un modèle 3D (*UV-Mapping*)

Une fois que nous obtenons la carte UV, nous utilisons pour la restitution du modèle de tête obtenu, une technique de rendu 3D (*shader*) connue sous le nom de Rendu Physique Réaliste –ou *Physically Based Rendering*– (PBR) [105]. Cette technique imite les propriétés physiques des modèles quand ils sont exposés à la lumière dans le monde réel. En l'appliquant, les objets 3D virtuels auront une apparence proche de celle qu'ils auraient en réalité. Cette apparence est bien plus attrayante que celle obtenue à partir d'autres techniques comme Phong [76] ou Blinn-Phong [106] utilisées dans les approches proposées dans l'état de l'art. Pour pouvoir appliquer la technique PBR, différentes composantes sont nécessaires pour définir les matériaux qui constituent un modèle 3D et donc la façon dont ils reflètent et diffusent la lumière reçue. Ces éléments de base sont : la carte d'albédo, la carte de rugosité, la carte métallique et la carte des normales.

Dans notre générateur, nous utilisons ces éléments de base sous forme de textures faciales qui permettent de donner un aspect visuel de haute-définition et d'enrichir les détails géométriques 3D de la tête. Elles peuvent porter des informations permettant de rajouter des subtilités qui

enrichissent l'apparence et le réalisme du rendu 3D (par exemple : rajouter des reliefs à la surface).

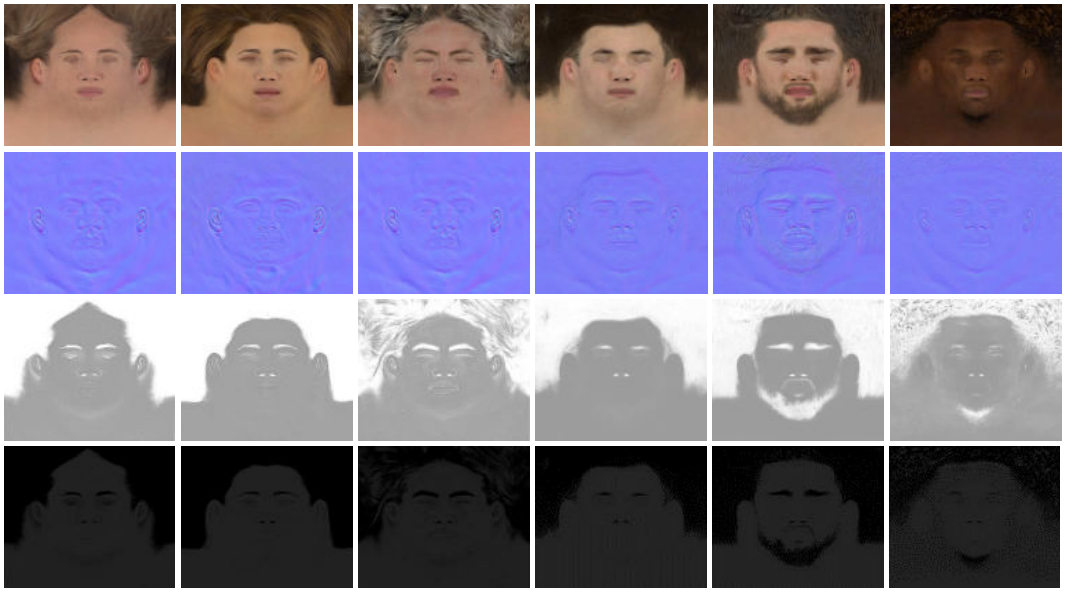

FIGURE 3.2 – Exemples des différents types de textures utilisées pour le rendu 3D des têtes générées. De haut en bas : carte d'albédo, carte des normales, carte de rugosité, carte métallique.

— La première texture est appelée carte d'albédo - *Albedo Map* (3 canaux) (Première ligne de la figure 3.2). Il s'agit de la carte principale qui stocke les couleurs brutes de la surface de l'objet 3D (le modèle de la tête dans notre cas). Elle est utilisée durant le processus de rendu pour fournir la première apparence de la tête synthétique.

— La deuxième texture est la carte des normales - *Normal Map* (3 canaux) (Deuxième ligne de la figure 3.2). Elle est utilisée pour améliorer l'apparence d'un maillage lisse sans ajout de polygones. Cette carte contient plusieurs aspérités qui vont améliorer la perception des reliefs de l'objet et donc donner plus de réalisme au rendu visuel 3D. Dans notre cas, la carte des normales fait apparaître des détails du visage comme les piqûres, les rides, les cicatrices, les grains de beauté, etc.

— La troisième texture est la carte de rugosité - *Roughness Map* (1 canal) (Troisième ligne de la figure 3.2). Il s'agit d'une texture en nuances de gris qui est utilisée pour définir dans quelle mesure la surface de l'objet 3D est rugueuse ou lisse. Plus le pixel est clair, plus la rugosité est élevée, et par conséquent, la lumière est plus dispersée et les reflets deviennent plus flous. À l'inverse, quand le pixel est sombre, il est considéré lisse et les reflets sont plus nets. Avec cette carte, la rugosité de la surface est contrôlée avec un paramètre qui contrôle l'intensité de cet effet. Dans notre cas, la rugosité est importante surtout à l'intérieur des pores et des rides ce qui les rend moins brillantes.

— La dernière texture est la carte métallique - *Metallic Map* (1 canal) (Quatrième ligne de la figure 3.2). Elle indique le niveau de réflexion de lumière sur chaque partie de l'objet 3D. Les réflexions sur les surfaces métalliques ont tendance à être plus élevées, tandis que les réflexions sur les surfaces non-métalliques sont plus neutres. Comme pour la carte de rugosité, l'effet de la carte métallique est contrôlé par un paramètre. Dans notre cas, cette carte indique le niveau de brillance de la peau qui peut être due à l'humidité.

Nous intégrons dans notre générateur 200 exemples de textures (également réparties entre hommes et femmes), où chacun d'entre eux est formé de quatre types de textures décrits ci-dessus. À chaque fois où nous voulons produire un nouvel exemple de tête synthétique, nous tirons au hasard un exemple de la banque de texture en prenant en considération le modèle qui a été sélectionné pour produire la géométrie (LYHM-*male* ou LYHM-*female*). L'ensemble de ces cartes est fourni par l'entreprise dans laquelle s'est déroulée cette thèse (Thèse CIFRE effectuée au sein de la start-up *MCQ-Scan*). Enfin, nous montrons quelques résultats obtenus de l'application de certaines textures avec des têtes synthétiques générées dans la figure 3.3.

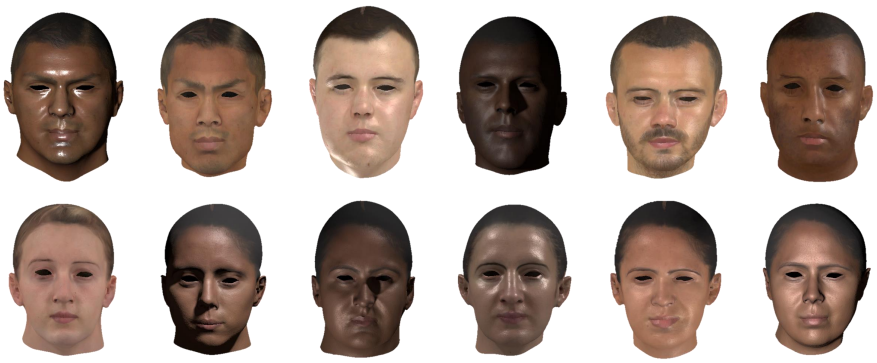

FIGURE 3.3 – Exemples de têtes synthétiques 3D texturées construites à partir de notre générateur de données.

3.3.2 Les yeux

La géométrie : Le modèle déformable 3DMM que nous utilisons dans ce procédé de génération de têtes synthétiques ne couvre pas la géométrie des yeux (voir les exemples texturés dans la figure 3.3). De ce fait, nous créons un maillage supplémentaire pour couvrir cette région et qui a pour but de compléter la géométrie du modèle déformable et donc de produire un modèle 3D qui se rapproche d'une tête humaine réaliste. En particulier, nous modélisons deux maillages distincts pour former une géométrie approximative d'un œil humain, il s'agit du globe oculaire et de la cornée. Ces deux parties sont ensuite assemblées pour former un seul maillage que nous

pouvons le visualiser dans (a) de la figure 3.4. Après avoir appliqué les différentes textures que nous décrivons dans la suite de cette section, nous dupliquons ce modèle 3D texturé pour former les yeux que nous recalons dans des endroits bien déterminés.

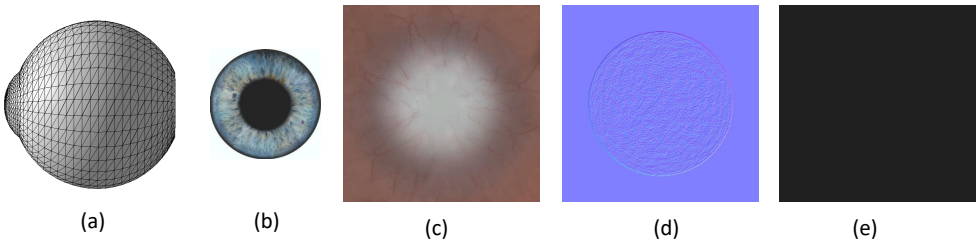

FIGURE 3.4 – Les différents éléments utilisés pour constituer le modèle des yeux. (a) maillage, (b) texture de l'iris. (c), (d) et (e) représentent respectivement les cartes d'albédo, de normales et métallique pour le globe oculaire.

Le processus d'alignement entre le modèle des yeux et le modèle 3D de la tête est effectué en deux étapes. Une étape de pré-alignement où nous estimons les paramètres de recalage pour une seule fois en utilisant les deux maillages des yeux et le modèle moyen X_0 de notre 3DMM. Ces paramètres seront par la suite utilisés à chaque fois que nous générons un nouvel exemple de tête synthétique complet. Ensuite, la deuxième étape consiste à effectuer un recalage fin pour optimiser la position des yeux à chaque modèle synthétique.

Pré-alignement : Dans cette étape, nous expliquons le processus de pré-alignement entre le modèle des yeux et le modèle moyen X_0 de notre 3DMM. Ce recalage consiste à aligner les deux modèles en utilisant des points caractéristiques choisis manuellement sur chaque modèle. Il s'agit de quatre paires de sommets qui sont illustrés dans la figure 3.5.

Ils permettent de calculer une transformation non rigide entre les deux modèles puis de l'appliquer sur le modèle approprié (modèle des yeux dans notre cas) pour minimiser la distance entre eux au sens des moindres carrés et à cet effet, les modèles deviennent alignés. Les paramètres de cette transformation sont : mise à l'échelle, rotation et translation. Ils sont estimés en utilisant l'algorithme itératif du point le plus proche voisin ICP [107]. Cette étape présente une bonne initialisation à l'étape d'affinement suivante pour chaque tête générée, considérant que la position des sommets des contours des yeux est principalement la même entre le modèle moyen X_0 et un modèle synthétisé X. Ce qui veut dire que si nous superposons les deux modèles, la distance entre les sommets de ces endroits est minimale. Donc le processus de recalage fin ne sera pas coûteux et il s'exécutera rapidement.

Recalage fin : Après avoir obtenu un modèle de tête 3D synthétisé comme nous l'avons

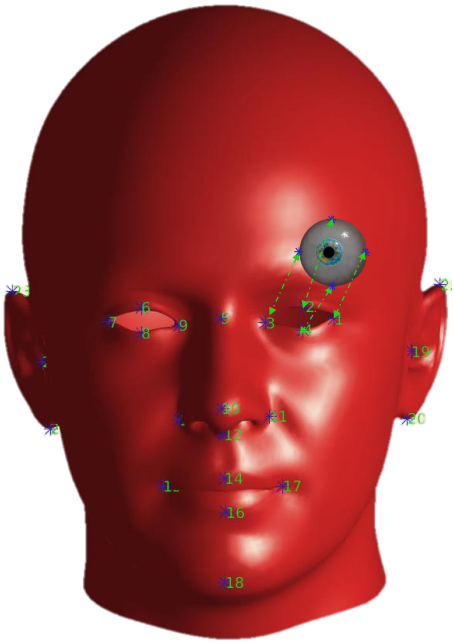

FIGURE 3.5 – Recalage du modèle 3D d'oeil en utilisant 4 points.

décrit dans la section 3.3.1, nous utilisons les paramètres que nous avons estimés dans l'étape de pré-alignement pour avoir un recalage grossier entre les yeux et la tête 3D. Par la suite et pour affiner l'emplacement des yeux, nous re-appliquons le processus de recalage de nouveau, mais en utilisant cette fois la tête synthétique générée X. Ce processus de mise en correspondance nous permet d'avoir des paramètres de transformation propres à chaque exemple généré.

La texture : Nous utilisons pour l'aspect visuel des yeux un ensemble de textures que nous appliquons séparément pour le globe oculaire et la cornée. Ce faisant, nous appliquons les mêmes techniques décrites ci-dessus (plaquage de texture et rendu 3D PBR) et ceci dont le but d'avoir un aspect plus réaliste du rendu final. Dans la figure 3.4, les textures (c) (d) et (e) sont respectivement utilisées comme carte d'albédo, carte de normales et carte métallique pour le globe oculaire. Vu que cette partie des yeux possède généralement les mêmes caractéristiques pour toutes les personnes, nous conservons ces textures pour tous les modèles d'yeux utilisés dans notre générateur. Par contre, parmi une collection de textures possédantes différentes couleurs, nous tirons une texture au hasard pour créer plus de variations pour l'iris. Nous illustrons dans la figure 3.6, les différentes textures utilisées pour l'iris. Pour chaque tête générée, nous tirons au hasard une texture d'iris que nous plaquons directement au maillage de la cornée.

Finalement, dans la figure 3.7, nous montrons un exemple d'un maillage des yeux qui sont prêts à être alignés avec la tête 3D synthétisée.

1. Source : https://edouardjanssens.com/

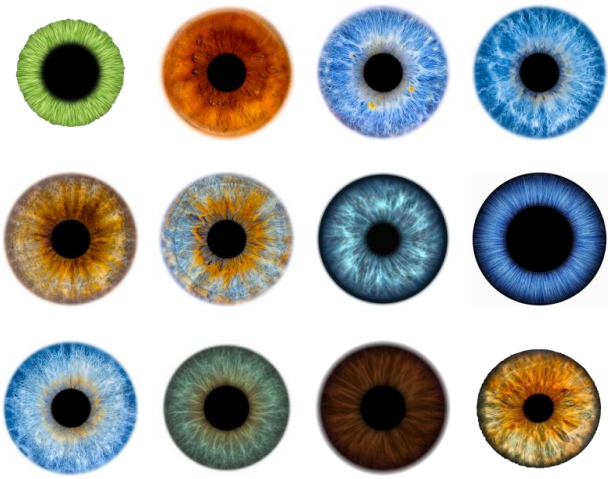

FIGURE 3.6 – Exemples de quelques iris [1] utilisés pendant la génération des modèles de yeux.

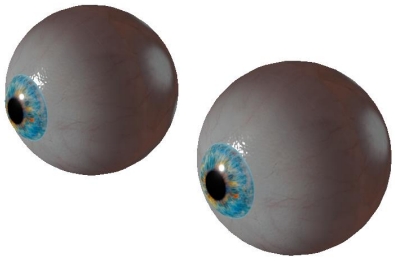

FIGURE 3.7 – Un exemple d'un maillage de yeux utilisé dans notre générateur de tête synthétique.

3.3.3 Les cheveux

La géométrie : À la différence des autres approches qui n'utilisent que le modèle déformable de visage 3DMM pour créer des images synthétiques, pour notre générateur de données, nous proposons de modéliser les cheveux pour se rapprocher d'un rendu réaliste d'une tête humaine 3D. Contrairement aux objets qui sont facilement paramétrables, comme le visage humain, les cheveux couvrent un large éventail de variations de forme et peuvent être très complexes en raison de leur structure volumétrique et du niveau de déformabilité de chaque poil. Pour répondre à cette problématique, nous utilisons une base de données de coiffures 3D appelée *USC-HairSalon* et qui contient 343 modèles de coiffures obtenus manuellement grâce à une communauté de jeux vidéo [108]. Dans cette base de données, toutes les coiffures sont alignées avec un modèle de tête 3D standard et chaque modèle est composé de $N_P = 10000$ poils qui ont tous des racines uniformément réparties sur le cuir chevelu. Plus précisément, chaque poil contient $M_P = 100$ sommets qui sont alignés l'un après l'autre partant de la racine jusqu'à la pointe en formant une courbe qui donne une allure d'un vrai poil de cheveux.

Augmentation de données : Dans le but d'augmenter le nombre de modèles, nous adoptons la technique décrite dans [109]. Elle consiste à appliquer un effet miroir et un mélange par paire d'exemples. Avant d'appliquer ce dernier, les modèles de cheveux sont séparés en 12 classes en fonction de quelques combinaisons de leurs styles et tailles (court, moyen, long, lisse, bouclé), de ce fait, le mélange est appliqué pour chaque paire de coiffures au sein de la même classe pour avoir des exemples plus naturels. L'opération de mélange consiste à regrouper les poils d'un modèle pour former cinq mèches [1] qui ont chacun un poil central. Par la suite, en se servant de ces poils centraux, nous réalisons plusieurs mélanges entre les différents modèles. À travers ce procédé, nous générons un ensemble de 9000 modèles de coiffures en créant des combinaisons au hasard entre les paires de chaque classe. Nous montrons dans la première ligne de la figure 3.8, quelques exemples obtenus après avoir effectué ce pré-traitement.

Création de maillages : Les modèles obtenus après cette étape d'augmentation, nous ne permettons pas de réaliser un rendu graphique similaire à ce qui a été fait avec les modèles de tête et de yeux décrits précédemment. Ceci est dû au fait que ces modèles sont seulement composés de sommets 3D et ils ne disposent pas de triangles qui les relient. Donc, ceci nous empêche d'effectuer la technique de plaquage de texture et avoir donc un rendu graphique 3D similaire aux autres modèles. De ce fait, il est donc nécessaire d'effectuer un second ensemble de transformations sur chaque modèle en vue d'obtenir un maillage complet composé de sommets accompagnés d'un groupe de triangles possédant une structure bien définie comme il a été adopté dans [110–113]. La deuxième ligne de la figure 3.8 contient des exemples obtenus après avoir appliqué les différentes transformations de création de mailles. Ces transformations ont pour autre motivation de réduire le grand nombre de sommets lorsque chaque poil du modèle de coiffure est représenté individuellement et de minimiser donc la taille de l'ensemble de données et par conséquent augmenter la vitesse de chargement des modèles ce qui améliore les performances du processus de génération des modèles de tête synthétique. En moyenne, le temps de chargement d'un modèle est passé de 0.5 secondes à 10^{-4} secondes après avoir appliquer le traitement.

Le processus de création d'un maillage pour chaque modèle de coiffure est donc basé sur les étapes suivantes :

— À partir des racines de chaque modèle de coiffure, nous appliquons l'algorithme de partitionnement de données *K-Means* [114] pour diviser l'ensemble des N_P poils en $k = 600$ groupes. Plus précisément, pour tous les sommets de p_1^1 à $p_1^{N_P}$, représentant les racines des

1. mèche : ensemble de poils regroupés

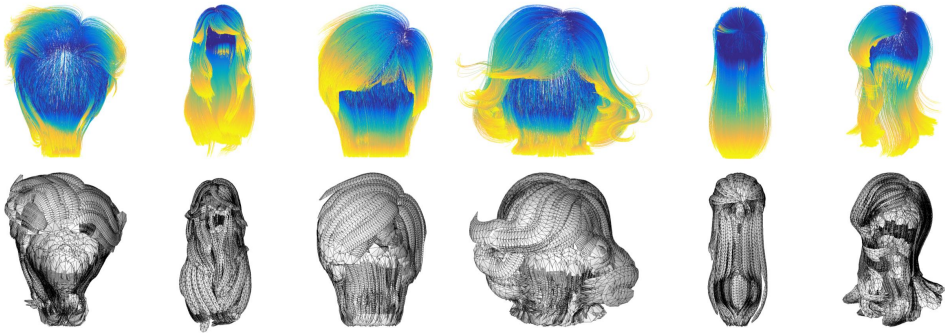

FIGURE 3.8 – Quelques exemples de modèles de coiffure avant et après leurs transformations en maillages. Pour chaque colonne, le modèle en nuage de points se trouve en première ligne et son maillage correspondant est à la deuxième ligne.

poils, l'algorithme de regroupement *K-Means* [114] est exécuté pour trouver une partition $C = \{C_1, C_2, ..., C_k\}(k \leq N_P)$ en minimisant la distance entre les points à l'intérieur de chaque partition :

$$\arg\min_{C} \sum_{j=1}^{k} \sum_{x_1 \in C_j} ||p_1 - \mu_j||^2$$

où μ_j est le barycentre des points dans C_k. Une fois que l'algorithme converge, nous récupérons les barycentres μ de l'ensemble des partitions de C, ce qui nous donne en total 600 poils. Les poils échantillonnés sont représentés en rouge dans (b) et sont isolés dans (c) de la figure 3.11.

— À ce stade, pour chaque poil composé de M_P sommets et qui représente un barycentre dans l'étape précédente, nous résolvons 11 équations qui vont produire $6M_P + 1$ sommets formant un nuage de point autour du poil sélectionné. Ce nuage de point représente une approximation de la surface d'une plaque mince et courbée (en forme de U [110]) qui entoure le poil et donne plus de volume. Pour atteindre cet objectif, nous calculons tout d'abord un certain nombre de composantes pour chaque sommet $\boldsymbol{p}_i$ de ce poil définissant ainsi un repère de *Frenet* [115] pour chaque point [116]. Ce repère est un outil d'étude du comportement local de la courbe constituée par une tangente t_i, une normale n_i et une bi-normale b_i. Ces composantes vont être utilisées dans ce qui suit pour la création des différents sommets qui vont former la forme U.

$$t_i = \frac{p_{i+1} - p_i}{||p_{i+1} - p_i||} \quad , \quad n_i = \frac{t_i \times (p_i - \bar{p})}{||t_i \times (p_i - \bar{p})||} \quad b_i = \frac{n_i \times t_i}{||n_i \times t_i||} \tag{3.2}$$

avec $\bar{p}$ est le point central du modèle de coiffure. Nous montrons dans la figure 3.9 un exemple des repères de *Frenet* pour quelques sommets d'une courbe 3D.

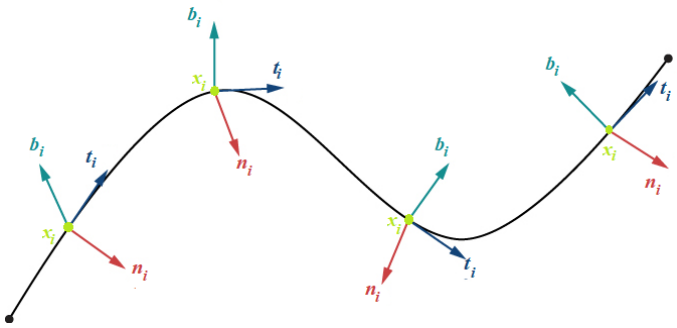

FIGURE 3.9 – Illustration des repères de *Frenet* composés de (tangente (t_i), normale (n_i) et bi-normale (b_i)) pour quelques sommets (p_i) d'un exemple de poil.

Après, nous déterminons les cinq courbes P^1, P^2, P^3, P^4, P^5 et les six sommets A_F, B_F, C_F, A_L, B_L, C_L qui forment le nuage de point qui entoure le poil choisi à partir des équations suivantes :

$$\text{Sommets à la racine :} \begin{cases} A_F = p_1 + \gamma_h\, \gamma_n(1)\, n_1 + \gamma_h\, \gamma_b(0)\, b_1 - \gamma_h\, \gamma_t(0)\, t_1 \\ B_F = p_1 + \gamma_h\, \gamma_b(0)\, b_1 - \gamma_h\, \gamma_t(1)\, t_1 \\ C_F = p_1 + \gamma_h\, \gamma_n(2)\, n_1 + \gamma_h\, \gamma_b(0)\, b_1 - \gamma_h\, \gamma_t(2)\, t_1 \end{cases}$$

$$\text{Sommets à la pointe :} \begin{cases} A_L = p_{M_P} + \gamma_h\, \gamma_n(1)\, n_{M_P} + \gamma_h\, \gamma_b(0)\, b_{M_P} + \gamma_h\, \gamma_t(0)\, t_{M_P} \\ B_L = p_{M_P} + \gamma_h\, \gamma_b(0)\, b_{M_P} + \gamma_h\, \gamma_t(1)\, t_{M_P} \\ C_L = p_{M_P} + \gamma_h\, \gamma_n(2)\, n_{M_P} + \gamma_h\, \gamma_b(0)\, b_{M_P} + \gamma_h\, \gamma_t(2)\, t_{M_P} \end{cases}$$

Et pour i allant de 1 à M_P :

$$\begin{array}{c}\text{Sommets qui constituent} \\ \text{la plaque courbée qui recouvre le poil :}\end{array} \begin{cases} P_i^1 = p_i + \gamma_h\, \gamma_n(0)\, n_i + \gamma_h\, \gamma_b(0)\, b_i \\ P_i^2 = p_i + \gamma_h\, \gamma_n(1)\, n_i + \gamma_h\, \gamma_b(1)\, b_i \\ P_i^3 = p_i + \gamma_h\, \gamma_b(2)\, b_i \\ P_i^4 = p_i + \gamma_h\, \gamma_n(2)\, n_i + \gamma_h\, \gamma_b(3)\, b_i \\ P_i^5 = p_i + \gamma_h\, \gamma_n(3)\, n_i + \gamma_h\, \gamma_b(4)\, b_i \end{cases}$$

où $\gamma_h = 0.015$, $\gamma_n = [-1.5, -1, 1, 1.5]$, $\gamma_b = [-0.2, 0.2, 0.4, 0.2, -0.2]$ et $\gamma_t = [1.5, 2, 1.5]$ sont des paramètres qui définissent le niveau de courbure, la taille et la forme globale de la plaque courbée. Le résultat de tout ce processus est illustré pour un exemple de poil dans (e) de la figure 3.11 et avec une vue de dessus pour un autre exemple dans la figure 3.10.

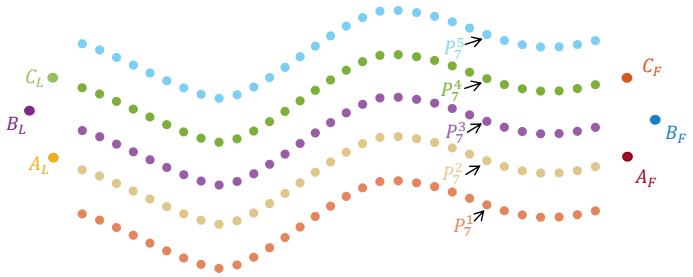

FIGURE 3.10 – Vue de dessus d'un exemple de nuage de point généré à partir d'un poil.

— Pour former un maillage à partir du nuage de point obtenu, nous définissons une structure de triangles qui relient les sommets entre eux pour former des liens [110–113, 117]. Les différents polygones sont placés l'un à côté de l'autre et partagent leurs arêtes pour éviter les trous, ce qui produit une approximation d'une surface qui couvre le poil mis en jeu (Voir (f) de la figure 3.11).

— Nous répétons maintenant la deuxième et la troisième étape pour tous les poils obtenus dans la première étape pour former une maille (forme de plaque courbée) pour chaque poil. Finalement, nous obtenons un groupe de mailles qui se superposent les unes aux autres sur le dessus du cuir chevelu pour former un seul maillage global pour chaque modèle de coiffure (Voir (g) de la figure 3.11).

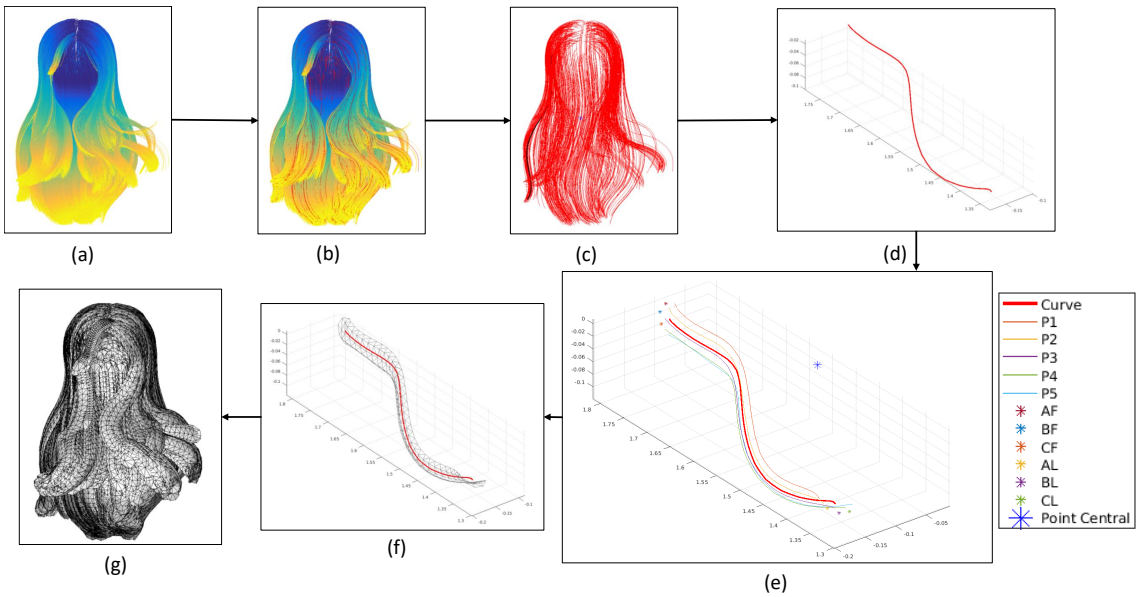

FIGURE 3.11 – Processus de création de maillage à partir de chaque nuage de point du modèle de cheveux. À partir de chaque modèle (a) nous effectuons un échantillonnage pour avoir un certain nombre de poils (b et c). Puis, pour chaque poil récupéré (d), nous générons un nuage de point autour du poil mise en jeu (e) qui devient par la suite un maillage (f). Enfin, cette démarche est répétée pour tous les échantillons pour former un modèle global (maillage) d'une coiffure.

Toutes les étapes décrites ci-dessus sont appliquées aux 9000 modèles de coiffures qui ont été produites à partir de l'étape d'augmentation de données. Après, pour chaque modèle choisi durant le processus de génération d'une tête humaine synthétique, le modèle choisi aléatoirement est ensuite aligné avec un exemple de modèle de tête LYHM qui a été généré comme nous l'avons décrit dans 3.3.1. La démarche d'alignement est similaire à ce que nous avons fait pour les yeux (Voir Section 3.3.2). En ce sens, nous commençons par un pré-alignement entre le modèle de tête moyenne X_0 et tous les modèles de la base de données de coiffures en utilisant le modèle de tête fourni avec la base qui, par définition, est aligné avec les modèles de coiffure de la base. Ensuite, une fois cette étape de recalage initiale est effectuée. Nous affinons par la suite cet alignement entre chaque modèle de coiffure et chaque modèle de tête généré X visant à donner un aspect visuel optimal pour le rendu final.

La texture : La simulation numérique des cheveux est un élément fondamental de la création d'humains virtuels convaincants. De nombreuses applications graphiques, y compris la réalisation des films d'animation et le développement de jeux-vidéo, font un fort usage de personnages virtuels dont les coiffures doivent être modelées de manière réaliste. Dans notre cas, étant donné que nous avons obtenu une banque de modèles de coiffures complets, les utiliser pour avoir un rendu réaliste est devenu un problème de synthèse puis de plaquage de texture 2D. Cette technique est similaire à ce qui a été fait dans [111–113, 117] et ce que nous avons proposé pour les modèles de la tête et des yeux décrits précédemment.

Comme nous l'avons décrit auparavant, pour pouvoir plaquer une texture à un maillage 3D, il est primordial de disposer d'une paramétrisation UV. Pour ce qui concerne les modèles de coiffures, étant donné que nous connaissons la structure du maillage de chaque modèle de coiffure, nous définissons une paramétrisation UV pour une seule mèche d'un modèle. Vu que nous utilisons la même structure (triangulation) pour toutes les mèches qui constituent un modèle de coiffure, nous gardons aussi la même paramétrisation UV pour toutes les mèches. Par conséquent, tous les exemples de la base des coiffures partagent les mêmes coordonnées UV. De ce fait, pour chaque sommet donné p d'un maillage, sa coordonnée paramétrique (u, v) est déjà connue.

Le modelage des cheveux est une tâche difficile, principalement en raison de la complexité des cheveux. Une tête humaine est généralement constituée d'un grand volume de cheveux, avec en moyenne 120 000 poils, avec un petit diamètre pour chaque poil. Compte tenu de cette dualité, notre technique de création de texture pour les coiffures repose sur la simulation d'un ensemble

de poils en utilisant une paramétrisation de courbe 3D pour chaque poil. Cette procédure va définir le caractère global des cheveux en simulant des aspects comme la couleur, la longueur, la densité et le niveau d'ondulation. Une fois généré, l'ensemble des poils 3D est rendu dans le plan (XY) puis mappé sur toutes les mèches d'un modèle de coiffure choisi. Avec suffisamment de mèches texturées, nous simulons un cuir chevelu rempli de cheveux qui se chevauchent. Pour donner un aspect réaliste et éviter un rendu grossier du modèle texturé obtenu, nous utilisons la technique de rendu graphique 3D de Kajiya-Kay, conçu spécialement par [118] pour le rendu des cheveux. Le *shader* utilisé peut piloter divers effets tels que la profondeur, l'effet de diffusion de la lumière, la transparence, etc. Pour pouvoir appliquer cette technique, plusieurs types de textures doivent être rendus à partir de l'ensemble des poils 3D. Dans la suite de cette partie, nous décrivons le processus de génération des poils en 3D et par la suite, nous présentons les différentes textures générées et utilisées par le *shader* de Kajiya-Kay pour le rendu final des cheveux.

Paramétrisation 3D des poils : Pour faciliter la synthèse des textures qui seront mappées par la suite sur les différentes mèches de chaque coiffure, nous définissons une paramétrisation 3D pour simuler l'allure de chaque poil en se basant sur des fonctions sinusoïdales qui s'inspire de [119–121]. Dans ce sens, chaque poil est considéré comme une chaîne de points 3D.

En premier temps, nous fixons un ensemble de paramètres qui définissent la forme globale des poils. Il s'agit de la longueur moyenne des poils $(\overline{h})$, le nombre des poils (N_H) et le nombre de sommets de chaque poil (M_H). Nous initialisons aussi les paramètres de quelques lois de distributions gaussiennes (m_x,ω_x) qui vont être utilisés pour générer aléatoirement les coefficients de la fonction sinusoïdale qui synthétise un poil 3D. Après ceci, nous tirons au hasard les coefficients $\{(A_x, A_z)$: amplitudes, ω : pulsation, (ϕ_x, ϕ_z) : phases à l'origine, δ_y : décalage entre les sommets d'un poil$\}$ qui définissent les paramètres de la fonction génératrice de chaque poil. Finalement, pour obtenir une structure 3D d'un ensemble de poils uniformément répartis sur l'axe X, nous répétons le processus de simulation de poil défini par une fonction sinusoïdale 3D qui varie le long de l'axe Y.

Cette méthode de génération est pratique pour de nombreuses raisons : elle est facile à mettre en œuvre, efficace (en répétant le processus de génération d'un poil, un nombre illimité de poils peuvent être imités de cette façon.) et de plus la géométrie des poils est correctement établie. L'algorithme 1, résume les étapes du processus de génération des poils 3D et la partie (e) de la figure 3.12 illustre un exemple de simulation des poils en 3D.

Algorithme 1 : Algorithme de génération d'un ensemble de poils en 3D

Entrées : N_H , M_H
m_x , ω_x // Initialisation des paramètres des sinusoïdes
$\overline{h}$ // Longueur moyenne des poils
Résultat : $Poils$

Initialiser l'ensemble des $Poils$;
$\delta_y \leftarrow \frac{\overline{h}}{M_H}$ // Décalage entre les sommets d'un $Poil$
pour $P_{idx} \leftarrow 1$ **à** N_H **faire**
    Initialiser un nouveau $Poil$;
    $x_0 \leftarrow P_{idx}$;
    y_0 , $z_0 \leftarrow 0$;
    /* Tirer au hasard les paramètres des sinusoïdes selon des distributions gaussienne */
    $\phi_x \leftarrow N(0, \pi)$ // N : loi gaussienne de moyenne nulle et d'écart-type π
    $\phi_z \leftarrow N(0, \pi)$;
    $A_x \leftarrow N(m_x, 0.2\, m_x)$;
    $A_z \leftarrow N(m_x, 0.2\, m_x)$;
    $\omega \leftarrow N(\omega_x, 0.2\, \omega_x)$;
    /* Génération des points 3D pour créer un poil */
    pour $i \leftarrow 0$ **à** M_H **faire**
        $y \leftarrow y_0 + i\, \delta_y$;
        $x \leftarrow x_0 + A_x \cos(\omega\, y + \phi_x)$;
        $z \leftarrow z_0 + A_z \cos(\omega\, y + \phi_z)$;
        $Poil \leftarrow \{x,\, y,\, z\}$;
    $Poils \leftarrow Poil$;

Rendu graphique de cheveux : Pour visualiser la coiffure complète de manière réaliste, il est nécessaire de prendre en compte l'interaction lumineuse qui se produit entre l'ensemble des poils et les propriétés optiques de chacun d'entre eux. Pour ce faire, nous adaptons le *shader* de Kajiya-Kay [118] en utilisant différentes textures générées à partir de la projection de l'ensemble des poils 3D dans le plan (XY). Ces textures sont par la suite plaquées sur les différentes mèches d'un modèle de coiffure. Pour l'implémentation de cette méthode de rendu, nous nous inspirons du travail de [122].

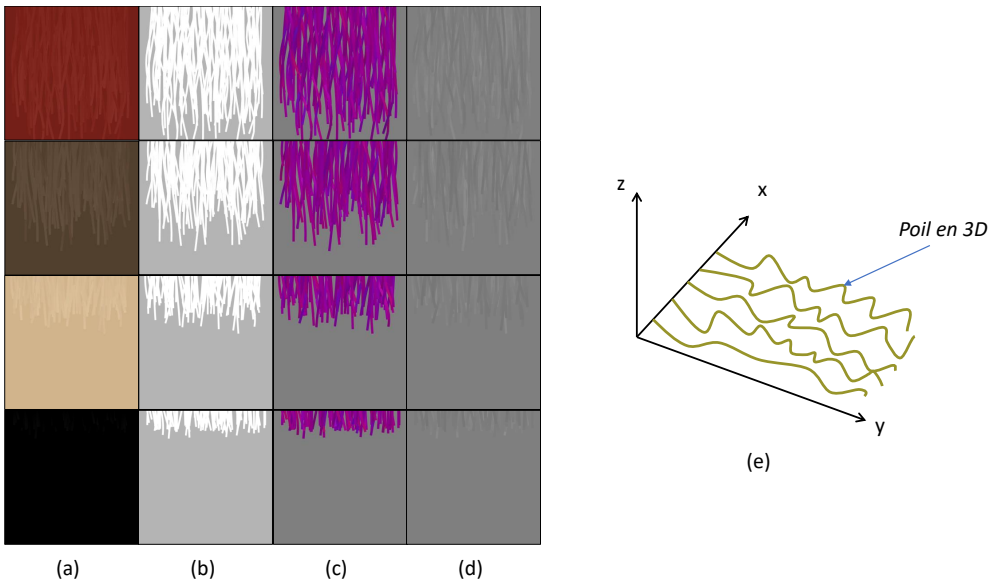

FIGURE 3.12 – Différentes textures générées à partir de la simulation des poils en 3D. (a) carte d'albédo, (b) carte alpha, (c) carte des tangentes, (d) carte d'ombrage. Dans (e), nous montrons un exemple de simulation des poils en 3D

Le *shader* de Kajiya-Kay [122] est conçu à l'origine pour simuler le rendu de fourrure. Aujourd'hui, il est largement utilisé pour le rendu graphique des cheveux humains [122–124] étant donné qu'elle prend bien en compte les propriétés optiques qui traduisent la nature des reflets émis de la chevelure. Pour modéliser finement l'apparence d'une chevelure, il faut prendre en compte la manière dont chaque poil renvoie ou absorbe la lumière incidente. Afin de simuler ses différents aspects, quatre types de textures sont nécessaires pour permettre un rendu réaliste. Ces cartes de textures sont : la carte d'albédo, la carte d'alpha, la carte des tangentes et la carte d'ombre (Voir figure 3.12). Nous montrons dans la figure 3.13, quelques exemples de plaquage de textures sur différents modèles de coiffures.

— La première carte est la carte d'albédo - *Albedo Map* (3 canaux) (Première colonne de la figure 3.12) : Cette texture contient les couleurs brutes qui donnent la première apparence des cheveux. Son effet est similaire à la carte d'albédo utilisée pour le modèle de la tête.

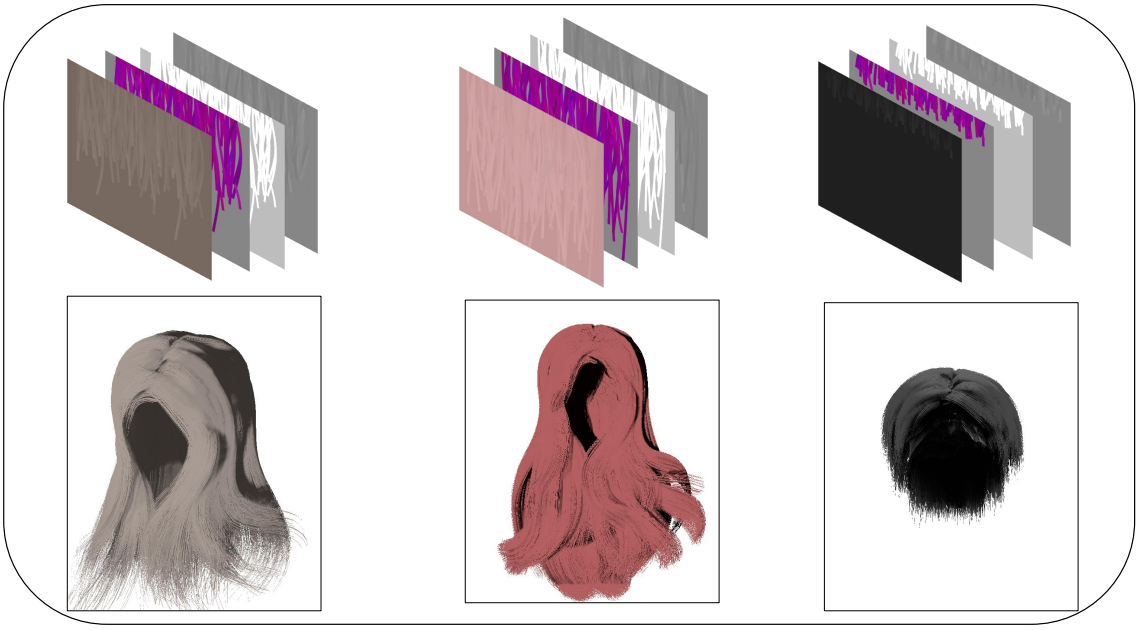

FIGURE 3.13 – Plaquage de différentes textures (Première ligne) sur des modèles de coiffures (Deuxième ligne).

— La deuxième carte est la carte alpha - *Alpha Map* (1 canal) (Deuxième colonne de la figure 3.12) : Cette texture est utilisée pour contrôler le degré de transparence [113]. Elle nous permet d'éviter la restitution de l'arrière-plan de couleur grise utilisé dans toutes les cartes.

— La troisième carte est la carte des tangentes - *Tangent Map* (3 canaux) (Troisième colonne de la figure 3.12) : En supposant que chaque poil 3D simulé apparaisse de la racine à la pointe, nous calculons sa tangente t_i en parcourant tous les sommets (Voir équation (3.2)). Ainsi chaque point projeté à partir d'un sommet est converti en un pixel qui représenté par la valeur de tangente sous forme de couleur. Cette carte est utilisée pour contrôler le déplacement du reflet spéculaire sur la longueur d'un poil.

— La quatrième carte est la carte d'ombre - *Shadow Map* (1 canal) (Quatrième colonne de la figure 3.12) : Cette carte est utilisée pour définir les ombres. En stockant la profondeur de chaque partie de la chevelure, elle permet de prendre en compte les propriétés globales qui concernent la projection des ombres à l'intérieur du modèle de coiffure. Cet effet est essentiel puisqu'il contribue de façon primordiale à l'impression de volume d'une chevelure.

Contrairement aux films d'animation qui utilisent une énorme puissance de traitement informatique pour simuler chaque mèche de cheveux sur une tête virtuelle, notre technique ne nécessite pas une puissance importante et elle permet d'avoir un rendu graphique de qualité supérieure. Nous simulons pour notre générateur 170 exemples de textures de cheveux avec dif-

férentes longueurs et couleurs. Chaque exemple d'entre eux est formé de quatre types de cartes décrits ci-dessus avec une résolution de 512×512. À chaque fois où nous générons un nouvel exemple de tête synthétique, nous choisissons aléatoirement un exemple de la base de données de texture. Dans la figure 3.14, nous montrons le rendu final de quelques exemples de modèles de coiffures.

FIGURE 3.14 – Exemples de modèles de coiffures texturées construites à partir de notre générateur de données.

3.3.4 Les lunettes

Une fois le modèle 3D d'une tête humaine synthétique est complet, nous choisissons de rajouter des modèles de lunettes en 3D pour générer quelques occlusions et donner un aspect plus réaliste au rendu final. Dans ce sens-là, nous avons pris cinq modèles de lunettes 3D texturés qui sont disponibles gratuitement dans un site-web[2] qui offre différents types d'objets 3D. Le recalage d'un exemple de lunettes avec une tête 3D est réalisé de la même façon que pour les yeux (Section 3.3.2). Pour ce faire, nous sélectionnons manuellement quatre paires de points entre le modèle 3DMM et le modèle 3D des lunettes. Ces points sont distribués comme ceci : pour le modèle de la tête (un point sur l'hélix de chaque oreille et deux points sur l'os nasal du nez qui définissent l'emplacement sur lequel repose les plaquettes des lunettes) et pour les lunettes (un point sur chaque manchon et un point sur chaque plaquette). Dans le but de créer un ensemble de données équitable, nous tirons au hasard une paire de lunette parmi les cinq modèles pour l'appliquer à une partie des têtes générées. Dans la figure 3.15, nous illustrons les différents modèles de lunettes utilisées dans (a) et le processus de recalage dans (b).

2. Source : https://free3d.com

FIGURE 3.15 – Exemples de quelques modèles de lunettes [2] utilisés dans notre générateur de données synthétiques.

3.4 Rendu final

En combinant les différentes parties que nous ayons constituées pour former un modèle de tête humaine réaliste, nous pouvons synthétiser un ensemble d'exemples 3D et obtenir ces images 2D correspondantes rendues à partir de différentes vues. Nous utilisons pour ceci le moteur de rendu graphique OpenGL qui permet de déclarer et afficher les différentes géométries et textures que nous avons modélisées dans les étapes précédentes. Il permet ensuite de réaliser les calculs nécessaires de projection, en vue de déterminer l'image à l'écran, en tenant compte de la distance, de l'orientation, des ombres, de la transparence, etc.

FIGURE 3.16 – Exemples d'images synthétiques de têtes humaines synthétisées à partir de notre générateur.

Pour générer des images synthétiques plausibles prises dans différentes vues aléatoires, il est

essentiel de contrôler correctement les paramètres d'orientation et de positionnement pour chaque exemple synthétisé. Dans ce contexte, nous initialisons tout d'abord une caméra dans une même position qui vise le centre de la tête généré. Ensuite, nous faisons tourner le modèle de tête de manière aléatoire pour simuler différentes poses en choisissant les trois paramètres de rotation (yaw, pitch, roll) de manière aléatoire, respectivement dans les intervalles $[-80°, 80°]$, $[-40°, 40°]$, $[-10°, 10°]$. Nous rajoutons finalement quelques décalages de translations dans le plan (XY) pour créer plus de diversité entre les images rendues. Tous ces paramètres sont échantillonnés à partir d'une distribution normale avec une valeur moyenne réglée pour assurer que l'image rendue I englobe la forme de la tête et qu'elle reste plus au moins centrée dans le plan.

Avec les différentes techniques de rendu 3D que nous utilisons, nous adaptons en plus de la lumière ambiante, deux sources d'éclairage ayant des directions uniformément échantillonnées dans la demi-sphère frontale de la tête. Similairement, pour les paramètres de brillance et de rugosité, au lieu d'utiliser des constantes pour toutes les têtes 3D, nous générons également de manière aléatoire un ensemble de paramètres pour chaque exemple. La dernière étape est la projection en perspective de la géométrie et les différentes ombres sur le plan de l'image. Plusieurs exemples de modèles synthétiques de têtes sont représentés dans la figure 3.16.

Pour pouvoir utiliser nos données synthétiques pour l'entraînement des réseaux de neurones. Il faut aussi générer d'autres types de cartes considérés comme donnés de vérité terrain. Ces cartes vont comporter des informations riches en caractéristiques géométriques de chaque tête générée. Elles vont servir aux deux processus de reconstruction 3D que nous présentons dans les chapitres 4 et 5. Dans cette perspective, nous présentons dans ce qui suit, les différents types de cartes générées avec chaque exemple d'un modèle 3D synthétique.

Carte de champ des normales En géométrie, la droite normale à une surface en un point est la droite orthogonale au plan tangent en ce point. Le vecteur directeur de cette droite est appelé vecteur normale à la surface. Par définition, pour une surface fermée, ce vecteur est unitaire et orienté vers l'extérieur.

Pour nos modèles 3D utilisés, nous calculons le champ des normales (tous les vecteurs normaux du maillage) grâce à l'équation de pondération angulaire (3.3) qui a été proposée dans [125].

$$\begin{aligned} \boldsymbol{n}_i &= \frac{1}{k}\sum_{j=1}^{k} \omega_j \frac{(\boldsymbol{q}_{i,j} - \boldsymbol{p}_i) \times (\boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i)}{||(\boldsymbol{q}_{i,j} - \boldsymbol{p}_i) \times (\boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i)||}, \\ \omega_j &= \arccos\left(\frac{\langle \boldsymbol{q}_{i,j} - \boldsymbol{p}_i, \boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i \rangle}{||\boldsymbol{q}_{i,j} - \boldsymbol{p}_i||\,||\boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i||}\right). \end{aligned} \tag{3.3}$$

Les valeurs normales $\boldsymbol{n}_i$ sont calculées pour chaque emplacement de sommet $\boldsymbol{p}_i \in \mathbb{R}^3$, étant donné l'ensemble des sommets $\{\boldsymbol{q}_{i,1}, \boldsymbol{q}_{i,2}, ..., \boldsymbol{q}_{i,k}\}$ qui sont adjacents à $\boldsymbol{p}_i$. Dans la figure 3.17, nous montrons les différents vecteurs du champ des normales calculés à partir de notre modèle déformable LYHM.

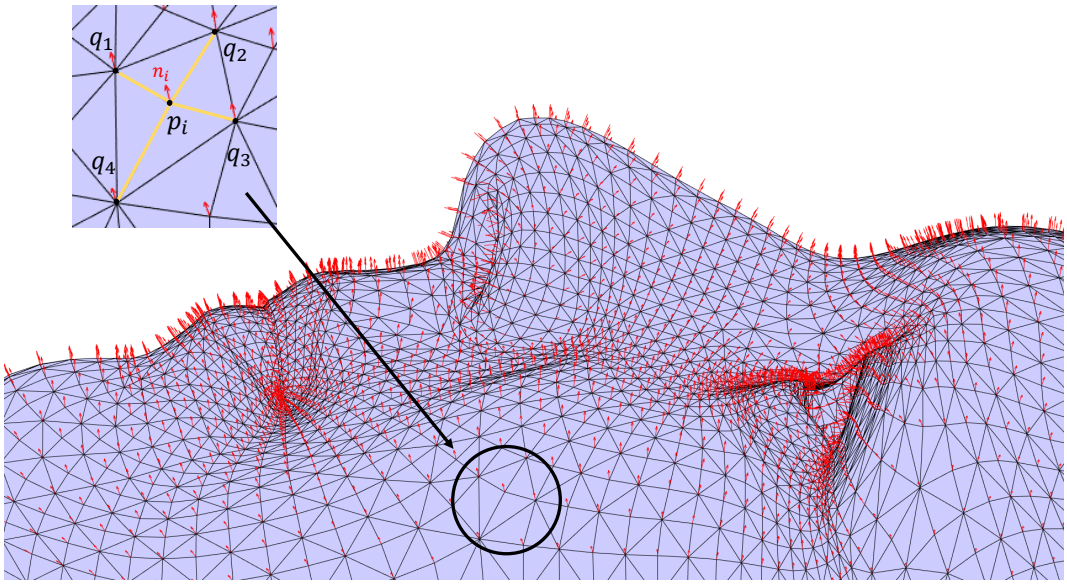

FIGURE 3.17 – Illustration d'un champ de normales calculé pour notre modèle déformable.

Pour chaque exemple de tête humaine synthétisé, nous calculons le champ de normales des modèles 3D de la tête et des yeux et nous projetons ensuite le résultat pour avoir la carte des normales $\mathcal{N}$ (3 canaux). Cette carte stocke la direction des normales directement dans les valeurs RVB d'une image. Un exemple de cette carte est illustré dans (f) de la figure 3.18. Ainsi, en utilisant le paramétrage de la surface par les coordonnées de l'image (u, v), le champ de normales dans la carte est défini de cette façon :

$$\begin{aligned} \mathcal{N} : \mathbb{R}^2 &\rightarrow \mathbb{S}^2 \subset \mathbb{R}^3 \\ (u, v) &\rightarrow \mathcal{N}(u, v) = [\mathcal{N}_x(u, v), \mathcal{N}_y(u, v), \mathcal{N}_z(u, v)]^\intercal \end{aligned} \tag{3.4}$$

où $\mathbb{S}^2$ désigne la sphère unité de $\mathbb{R}^3$. Bien que cette carte contienne un champ de normales similairement à la carte utilisée pendant la génération des données synthétiques (deuxième ligne de la figure 3.2). Nous notons qu'il existe des différences entre elles. D'abord, les normales de la carte $\mathcal{N}$ sont juste définies pour les parties qui incluent la peau d'un visage alors que pour l'autre carte, il s'agit d'une paramétrisation UV d'un modèle 3D et donc les normales sont définies sur toute la carte. Ensuite, nous intégrons la carte $\mathcal{N}$ dans nos approches de reconstruction 3D de visage dans les prochains chapitres alors que l'objectif derrière la carte utilisée dans notre

générateur de tête synthétique est de rajouter des détails à la forme 3D obtenue à partir du modèle LYHM.

Carte des points de repère Nous sélectionnons 24 sommets sur le modèle déformable de tête LYHM. Ces points de repère 3D sont situés à des endroits saillants de la tête telle que les yeux, le nez, la bouche, le menton et les oreilles. Après cela, nous projetons les points qui ne sont pas cachés par les cheveux ou à cause des larges poses dans le plan (XY) pour obtenir les des points projetés $\boldsymbol{b}_j$ avec j allant de 1 à 24. Nous générons ainsi la carte de repère composée de 24 canaux où chaque canal contient un point de repère $\mathcal{Z} : \mathbb{R}^2 \rightarrow \mathbb{R}^{24}$. Un exemple des points de repère est illustré dans (g) de la figure 3.18 (Points de repère indiquées par les carrés rouges).

Module du gradient de la carte de profondeur En utilisant le tampon de profondeur du moteur graphique OpenGL que nous utilisons pour rendre les différents modèles 3D, nous effectuons le rendu de la carte de profondeur des modèles géométriques de la tête et des yeux (Voir (h) de la figure 3.18). Cette carte contient les informations relatives à la distance des surfaces de ces différents objets à partir d'un point de vue. La distance des objets à la caméra est proportionnelle à la luminance de la carte. Plus le point 3D est proche, plus il est foncé et vice-versa.

En se basant sur cette carte de profondeur, nous calculons son gradient en utilisant les dérivées partielles :

$$\nabla f(u,v) = \begin{pmatrix} \nabla_u f \\ \nabla_v f \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial u}(u,v) \\ \frac{\partial f}{\partial v}(u,v) \end{pmatrix} \tag{3.5}$$

Puis nous déterminons le module de gradient $|\nabla f(u,v)|$ comme ceci :

$$\begin{gathered} \mathcal{W} : \mathbb{R}^2 \rightarrow \mathbb{R} \\ (u,v) \rightarrow \mathcal{W}(u,v) = |\nabla f(u,v)| = \sqrt{\frac{\partial f}{\partial u}(u,v)^2 + \frac{\partial f}{\partial v}(u,v)^2} \end{gathered} \tag{3.6}$$

Le module du gradient de la carte de profondeur $\mathcal{W}$ est illustrée dans (i) de la figure 3.18.

3.5 Conclusion

Les données synthétiques sont essentielles pour le développement des applications d'intelligence artificielle. De nombreuses applications nécessitent une annotation qui peut être coûteuse ou difficile à réaliser, d'autres types d'applications ont une large distribution de données sous-

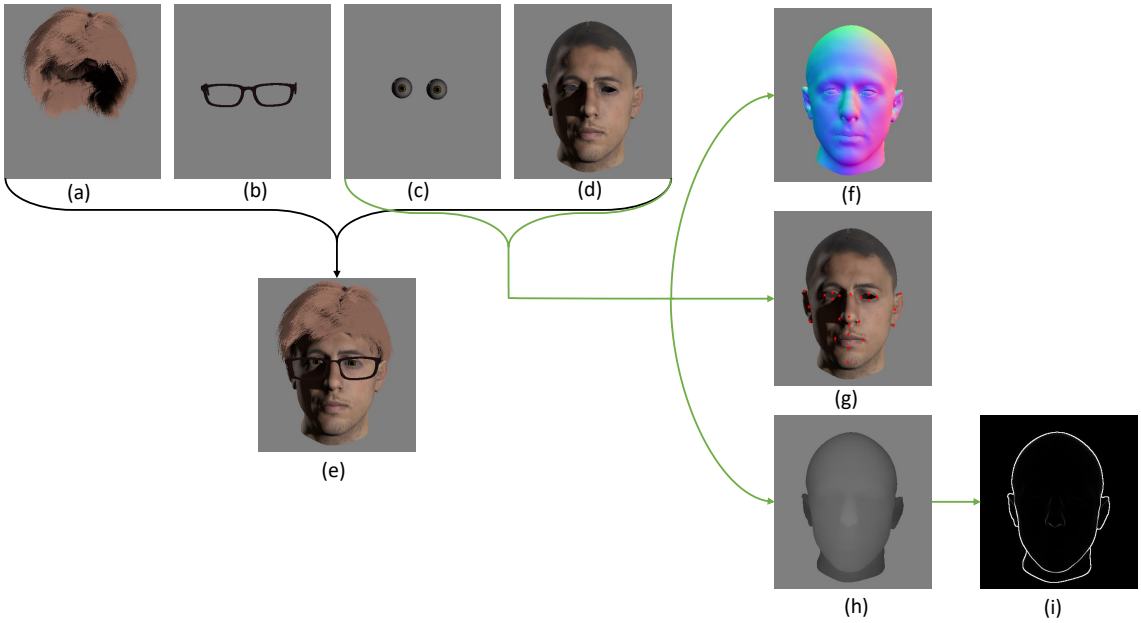

FIGURE 3.18 – À partir des différentes composantes géométriques (a-d), nous constituons une tête humaine synthétique (e). En utilisant seulement les modèles géométriques de la tête et des yeux, nous produisons les cartes : de champ des normales (f), de repères (h), de profondeur (h) et du module de gradient de la profondeur (i).

jacentes que les ensembles de données réelles ne couvrent pas. De ce fait, nous pensons que la génération de données synthétiques sera étendue et développée dans les années à venir.

Ce chapitre a présenté l'ensemble d'éléments combinés pour l'obtention de notre générateur de données synthétiques de visages 3D. La mise en place de ce générateur repose sur un modèle déformable 3DMM qui est un élément essentiel pour constituer la géométrie 3D d'une tête humaine. Au lieu d'utiliser le modèle paramétrique de texture du même modèle déformable, nous appliquons la technique d'*UV-mapping* en utilisant plusieurs textures de visages réalistes. Avec ceci, nous proposons de rajouter d'autres composants pour pouvoir synthétiser des exemples assez réalistes. Premièrement, nous rajoutons un modèle de yeux 3D fixes avec plusieurs textures pour pouvoir changer la couleur d'iris. Deuxièmement, nous alignons des coiffures 3D issues d'une base de données avec les modèles de têtes conçus. Après un processus d'augmentation, chaque modèle de coiffure a fait l'objet d'un traitement pour être transformé en maillage 3D. Juste après, nous créerons un ensemble de textures avec différentes caractéristiques pour les appliquer modèles de cheveux. Troisièmement, nous décidons d'utiliser quelques exemples de lunettes 3D pour composer des occlusions aux modèles générés. Finalement, un moteur de rendu graphique 3D a été utilisé pour établir le rendu final de chaque exemple formé. Avec ceci, nous générons un ensemble d'images qui sont utilisées comme vérité terrain pour l'apprentissage des réseaux de

neurones présenté dans les chapitres suivant.

Chapitre

4 Reconstruction de la surface faciale : intégration d'un champ vectoriel des normales

4.1 Introduction

Dans ce chapitre, nous présentons une nouvelle méthode de reconstruction de tête en 3D à partir d'une seule image d'entrée. Il s'agit de notre première approche hybride à base d'apprentissage profond et d'une technique géométrique. Nous introduisons un réseau neuronal profond formé uniquement sur des données synthétiques produites à partir de notre générateur décrit dans le chapitre précédent. Ce réseau produit deux différentes cartes à partir d'une seule image faciale d'entrée I. La première est une carte de champ des normales de la surface du visage $\mathcal{N}$ et la deuxième contient le module du gradient de la carte de profondeur du même visage $\mathcal{W}$. Ensuite, en utilisant les sorties du réseau, nous récupérons la géométrie faciale 3D en utilisant une technique d'intégration des normales établie grâce aux moindres carrés pondérés. Enfin, avec des tests d'évaluation qualitatifs et quantitatifs, nous montrons l'efficacité et la robustesse de notre approche proposée.

Nos principales contributions dans ce chapitre sont :

— Nous utilisons pour la première fois, nos données entièrement synthétiques de têtes humaines 3D composées de différents éléments pour former un réseau de neurones convolutif. Ces données ont été créées à partir de notre générateur de données décrit dans le chapitre 3,

— Un réseau de neurones composés d'un encodeur-décodeur et d'un discriminateur. Il s'agit d'une architecture proposée par [46] et qui s'inspire du modèle génératif GAN. Nous l'adaptons dans notre problématique pour prédire deux différentes cartes à partir de l'image fa-

ciale d'entrée I. Les deux cartes sont utilisées par la suite lors de l'étape de reconstruction,

— Nous proposons une reconstruction fiable de la tête utilisant une nouvelle technique d'intégration des normales basée sur une méthode des moindres carrés pondérés.

Dans ce chapitre, nous commençons tout d'abord par une présentation des méthodes d'estimation de champ des normales qui utilisent les CNN dans la section 4.2. Ensuite, nous introduisons le cadre général de notre méthode de reconstruction de tête 3D dans la section 4.3. Puis, nous consacrons la section 4.4 à la description du jeu de données construit à partir de notre générateur présenté dans le chapitre 3. Dans la section 4.5, nous présentons l'architecture de notre réseau *Face-Normal-Net* qui produit deux différentes cartes à partir d'une image faciale d'entrée. Les sorties de notre réseau sont ensuite utilisées dans un processus de reconstruction 3D que nous exposons dans la section 4.6. Après ceci, dans la section 4.7, nous examinons en premier lieu la qualité des cartes de normales produites par notre réseau et en deuxième lieu, nous étudions l'efficacité de notre méthode de reconstruction avant de l'évaluer qualitativement et quantitativement. Nous finissons ce chapitre par une conclusion dans la section 4.8.

4.2 État de l'art : Prédiction du champ des normales à base de CNN

Les cartes du champ des normales sont utilisées dans diverses applications graphiques telles que la reconstruction de formes 3D ou l'ajout de détails pour permettre au rendu des surfaces d'être plus réaliste. Par contre, la production des cartes du champ des normales de haute qualité pour des objets complexes est une tâche assez difficile. Pour résoudre ce problème, plusieurs travaux basés sur l'apprentissage profond ont été proposés.

Une partie de ces travaux est consacrée à la génération des cartes du champ des normales à partir de croquis. Dans le travail de Su *et al.* [46], une méthode interactive pour la génération des cartes de normales à partir d'un croquis a été proposée dans le cadre d'un réseau génératif conditionnel GAN. Dans cette architecture, le réseau U-Net [47] a été adapté pour permettre une transition de données en douceur. Puis, pendant le processus d'entraînement, l'utilisation de la distance de Wasserstein [83] a permis de fournir des informations précises et de réduire l'instabilité du processus. Ce réseau conditionnel est contrôlé par des points spécifiés par l'utilisateur. Ils permettent une maîtrise plus directe lors de la génération des cartes du champ des normales et éliminent efficacement l'ambiguïté de la représentation du croquis.

Dans une autre approche [126], les auteurs ont utilisé un réseau ConvNet (CNN) pour prédire la profondeur et les cartes du champ des normales à partir de plusieurs croquis générés dans différentes vues. Ensuite, les sorties du réseau sont combinées en un seul nuage de points 3D via la minimisation d'énergie.

Un autre travail basé sur des croquis a été proposé par [127], où les auteurs ont présenté une technique pour prédire directement des images contenant des champs de normales à haute résolution sans aucune interaction de l'utilisateur. En utilisant une représentation multi-échelle de leurs images d'entrée, ils ont souligné l'efficacité et la qualité de leurs données produites. Alors que leur travail se concentre uniquement sur la reconstruction des cartes de normales, leur réseau utilisé peut être étendu pour reconstruire également la profondeur et donc des modèles 3D complets.

Une autre catégorie de méthodes a été proposée pour prédire les cartes du champ des normales à partir de différents objets ou scènes (extérieur/intérieur).

Par exemple, dans [128], un modèle appelé *Skip-Network* a été proposé pour la prédiction des champs de normales à partir des scènes d'intérieur. Qiu *et al.* [129] utilisent un réseau de neurones pour estimer la normale de surface à partir d'une image (RVB) et des données *LiDAR* et utilisent en outre ces normales de surface récupérées pour guider la reconstitution de la profondeur. Wang *et al.* [130] ont séparé leur réseau en deux processus, le premier pour estimer une structure grossière de l'image d'entrée et la seconde est consacrée à la prédiction d'un patch local plus fin. Au final, un réseau de fusion a été utilisé pour le regroupement des deux sorties.

De manière similaire à ce que nous présentons dans ces travaux de thèse, plusieurs chercheurs ont proposé des approches pour générer des cartes du champ des normales à partir d'images faciales. Par exemple, Trigeorgis *et al.* [65] estiment le champ des normales à partir des images faciales en utilisant un réseau entièrement convolutif (CNN). Ce dernier a été formé à partir d'une combinaison de données réelles et synthétiques. Pendant l'entraînement, les auteurs ont utilisé une fonction de coût basée sur le cosinus afin de minimiser la distance angulaire entre les prédictions du réseau et les champs de normales de vérité terrain. Finalement, ils ont utilisé la méthode standard de Frankot-Chellappa [91] pour reconstruire une géométrie faciale grossière à partir des champs de normales prédites.

Dans une autre méthode, l'architecture *SfSNet* [99] qui s'est inspirée d'un modèle de rendu physique, a été proposée pour produire une décomposition précise d'une image d'un visage humain. Cette décomposition est constituée de trois différentes images : champ des normales, al-

bédo et l'intensité de l'éclairage. La base d'apprentissage du réseau est constituée d'un mélange d'images synthétiques étiquetées et d'images réelles non étiquetées. Ce modèle consiste en une nouvelle architecture de décomposition avec des blocs résiduels qui apprennent une séparation complète de l'albédo et du champ des normales.

Une approche similaire à [99], a été proposée par Shu *et al.* [131] où un réseau génératif (GAN) entraîné de bout en bout, produit une carte de champ des normales, une carte pour l'albédo et une pour l'éclairage, mais aussi un masque alpha. Le réseau a été formé à partir d'une base d'images réelles sans vérité terrain, avec des fonctions de coût appropriées. Cette architecture, basée sur le rendu physique, permet une édition de visage réaliste tout en préservant les propriétés d'identités.

Plus récemment, Abrevaya *et al.* [132] ont proposé une nouvelle architecture d'un réseau encodeur-décodeur profond. La contribution principale derrière leur approche repose sur l'exploitation de toutes les bases de données d'images faciales disponibles, ce qui leur permet d'utiliser de nombreux grands ensembles de données d'images qui ne possèdent pas de cartes du champ de normales de vérité terrain. Leur architecture dépend de deux réseaux encodeurs-décodeurs avec un espace latent partagé. Cet espace assure le transfert des détails de visage entre les domaines de l'image RVB et de la carte de champ des normales via des connexions par saut entre l'encodeur de l'image d'entrée et le décodeur de la carte de champ des normales. À la fin de l'entraînement du réseau, les cartes prédites ont été utilisées pour améliorer une géométrie 3D grossière obtenue de la méthode PRN [8].

4.3 Notre proposition

Dans cette section, nous décrivons les détails de notre cadre proposé. Notre méthode prend en entrée une image faciale I, puis en utilisant notre réseau *Face-Normal-Net*, nous créerons deux sorties qui sont totalement alignées, à l'échelle du pixel, avec l'image d'entrée. Ces deux différentes cartes sont : la carte de champ des normales $\mathcal{N}$ et le module du gradient de la carte de profondeur $\mathcal{W}$. Ces deux sorties sont ensuite utilisées dans un algorithme de reconstruction 3D guidé par $\mathcal{W}$ qui permet de récupérer la surface 3D du visage. Tout d'abord, nous introduisons l'ensemble de nos données synthétiques utilisées pour l'étape d'entraînement dans la section 4.4 et qui ont été créées à partir de notre générateur de données synthétiques présenté dans le chapitre 3. Dans la section 4.5, nous décrivons l'architecture de notre réseau qui a été inspirée

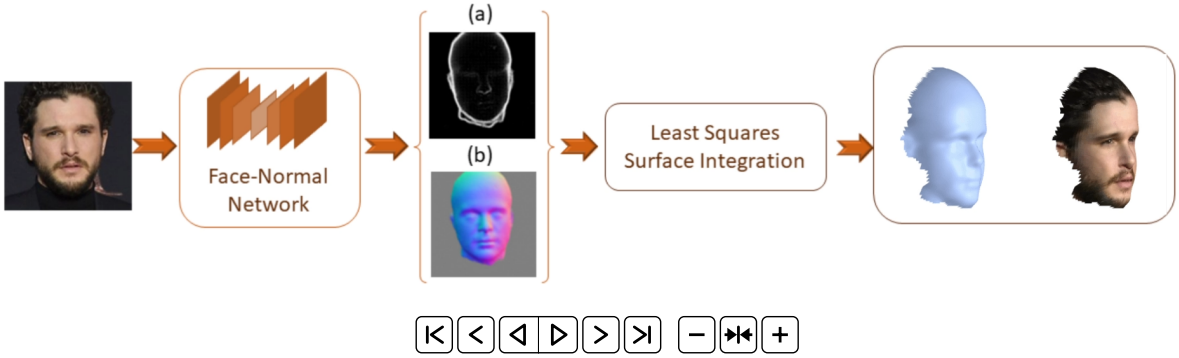

FIGURE 4.1 – Le pipeline de notre méthode proposée. Étant donné une image faciale d'entrée I, nous estimons deux cartes différentes (module du gradient de la carte de profondeur $\mathcal{W}$ (a), carte de champ des normales $\mathcal{N}$ (b)) à travers un réseau qui a été formé à partir d'un ensemble de données entièrement synthétiques. En utilisant ces cartes générées, nous reconstruisons la forme du visage 3D par une technique d'intégration des normales à base des moindres carrés pondérés où $\mathcal{W}$ agit comme une carte de poids.

de [46]. Enfin, les détails de notre étape de reconstruction sont expliqués dans la section 4.6. Nous illustrons dans la figure 4.1 les différentes parties qui constituent notre approche.

4.4 Données d'entraînement : *Face-Normal-Net*

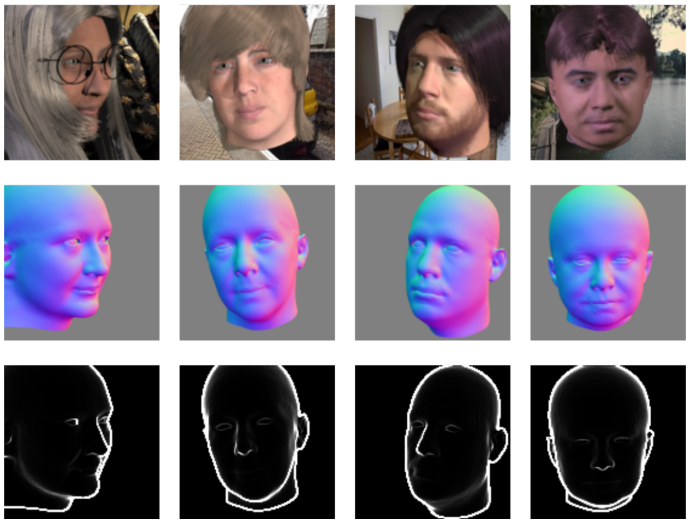

FIGURE 4.2 – Quelques exemples de données d'entraînement. Du haut en bas : images faciales synthétiques I, cartes du champ des normales $\mathcal{N}$, module du gradient de la carte de profondeur $\mathcal{W}$.

Comparé au travail [65], qui était basé sur un mélange de données synthétiques et réelles pour former le réseau, notre modèle proposé a été formé uniquement avec des données synthétiques. Quelques exemples du jeu de données d'entraînement utilisées pour la formation de notre réseau sont illustrés dans la figure 4.2. Le générateur de données synthétiques présenté dans le chapitre

3, nous a permis de constituer l'ensemble des données de l'entraînement. En plus de l'image du visage, nous utilisons aussi l'élément essentiel de notre reconstruction faciale qui est la carte de champ des normales calculé uniquement pour la tête et les yeux. Ce calcul a été réalisé en utilisant la méthode de pondération angulaire de [125] décrite dans l'équation (3.3). Le choix s'est porté sur l'utilisation des normales pour diverses raisons. Tout d'abord, les composantes du champ des normales définissent des propriétés géométriques locales et donc dissociées les unes des autres sur certaines distances, en ce sens que nous pouvons les prédire de manière totalement indépendante, contrairement aux valeurs de profondeur qui devraient être prédites toutes ensemble. En d'autres termes, sans connaître la valeur de profondeur du bout du nez, par exemple, nous ne pouvons pas prédire la valeur pour les yeux. Deuxièmement, les normales sont invariables à la translation et à l'échelle. Pour améliorer la qualité de reconstruction 3D, nous proposons aussi d'utiliser le module de gradient de la carte de profondeur $\mathcal{W}$ (troisième ligne de la figure 4.2). L'effet de l'utilisation de cette carte est analysé dans la section 4.6.

4.5 Architecture du réseau : *Face-Normal-Net*

Notre réseau de neurones est une architecture de transformation d'image en image qui est inspirée du modèle proposé dans l'approche de Su *et al.* [46]. Dans leur approche, un réseau génératif conditionnel CGAN est formé pour créer une carte de champ des normales à partir d'un croquis et d'un masque de point binaire. En utilisant la configuration conditionnelle, les distributions de données que le générateur G et le discriminateur D essaient d'approximer deviennent des distributions conditionnelles [133] où l'information supplémentaire est concaténée avec le vecteur latent z dans l'entrée du réseau. En ce qui concerne le problème de transformation d'image en image, l'introduction d'une image d'entrée qui guide le générateur G est une opération assez intuitive pour la formation du réseau, ainsi au lieu d'utiliser un bruit aléatoire à l'entrée de réseau, les auteurs de [46] ont choisi d'utiliser des images d'entrées qui constituent un moyen efficace d'incorporer les informations du croquis dans leur méthode.

Quelques modifications ont été apportées à ce réseau pour l'adapter à notre problématique (Voir les détails dans la figure 4.3). Dans ce contexte, notre modèle est composé d'un réseau générateur adapté à partir de l'architecture U-Net [47] et d'un discriminateur. Toutes les images d'entraînement ont une taille de 128×128 pixels. En entrée, nous empilons trois images, l'image du visage (RVB) I, $\mathcal{N}$ (trois canaux) et $\mathcal{W}$ (canal unique), alors que nous n'avons que deux

cartes générées.

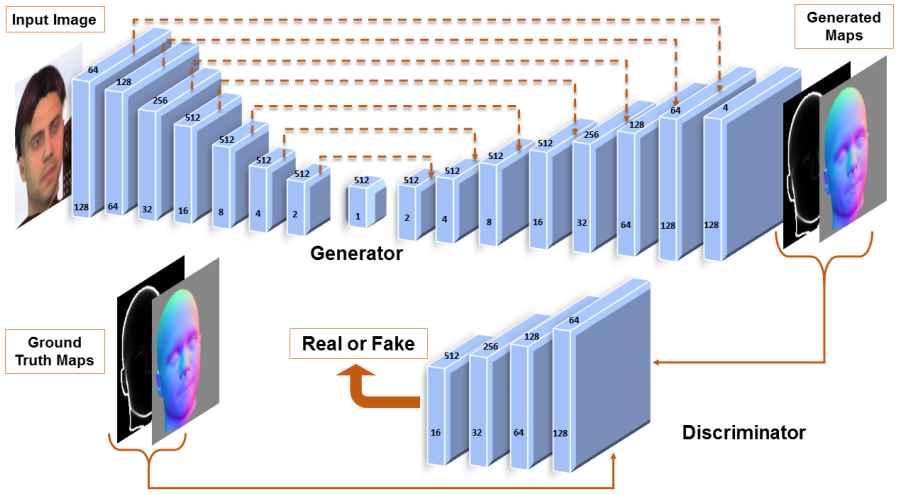

FIGURE 4.3 – Notre architecture proposée vise à générer deux cartes à partir d'une image d'entrée faciale. Les données d'entrée, comme indiqué à gauche, sont composées d'une image d'entrée faciale I et de deux cartes de vérité terrain : $\mathcal{N}$ et $\mathcal{W}$. L'entrée de l'encodeur-décodeur est l'image du visage, tandis qu'à la sortie, il produit deux cartes différentes (illustrées à droite). Après cela, nous injectons la vérité terrain et les cartes générées avec l'image du visage comme entrée du discriminateur. Dans cette étape, nous vérifions si les cartes générées sont réelles ou fausses, nous encourageons donc l'encodeur-décodeur à produire des cartes plus réalistes en fonction de l'entrée d'image du visage. La taille spatiale et le nombre de couches sont indiqués respectivement dans et au-dessus de chaque bloc.

L'architecture de notre générateur G est un encodeur-décodeur composé de 16 couches. Ce type d'architecture est favorisé pour de nombreux problèmes d'apprentissage à base d'image. Il permet d'extraire les informations d'entrée (les caractéristiques du visage dans notre cas), puis de déduire et produire les informations du champ des normales et du module de gradient de profondeur sur la base de la représentation extraite en bas niveau de l'entrée. De ce fait, l'image passe progressivement par des couches de sous-échantillonnage puis de sur-échantillonnage. Pendant que les éléments de l'encodeur du générateur G sont similaires à celles du discriminateur D, les couches du décodeur sont composées d'une déconvolution, d'une normalisation de batch, d'un dropout et d'une fonction d'activation de type $ReLU$. Le discriminateur est composé de 4 couches où chacune d'entre elles est constituée d'une convolution, d'une normalisation de batch et d'une fonction d'activation $ReLU$. Pour réduire la perte d'informations entre les couches successives, nous utilisons une connexion de saut pour les couches symétriques du générateur comme il a été proposé dans l'architecture U-Net [47]. Plus précisément, les connexions sont ajoutées après la normalisation de batch dans le générateur G entre chaque couche i et la couche $n-i$, où n est le nombre total de couches dans G. À la sortie du générateur G, nous utilisons une fonction

d'activation de type tangente hyperbolique *tanh* puisque les valeurs de la carte de champ des normales se situent dans la plage $[-1, 1]$.

En ce qui concerne la formation du réseau, nous adoptons la fonction coût du GAN conditionnel utilisée dans [46] pour nos besoins, telle que définie ci-dessous :

$$\min_G \max_D \mathbb{L}_{CGAN}(G, D) = \mathbb{E}_{x \sim p_{data}, y \sim p_m}[D(y \mid x)] - \mathbb{E}_{\tilde{y} \sim p_{gen}}[D(\tilde{y} \mid x)] \tag{4.1}$$

Où x représente l'image du visage d'entrée, y représente les cartes d'entrée concaténées correspondantes, $\tilde{y}$ représentent les sorties générées. p_{data}, p_m et p_{gen} sont respectivement les distributions des données d'entrée réelles, de la carte d'entrée et des données de sortie générées. Le discriminateur D essaie de distinguer les images prédites $\tilde{y}$ des images réelles y en maximisant la fonction objectif, tandis que le générateur G essaie de tromper le discriminateur en minimisant la fonction objectif.

Pour éviter que $\tilde{y}$ ne soit éloigné de la vérité de terrain y, nous utilisons également une fonction coût de distance L_2 pour contraindre davantage le générateur G et superviser le processus d'apprentissage :

$$\mathbb{L}_2(G) = \mathbb{E}_{y \sim p_m, \tilde{y} \sim p_{gen}}[||y - \tilde{y}||_2] \tag{4.2}$$

Enfin, la fonction objective finale de notre réseau est composée de deux termes avec un paramètre de pénalisation λ_1 pour la fonction L_2. Nous visons à former un générateur G^* optimal pour satisfaire la fonction objective ci-dessous :

$$G^* = \arg\min_G \max_D \mathbb{L}_{CGAN}(G, D) + \lambda_1 \mathbb{L}_2(G) \tag{4.3}$$

4.6 Reconstruction 3D : Intégration robuste des normales

Notre solution de reconstruction 3D est basée sur l'intégration robuste des normales. Le processus de reconstruction prend en entrée les deux cartes $\mathcal{N}$ (carte de champ des normales) et $\mathcal{W}$ (module du gradient de la carte de profondeur). Ces deux cartes sont prédites à partir de notre réseau *Face-Normal-Net* décrit dans la section précédente.

L'objectif derrière cette intégration des normales est de retrouver la carte de profondeur caractéristique du relief de la surface faciale. Dans cette section, nous introduisons tout d'abord

le problème général d'intégration des normales dans la section 4.6.1. Ensuite nous présentons dans 4.6.2, notre méthode d'intégration robuste à base des moindres carrés pondérés qui est adaptée aux discontinuités de profondeur.

4.6.1 Présentation du problème

Connaissant que la surface d'un visage est constituée d'un ensemble de sommets $\boldsymbol{p} = [x, y, z]^\intercal \in \mathbb{R}^3$ dont les coordonnées sont définies relativement à un repère tridimensionnel (XYZ), les axes X et Y constituent le plan parallèle au plan image et l'axe Z est parallèle à l'axe optique. En projetant cette surface, chaque point visible est en bijection avec un point image $(u, v) \in \mathbb{R}^2$. Mentionnons que les coordonnées (u, v) d'un point image sont définies dans un repère bidimensionnel dont les axes sont parallèles aux axes X et Y. Ainsi, cette surface faciale est paramétrée sur un domaine $\Omega \subset \mathbb{R}^2$ que nous appelons domaine de reconstruction et qui définit la projection de la surface sur le plan image :

$$\mathrm{x}\,(u, v) = [\, u, v, h\,(u, v)\,]^\intercal \tag{4.4}$$

Par conséquent, si nous connaissons $h(u, v)$, nous pouvons déterminer le point objet $\mathrm{x}(u, v)$ sans ambiguïté. Dans ce cas, $h : \mathbb{R}^2 \to \mathbb{R}$ représente la carte de profondeur de la partie visible de la surface, qui associe à chaque point image (u, v) la troisième coordonnée cartésienne du point objet qui lui est conjugué. De ce fait, le problème de reconstruction 3D dans ce cas consiste à associer à chaque point image (u, v) une profondeur $h(u, v)$.

4.6.1.1 Relation entre la normale et le gradient de profondeur

Le problème d'intégration d'un champ de normales consiste à estimer, en chaque pixel $(u, v) \in \Omega$, le point $\mathrm{x}(u, v)$ de la surface faciale, connaissant sa normale $\mathcal{N}(u, v)$.

Partant de l'équation (4.4) qui définit la projection de la surface faciale, le plan tangent à cette surface en un point (u, v) est porté par les deux vecteurs $\partial_u \mathrm{x}(u, v)$ et $\partial_v \mathrm{x}(u, v)$ dont le produit vectoriel est :

$$\partial_u \mathrm{x}(u, v) \times \partial_v \mathrm{x}(u, v) = \begin{bmatrix} -\partial_u h(u, v) \\ -\partial_v h(u, v) \\ 1 \end{bmatrix} \tag{4.5}$$

Le vecteur résultant de ce produit vectoriel est colinéaire au vecteur de la normale, donc son produit vectoriel avec $\mathcal{N}(u,v) = [\mathcal{N}_x(u,v), \mathcal{N}_y(u,v), \mathcal{N}_z(u,v)]^\intercal$ est nul, ce qui constitue un système linéaire de trois équations à deux inconnues :

$$\begin{cases} \mathcal{N}_z(u,v)\,\partial_u h(u,v) = -\mathcal{N}_x(u,v) \\ \mathcal{N}_z(u,v)\,\partial_v h(u,v) = -\mathcal{N}_y(u,v) \\ \mathcal{N}_y(u,v)\,\partial_u h(u,v) - \mathcal{N}_x(u,v)\,\partial_v h(u,v) \end{cases} \tag{4.6}$$

qui sont les dérivées partielles de la profondeur $\partial_u h(u,v)$ et $\partial_v h(u,v)$. Par ce moyen, le problème d'intégration consiste à déterminer $\nabla h(u,v) = [\partial_u h(u,v),\ \partial_v h(u,v)]^\intercal$ à partir de $\mathcal{N}(u,v)$. Ce faisant, l'équation liant $h(u,v)$ à $\mathcal{N}(u,v)$ est l'équation linéaire aux dérivées partielles (EDP) suivante :

$$\nabla h(u,v) = \begin{bmatrix} -\frac{\mathcal{N}_x(u,v)}{\mathcal{N}_z(u,v)} \\ -\frac{\mathcal{N}_y(u,v)}{\mathcal{N}_z(u,v)} \end{bmatrix} \tag{4.7}$$

Et en supposant le gradient de profondeur G_p basé sur $\mathcal{N}$, nous obtenons :

$$\begin{gathered} G_p : \mathbb{R}^2 \to \mathbb{R}^2 \\ G_p(u,v) = \begin{bmatrix} \mathrm{p}(u,v) \\ \mathrm{q}(u,v) \end{bmatrix} = \begin{bmatrix} -\frac{\mathcal{N}_x(u,v)}{\mathcal{N}_z(u,v)} \\ -\frac{\mathcal{N}_y(u,v)}{\mathcal{N}_z(u,v)} \end{bmatrix} \end{gathered} \tag{4.8}$$

Ainsi, à partir de (4.7) et de (4.8), la résolution de l'équation suivante permet de reconstruire le relief de la surface observée :

$$\nabla h(u,v) = G_p(u,v) \tag{4.9}$$

4.6.2 Notre méthode d'intégration robuste

D'après [134], la résolution directe de l'équation (4.9) n'est pas possible, car le champ G_p n'est jamais facilement intégrable à cause du bruit et des discontinuités de profondeur. Afin de remédier à ces difficultés, le moyen le plus simple consiste à résoudre le problème en utilisant les moindres carrés [92, 134–137].

Dans notre approche, nous introduisons $G_p(u,v)$ et $\mathcal{W}(u,v)$ dans un solveur des moindres

carrés pondérés défini dans le domaine continu comme suit :

$$\begin{aligned} \min_{h} \iint_{(u,v)\in\Omega} w(u,v)||\nabla h(u,v) - G_p(u,v)||^2 \mathrm{d}u\mathrm{d}v \\ w(u,v) = \frac{1}{1+\lambda\mathcal{W}(u,v)} \end{aligned} \tag{4.10}$$

où $w(u,v)$ est le terme de poids utilisé pour imposer la conformité de la surface reconstruite avec le terme de gradient à proximité des discontinuités de la surface faciale. Plus précisément, cette discontinuité se présente dans le cas où $G_p(u,v)$ constitue une donnée aberrante en raison de la valeur de $\mathcal{N}(u,v)$ qui est non définie. Il s'agit d'un pixel (u,v) pour lequel $\mathrm{x}(u,v)$ est situé sur une arête ou une discontinuité de profondeur.

Dans (4.10), λ est un paramètre critique à régler (voir les détails dans la section 4.7.3). Dans cette étape, nous estimons les valeurs de la carte de profondeur d'une fonction $h : \mathbb{R}^2 \to \mathbb{R}$ dans le domaine de reconstruction Ω où G_p et $\mathcal{W}$ sont définis.

Cadre discret

Dans le cadre discret, notre problème de minimisation est formulé comme ci-dessous :

$$\min_{z} \sum_{u,v\in\Omega} w_{u+0.5,v}(h_{u+1,v} - h_{u,v} - \mathrm{p}_{u+0.5,v})^2 + w_{u,v+0.5}(h_{u,v+1} - h_{u,v} - \mathrm{q}_{u,v+0.5})^2 \tag{4.11}$$

où $(u+0,5,v)$ et $(u,v+0,5)$ sont les moyennes des points entre deux pixels successifs le long des axes horizontal et vertical respectivement.

Par exemple, le gradient entre les pixels est défini de cette manière :

$$\begin{aligned} \mathrm{p}_{u+0.5,v} &= \frac{1}{2}(\mathrm{p}_{u,v} + \mathrm{p}_{u+1,v}) \\ \mathrm{q}_{u,v+0.5} &= \frac{1}{2}(\mathrm{q}_{u,v} + \mathrm{q}_{u,v+1}) \end{aligned} \tag{4.12}$$

Une illustration de notre méthode de discrétisation sur une grille 2D carrée régulière est représentée dans la figure 4.4. Il s'agit d'un problème linéaire classique des moindres carrés, nous le résolvons en utilisant le solveur Ceres [138]. L'évaluation de la qualité de reconstruction est décrite dans la section 4.7.3.

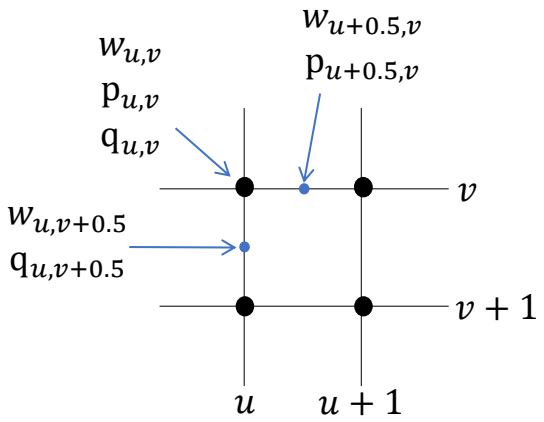

FIGURE 4.4 – Illustration de notre méthode de discrétisation.

4.7 Évaluation

Cette section présente trois séries d'expériences qui ont été menées pour évaluer la performance de notre méthode de reconstruction 3D. Tout d'abord, nous discutons de la qualité de notre modèle d'entraînement dans la section 4.7.2. Ensuite, nous montrons notre contribution à la méthode de reconstruction et l'avantage qu'elle apporte par rapport aux autres méthodes d'intégration de champ des normales dans la section 4.7.3. Afin d'analyser les performances du pipeline dans son ensemble, nous avons réalisé une expérience qualitative sur un ensemble d'images de célébrités dans la section 4.7.4 et une expérience quantitative sur un ensemble de données faciales en 3D pour tester sa précision dans la section 4.7.5.

4.7.1 Données de test

4.7.1.1 Données synthétiques

Pour évaluer la qualité des cartes prédites à partir de notre modèle entraîné et également de notre méthode de reconstruction 3D, nous produisons un jeu de données de test à partir de notre générateur décrit dans le chapitre 3.

Ce jeu de données de test se compose de 200 images faciales équitablement réparties entre hommes et femmes. Pour chaque exemple d'un visage, nous générons aussi les cartes correspondantes qui sont considérées comme vérité terrain. Il s'agit de la carte du champ des normales $\mathcal{N}$, la carte de profondeur et le module du gradient de la carte de profondeur $\mathcal{W}$. Nous soulignons que dans notre évaluation, la profondeur est mesurée en unités équivalentes à la taille des pixels.

4.7.1.2 Données réelles

Afin d'évaluer l'efficacité de nos approches de reconstruction 3D présentées dans ce chapitre et dans le chapitre 5, nous utilisons la base de données BU-3DFE [10]. Cette base est une collection d'images faciales avec leurs modèles 3D correspondants. Elle contient 100 sujets (56 femmes, 44 hommes) avec différentes expressions faciales, y compris l'expression neutre et six types d'expressions générales à quatre niveaux d'intensité. Les différents sujets sont âgés de 18 à 70 ans et possèdent une large variété ethnique/raciale. Les visages 3D de cette base sont acquis à l'aide d'un scanner 3D d'une résolution d'environ $8K$ et d'une précision d'environ $0.2mm$. À chaque modèle de forme, sont associées trois images faciales correspondantes, dont deux d'entre elles sont capturées en deux vues (environ $+45°$ et $-45°$) sous une configuration contrôlée. Compte tenu de cette diversité, cette base de données a été largement utilisée dans plusieurs domaines de vision par ordinateur.

Dans notre contexte d'évaluation, le modèle déformable que nous utilisons (LYHM) pour générer les données synthétiques, ne contient pas d'expressions faciales. Pour cela, nous nous contentons uniquement d'utiliser les images faciales avec l'expression neutre de la base BU-3DFE avec leurs modèles 3D de vérité terrain correspondant. Nous utilisons ces données de test pour examiner la capacité de généralisation de nos approches de reconstruction de visage 3D à partir d'image.

4.7.2 Évaluation de l'entraînement

Pour entraîner notre modèle, nous avons utilisé 40 000 images faciales (20 000 pour les hommes et aussi pour les femmes) et leurs cartes correspondantes. Nous entraînons le modèle pour environ 2000 époques avec un pas d'apprentissage de $1e-4$, 64 en taille de batch, 500 pour λ_1 et en utilisant l'optimiseur RMSprop. Pour éviter le sur-apprentissage des données, pendant le processus d'apprentissage et pour chaque image d'entrée I, nous ajoutons un effet de flou aléatoire et du bruit gaussien ainsi qu'une image d'arrière-plan tirée au hasard de l'ensemble de données COCO [139].

Pour évaluer notre modèle de transformation d'image-à-image, nous montrons dans la figure 4.5 différents exemples avec la vérité terrain et les cartes produites par notre modèle *Face-Normal-Net*. On peut bien remarquer qu'avec l'utilisation de nos données synthétiques qui contiennent des occlusions (cheveux, lunettes, arrière-plans aléatoires), le réseau réussit à

TABLE 4.1 – Évaluations du masque et du champ des normales pour l'ensemble de données de test. Nous montrons dans la partie supérieure du tableau nos résultats de segmentation en utilisant les pourcentages de la précision et du rappel. La deuxième partie contient les résultats d'erreur angulaire (moyenne et écart-type) et les pourcentages d'erreurs inférieures à différents seuils.

Mask Evaluation	
précision	93.72 %
rappel	98.40 %
Normals Evaluation	
Mean	10.01°
Std	12.45°
$< 10°$	67.50 %
$< 20°$	92.65 %
$< 30°$	97.13 %

séparer correctement toute la tête. Dans les situations où il y a des parties couvertes, le réseau tente de prédire une forme plus approximative de la morphologie précise cachée par les cheveux dans la plupart des cas. À l'aide de la base de données synthétiques de test , nous avons effectué des expériences pour évaluer la précision de notre réseau. Le tableau 4.1 montre les résultats indiquant un pourcentage significatif de *précision* et *rappel*, ce qui explique que la plupart des pixels produits par le réseau correspondent aux pixels des cartes de vérité terrain. Dans un second temps, nous avons évalué la précision sur les cartes des normales et pour cela, nous avons calculé la moyenne et l'écart-type de l'erreur angulaire entre les cartes de vérité terrain $\mathcal{N}_{\text{GT}}$ et celles produites par le réseau $\mathcal{N}$. L'erreur angulaire entre deux vecteurs $\mathcal{N}(u,v)$ et $\mathcal{N}_{\text{GT}}(u,v)$ est calculée comme ceci :

$$Err_{Ang}(u,v) = \arccos\left(\frac{\langle \mathcal{N}_{\text{GT}}(u,v)^{\intercal}\,,\, \mathcal{N}(u,v)\rangle}{||\mathcal{N}_{\text{GT}}(u,v)||\;||\mathcal{N}(u,v)||}\right). \tag{4.13}$$

Nous montrons aussi le pourcentage de pixels à différents seuils d'erreur angulaire comme il a été présenté dans [65, 99, 132].

4.7.3 Analyse de l'efficacité de notre méthode

En utilisant l'ensemble de données de test, nous évaluons dans cette étape, l'efficacité de notre méthode proposée dans la reconstruction de visage 3D. Pour ce faire, nous décrivons tout d'abord la procédure de réglage du paramètre λ de l'équation (4.10) pour optimiser notre méthode de reconstruction. Deuxièmement, nous montrons une comparaison entre la vérité terrain et la carte de profondeur reconstruite en utilisant la valeur optimale du paramètre λ qui minimise l'erreur

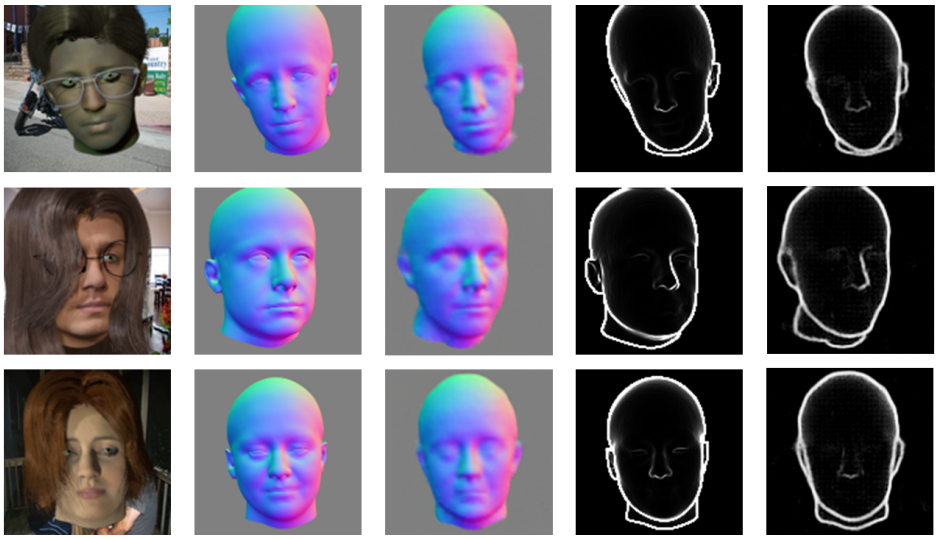

FIGURE 4.5 – Comparaison entre la vérité terrain et les estimations des cartes $\mathcal{N}$ et $\mathcal{W}$. La première colonne contient l'image d'entrée du visage I, la deuxième et la quatrième colonne contiennent des cartes de vérité terrain, la troisième et la cinquième contiennent des cartes estimées.

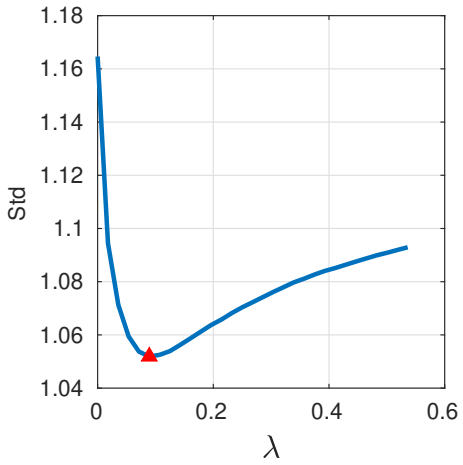

FIGURE 4.6 – Précision de la reconstruction. L'effet de variation de la valeur du paramètre λ sur la précision en termes d'écart-type sur l'ensemble de données de test.

de reconstruction. Une fois la profondeur reconstruite, nous la comparons à la vérité terrain, disponible dans le jeu de données de test. La procédure d'évaluation utilisée pour trouver la valeur optimale du paramètre λ est réalisée comme suit :

Étape 1: Nous calculons tout d'abord pour chaque exemple, la carte d'erreur entre l'image de profondeur de vérité terrain et celle reconstruite à partir des cartes $\mathcal{N}$ et $\mathcal{W}$ correspondantes :

$$Err_M = h_{GT} - h_R$$

où h_{GT} et h_R sont respectivement la carte de profondeur de vérité terrain et la carte de profondeur reconstruite ; dans ce calcul, seule l'intersection des domaines Ω_{GT} et Ω_R est utilisée.

Étape 2: Nous calculons la valeur médiane M_{val} de Err_M et étant donné un seuil fixe $\theta = 7$, nous retenons l'écart-type σ_M de Err_M pour les valeurs qui se trouvent dans $[M_{val} - \theta, M_{val} + \theta]$.

Étape 3: Nous déterminons la variance V_i de Err_M dans la plage $[M_{val} - 3\sigma_M, M_{val} + 3\sigma_M]$.

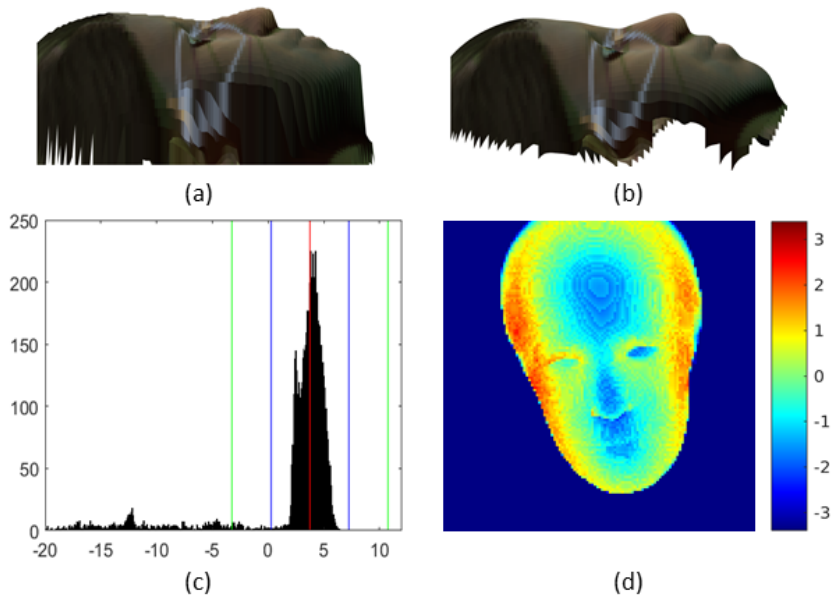

FIGURE 4.7 – Exemple de résultat de reconstruction sur des données synthétiques d'un jeu de données de test
(a) Surface de vérité terrain.
(b) Surface reconstruite.
(c) Histogramme des résidus. Les lignes rouges, vertes et bleues indiquent respectivement M_{val}, $M_{val} \pm \theta$ et $M_{val} \pm 3\sigma_M$.
(d) Carte de chaleur avec des erreurs de pixels après élimination du biais. L'erreur de profondeur est mesurée en unités équivalentes aux pixels. La résolution d'image est de 128×128.

Nous effectuons ces étapes pour tous les exemples de la base de données de test et nous calculons $\sigma = \sqrt{\sum_{i=1}^{N} Var_i/N}$ pour différentes valeurs λ allant de 0 à 0.3. L'influence du choix de λ sur la reconstruction 3D est illustrée sur la figure 4.6. Le choix $\lambda = 0$ correspond à la solution des moindres carrés sans pondération. Nous avons constaté que la valeur optimale est $\lambda = 0.1$, elle correspond à la valeur minimale de l'écart type σ. Cette valeur est utilisée dans toutes les expériences ci-après. Dans la figure 4.7, nous illustrons un exemple de la procédure de comparaison décrite ci-dessus.

4.7.4 Évaluation qualitative

Pour l'analyse qualitative, nous montrons dans la figure 4.8 nos résultats produits à partir des images de certaines célébrités. On peut voir que notre méthode produit des résultats de haute qualité qui correspondent bien à la structure globale de la tête présente dans l'image d'entrée. Comme nous utilisons un modèle de tête 3DMM complet qui comprend également la partie crânienne, notre méthode nous permet de récupérer le modèle 3D de la tête pour tout pixel visible du visage et elle prédit également toute zone cachée par les cheveux (troisième et quatrième lignes dans la figure 4.8). Nous indiquons également la qualité inférieure de la reconstruction pour les surfaces contenant le cou en raison de la discontinuité importante entre les parties du visage et du cou (première et deuxième lignes sur la figure 4.8).

4.7.5 Évaluation quantitative

Les résultats quantitatifs sont rapportés dans le tableau 4.2. Pour cette évaluation, nous utilisons l'ensemble de données BU-3DFE [10] présenté dans la section 4.7.1.2. En utilisant uniquement des sujets d'expressions neutres pour notre processus de comparaison, nous recadrons chaque modèle reconstruit sur la partie représentant les pixels valides à prendre en compte dans la comparaison. Après un processus d'alignement et d'enregistrement basé sur le point itératif le plus proche (ICP) avec le modèle de vérité terrain, nous calculons l'erreur de profondeur absolue. Notez que nous éliminons les exemples lorsque le processus d'alignement échoue (ce qui représente environ 3-4%). Enfin, nous présentons les erreurs de profondeur évaluées par la moyenne (μ), l'écart-type (σ), la médiane ($\widetilde{m}$) et la moyenne de 90% des erreurs les plus significatives ($\delta_{90\%}$). Nous notons que nous rapportons les résultats obtenus sur le même ensemble de données directement à partir de l'article [66].

À partir du tableau 4.2, nous pouvons voir que notre méthode produit des résultats compa-

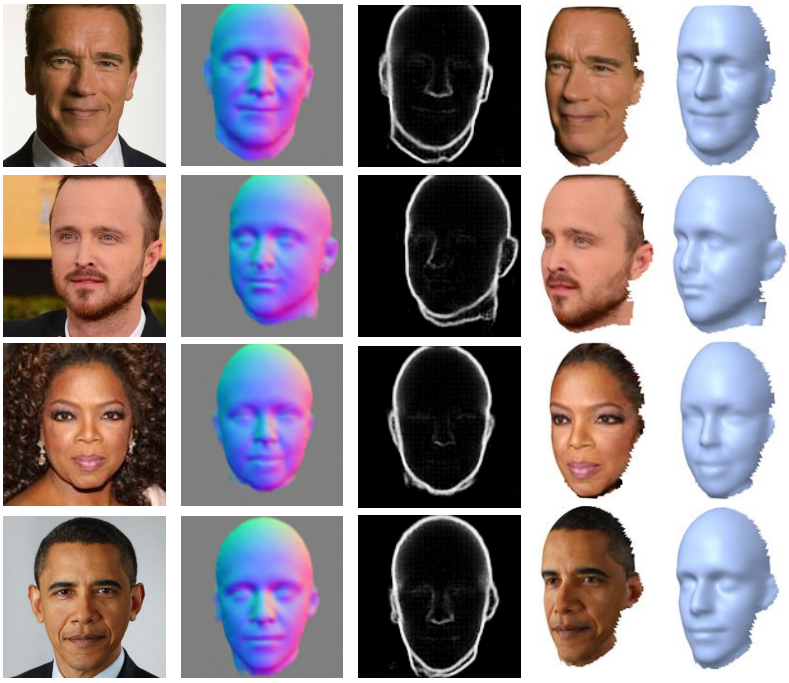

FIGURE 4.8 – Résultats de reconstruction de la surface à partir d'images faciales de certaines célébrités. Les colonnes contiennent dans l'ordre ; l'image d'entrée I, carte $\mathcal{N}$ estimée, carte $\mathcal{W}$ estimée et les deux dernières colonnes contiennent la reconstruction de la forme 3D.

FIGURE 4.9 – Résultats de reconstruction pour trois exemples de BU-3DFE à partir de deux points de vue différents. De gauche à droite : image d'entrée I, carte normale ($\mathcal{N}$), modèle de vérité terrain (vue de face), modèle reconstruit (vue de face), modèle de vérité terrain (vue latérale) et modèle reconstruit (vue latérale).

Méthode	μ	σ	$\widetilde{m}$	$\delta_{90\%}$	RMSE
Kemelmacher-Shlizerman and Basri [28]	3.89	4.14	2.94	7.34	N/A
Zhu *et al.* [16]	3.85	3.23	2.72	6.82	N/A
Richardson *et al.* [42]	3.61	2.99	2.72	6.82	N/A
Sela *et al.* [44]	3.51	2.69	2.65	6.59	N/A
Feng *et al.* [66]	**2.78**	2.04	**1.73**	**5.30**	N/A
Ours	3.04	**1.78**	2.62	5.48	2.05 ± 0.43

TABLE 4.2 – Comparaison quantitative sur l'ensemble de données BU-3DFE [10]. Les faibles valeurs indiquent les meilleures performances.

rables à l'état de l'art. Les performances du travail de [66] sont légèrement meilleures que les nôtres, et nous pensons que cela est dû au fait qu'ils utilisent une partie de l'ensemble de données BU-3DFE [10] pendant leur entraînement. Cet ensemble de données est acquis à l'aide d'un capteur spécial, tandis que notre modèle est entraîné uniquement avec des données synthétiques.

Nous proposons également une autre manière d'estimer l'erreur entre la reconstruction et la vérité terrain. Le critère est l'erreur quadratique moyenne (RMSE) point à plan entre les modèles de reconstruction et de vérité terrain. Les erreurs point à plan sont projetées sur les normales du modèle de la vérité terrain comme il est illustré dans la figure 4.10. Ce faisant, nous évaluons l'erreur entre les deux surfaces dans la direction normale au lieu des distances entre les points. Puisque les modèles n'ont pas les mêmes structures du maillage, nous trouvons le plus proche voisin pour chaque sommet du modèle reconstruit. Cette erreur est calculée comme ceci : une fois les modèles alignés, pour chaque sommet $\boldsymbol{p}_i$ du modèle de référence, nous cherchons son plus proche voisin $\boldsymbol{q}_i$ à partir du modèle reconstruit, et nous retenons ainsi le vecteur normal $\boldsymbol{n}_i$ du modèle de référence. Juste après, nous calculons l'erreur projetée à la normale et nous retenons l'RMSE :

$$\varepsilon = \sqrt{\frac{\sum_{i=1}^{N} \left((\boldsymbol{p}_i - \boldsymbol{q}_i) \cdot \boldsymbol{n}_i \right)^2}{M}} \tag{4.14}$$

avec M est le nombre de sommets du modèle de référence. Une fois que ε est calculé pour chaque sujet, nous pouvons trouver la moyenne et l'écart-type pour l'ensemble des exemples de test. Les valeurs de ce calcul sont reportées dans la sixième colonne du tableau 4.2.

4.8 Conclusion

Dans ce chapitre, nous avons présenté une première approche hybride de reconstruction de visage 3D composée à la fois d'une méthode d'apprentissage et d'une méthode géométrique.

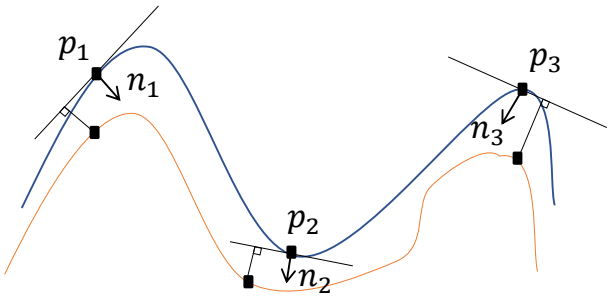

FIGURE 4.10 – Distance Point-a-Plan : n_i et p_i sont respectivement les normales et les sommets du modèle de vérité terrain.

La première étape de notre bloc principal est un réseau de transformation d'image en image qui, à partir d'une image d'entrée faciale I, produit une carte de champ des normales ($\mathcal{N}$) et une carte qui contient le module du gradient de profondeur ($\mathcal{W}$). La deuxième étape est l'intégration du champ des normales basée sur les moindres carrés pondérés, qui utilise les deux sorties du réseau pour générer la profondeur du visage. Notre modèle neuronal a été formé sur un ensemble de données faciales entièrement synthétiques. Et c'est lors de cette occasion que nous intégrons pour la première fois, nos données issues du générateur présenté dans le chapitre précédent.

À fin d'évaluer notre pipeline en entier, nous avons réalisé trois expériences : premièrement, nous montrons que le réseau neuronal génère des cartes du champ des normales précises. Ensuite, d'autres expériences confirment l'efficacité de l'utilisation de la carte $\mathcal{W}$ comme poids lors de l'étape de reconstruction pour résoudre les artefacts dus aux discontinuités. Dans l'expérience finale, nous avons démontré que le cadre proposé atteint les performances de pointe en matière de reconstruction de visage 3D en comparant nos modèles produits à ceux de la base de données BU-3DFE [10]. Nous avons proposé également une nouvelle méthode de calcul d'erreur, qui, à notre avis, est plus représentative pour ce type d'évaluation. L'utilisation de ce critère avec une équation explicitement écrite évite toute ambiguïté sur la manière exacte dont l'évaluation est effectuée et sur la signification de la valeur obtenue.

Notre méthode ne nécessite pas d'alignement précis grâce à l'architecture de transformation d'image en image de notre réseau qui assure l'harmonisation des images produites avec l'image d'entrée à l'échelle du pixel. Elle nous permet également de récupérer la géométrie 3D avec succès pour des images réelles, malgré le fait que le réseau de neurones n'a été formé que sur des données synthétiques.

Chapitre 5

Reconstruction 3D de la tête : Ajustement d'un modèle déformable en utilisant le champ vectoriel des normales

5.1 Introduction

Dans ce chapitre, nous présentons une deuxième méthode de reconstruction de tête en 3D à partir d'une ou de plusieurs images. Le cadre général de notre méthode ressemble à celle présentée dans le chapitre précédent, c'est-à-dire qu'elle repose également sur une approche hybride basée sur l'apprentissage en profondeur et sur une technique géométrique. Pour ce faire, nous proposons un réseau encodeur-décodeur avec des connexions de sauts inspiré de l'architecture U-Net [47] qui assure le transfert des informations à détails fins et permet donc de faire passer un flux d'informations maximal entre les couches du réseau. Nous formons encore une fois notre réseau avec des données synthétiques pour prédire à la fois une carte de champ des normales et une carte des points de repère à partir d'une seule photo d'entrée. Les points de repère sont utilisés pour le calcul de la pose et l'initialisation du problème d'optimisation, qui à son tour, reconstruit la géométrie de la tête 3D en utilisant un modèle paramétrique déformable et la carte de champ des normales prédite. Ensuite, nous réalisons des tests d'évaluation qualitatifs et quantitatifs pour deux configurations (à vue unique et a vues multiples), ce qui nous permet de montrer nos résultats qui sont compétitifs par rapport aux méthodes proposées dans l'état de l'art. Nous montrons aussi, pour une deuxième fois, que bien que notre modèle de réseau de neurones n'ait été formé que sur des données synthétiques, il récupère avec succès des géométries 3D et des poses précises pour des images du monde réel.

Nos principales contributions dans ce chapitre sont :

— Notre réseau (encodeur-décodeur) proposé prédit à la fois le champ vectoriel des normales et les points de repère facial dans deux cartes différentes à partir d'une seule photo d'entrée.

— Nous présentons une stratégie d'ajustement de maillage basée sur le champ des normales de surface ce qui permet la reconstruction de la tête directement à partir d'une ou plusieurs images et vise à récupérer une tête humaine complète ainsi que sa pose.

— Nous montrons que notre approche permet d'atteindre des performances équivalentes à quelques méthodes de l'état de l'art sur l'ensemble de donnéesBU-3DFE [10]. Nous montrons que le modèle proposé se généralise bien aux images du monde réel, même s'il n'a été formé que sur des données synthétiques.

Ce chapitre est organisé comme ceci : tout d'abord, nous présentons une vue d'ensemble de notre méthode dans la section 5.2. Dans la section 5.3, nous décrivons notre jeu de données d'entraînement conçu à partir de notre générateur de tête synthétique présenté dans le chapitre 3. Dans la section 5.4, nous présentons notre architecture de réseau qui régresse deux différentes cartes complètement alignées, au pixel près, avec l'image d'entrée I. Les détails de notre approche d'ajustement sont expliqués dans la section 5.5. Finalement, nous évaluons notre méthode dans la section 5.6 avant de terminer par une conclusion dans la section 5.7.

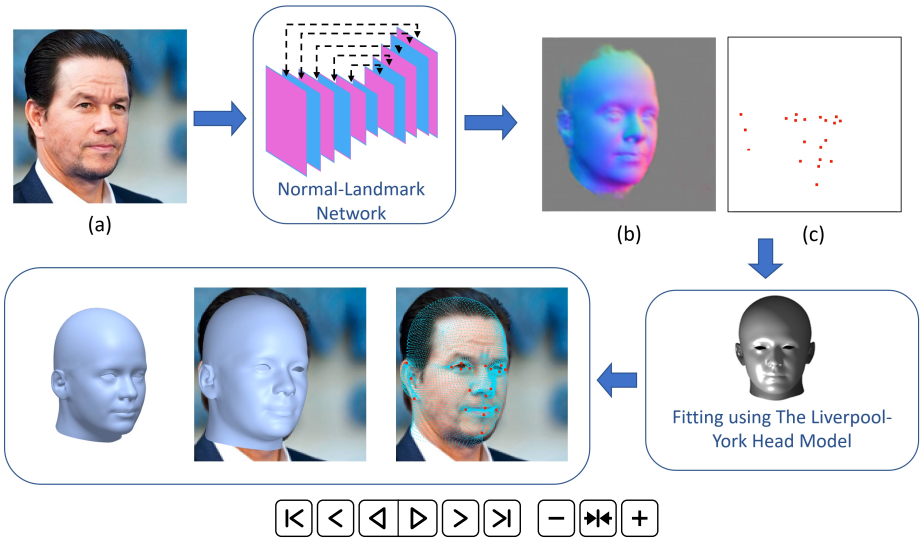

FIGURE 5.1 – Vue d'ensemble de notre méthode de reconstruction faciale 3D proposée. Étant donné une image faciale d'entrée I (a), nous estimons deux cartes différentes (carte de champ des normales $\mathcal{N}$ (b), carte des points de repère $\mathcal{Z}$ (c)) utilisées pour reconstruire la forme du visage 3D via un processus d'ajustement avec le modèle déformable LYHM [5].

5.2 Vue d'ensemble de notre approche

Notre méthode proposée est une approche hybride composée de méthodes d'apprentissage en profondeur et d'optimisation géométrique, et qui est capable de reconstruire un modèle de tête 3D à partir d'une ou de plusieurs images faciales. Le concept clé sur lequel repose notre méthode est le fait que les normales de surface ont un degré élevé d'invariance (échelle, translation), elles ne dépendent pas de la texture, des ombres et de l'éclairage ; et ils véhiculent des informations riches sur la géométrie. Tout d'abord, nous utilisons un réseau encodeur-décodeur qui produit à partir d'une image d'entrée faciale I, une carte de points de repère ($\mathcal{Z}$) et une carte de champ des normales ($\mathcal{N}$). Ensuite, en utilisant ces cartes dans un algorithme de régression paramétrique, nous reconstruisons le modèle facial 3D. Un aperçu de notre cadre proposé est présenté dans la figure 5.1.

5.3 Données d'entraînement : *Normal-Landmark-Net*

Comme nous l'avons fait pour notre méthode présentée dans le chapitre précédent, nous proposons un ensemble de données entièrement synthétiques pour l'entraînement de notre réseau de neurones profonds. Pour y parvenir, nous générons deux différentes cartes en plus des images faciales (première ligne de la figure 5.2) à l'aide de notre générateur de données décrit dans le chapitre 3. La principale source d'information que nous exploitons pour l'ajustement géométrique est la carte de champ des normales $\mathcal{N}$ (deuxième ligne de la figure 5.2). Mentionnons qu'il existe une différence entre les deux cartes de normales de vérité terrain utilisées dans nos deux méthodes. Dans la première méthode, le champ vectoriel des normales couvre les parties visibles du visage et aussi les zones cachées par les cheveux, ce qui revient à dire que le réseau tente de deviner les parties de la tête qui se cachent derrière les cheveux. Pour éviter toute fausse prédiction, nous choisissons d'utiliser dans cette méthode des cartes de normales avec un champ vectoriel qui couvre seulement les parties visibles du visage. Pour assurer l'alignement pendant le processus d'ajustement, nous utilisons aussi la carte de points de repère ($\mathcal{Z}$) (troisième ligne de la figure 5.2) qui contient un ensemble de point répartis sur l'ensemble de la tête. Ces points de repère sont seulement visibles s'ils ne sont pas cachés par les cheveux ou en raison d'une large pose de tête.

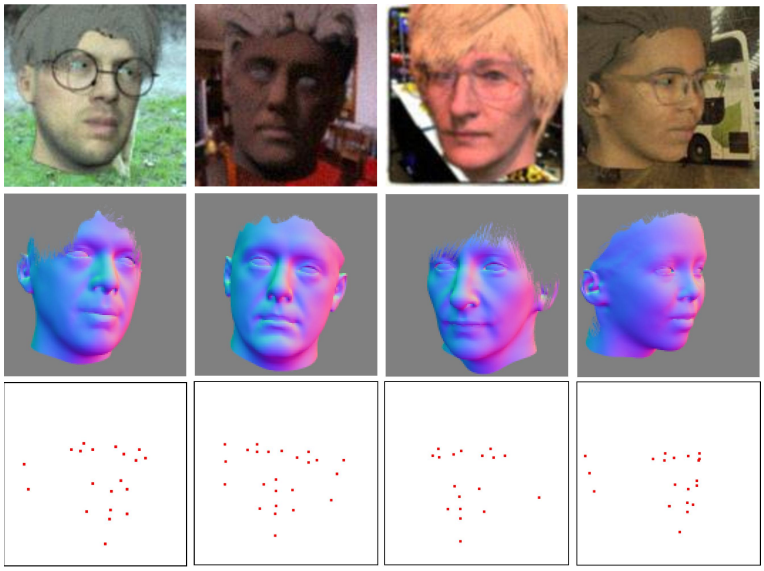

FIGURE 5.2 – Exemples de données d'entraînement. De haut en bas : images faciales synthétiques I, cartes du champ des normales $\mathcal{N}$ et cartes des points de repère $\mathcal{Z}$.

5.4 Architecture du réseau : *Normal-Landmark-Net*

Contrairement au modèle génératif que nous avons utilisé dans le chapitre précédent. Nous proposons cette fois pour l'architecture de notre réseau, un encodeur-décodeur basé également sur le réseau de Su *et al.* [46]. Ce modèle utilise la connexion symétrique entre l'encodeur et le décodeur pour réduire la perte d'information entre les couches successives, implémentant ainsi le modèle U-Net [47]. Afin d'adapter le réseau à notre problématique, un certain nombre de modifications ont été apportées. Le discriminateur a été écarté et nous nous sommes contentés du générateur présent sous forme d'encodeur-décodeur. Pour les données d'entraînement, nous donnons comme entrée l'image du visage (RVB) I, alors que nous n'avons que deux cartes générées $\mathcal{N}$ (3 canaux) et $\mathcal{Z}$ (24 canaux). Pour la fonction coût, nous utilisons une fonction composée pour mesurer la différence entre les cartes générées et les cartes d'entrée réelles, comme décrit ici :

$$\begin{aligned} L &= \lambda_N \, L_{\mathcal{N}} + \lambda_Z \, L_{\mathcal{Z}} \\ L_{\mathcal{N}} = ||\mathcal{N}_{\mathrm{GT}} - \mathcal{N}||_2^2 \quad &, \quad L_{\mathcal{Z}} = ||\mathcal{Z}_{\mathrm{GT}} - \mathcal{Z}||_2^2 \end{aligned} \tag{5.1}$$

En utilisant cette fonction de coût, nous imposons aux images rendues $\mathcal{N}$ et $\mathcal{Z}$ d'être respectivement similaires aux images d'entrées $\mathcal{N}_{\mathrm{GT}}$ et $\mathcal{Z}_{\mathrm{GT}}$. Nous utilisons également avec la fonction coût, les deux termes de pondération λ_N et λ_Z que nous avons varié au fur et à mesure le long du processus d'apprentissage pour assurer la meilleure prédiction des deux cartes $\mathcal{N}$ et $\mathcal{Z}$.

L'architecture de notre réseau *Normal-Landmark-Net* est illustrée dans la figure 5.3.

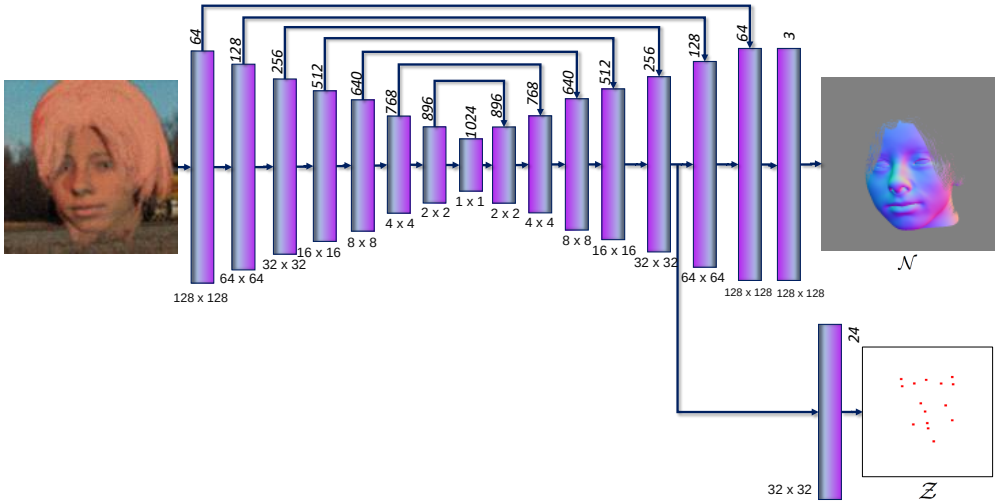

FIGURE 5.3 – Détails de notre réseau : Normal-Landmark-Net. Il s'agit d'un encodeur-décodeur qui produit deux cartes différentes ($\mathcal{N}$ et $\mathcal{Z}$) (montrées à droite) à partir d'une image d'entrée faciale (I) (montrée à gauche). La taille spatiale et le nombre de couches sont indiqués respectivement en dessous et au-dessus de chaque bloc.

5.5 Reconstruction 3D : Ajustement d'un modèle déformable

L'idée principale de notre méthode décrite dans ce chapitre est l'ajustement basé sur la géométrie d'un modèle déformable à un champ de normales prédit, en utilisant un modèle de projection de caméra. Nous décrivons dans cette section la formulation de ce problème.

5.5.1 Processus d'ajustement

Dans notre méthode de reconstruction, nous utilisons le modèle déformable LYHM [5] utilisé dans le processus de génération des données synthétiques dans le chapitre 3. L'objectif derrière cet ajustement est de retrouver le vecteur des paramètres d'identité $\boldsymbol{y} \in \mathbb{R}^{N_y}$ (voir l'équation (2.2)). Ceci nous permet ainsi de retrouver la géométrie 3D d'un visage qui sera la plus adéquate possible à celle projetée dans l'image d'entrée I.

5.5.1.1 Inclusion du champ des normales

Afin d'ajuster le modèle déformable à la carte de champ des normales $\mathcal{N}$, le vecteur normal $\boldsymbol{n}_i$ est calculé pour chaque emplacement de sommet $\boldsymbol{p}_i \in \mathbb{R}^3$, étant donné l'ensemble des sommets adjacents $\{\boldsymbol{q}_{i,1}, \boldsymbol{q}_{i,2}, ..., \boldsymbol{q}_{i,M}\} \subset \mathbb{R}^3$. Tout d'abord, nous calculons une version $\tilde{\boldsymbol{n}}$ non normalisée

pour tout les sommets allant de 1 à N_X comme ceci :

$$\tilde{\boldsymbol{n}}_i = \sum_{j=1}^{N_X} \frac{(\boldsymbol{q}_{i,j} - \boldsymbol{p}_i) \times (\boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i)}{\|(\boldsymbol{q}_{i,j} - \boldsymbol{p}_i) \times (\boldsymbol{q}_{i,j+1} - \boldsymbol{p}_i)\|} \tag{5.2}$$

Après, la normale finale $\boldsymbol{n}_i$ est obtenue par une simple normalisation vectorielle :

$$\boldsymbol{n}_i = \frac{\tilde{\boldsymbol{n}}_i}{||\tilde{\boldsymbol{n}}_i||} \tag{5.3}$$

Cette façon de calcul des normales n'est pas la plus exacte, mais elle est rapide, ce qui est important, car elle est calculée à chaque itération d'optimisation.

5.5.1.2 Modèle de projection

Afin d'obtenir la normale de référence pour chaque sommet $\boldsymbol{p}$, ce dernier est projeté sur le plan image de $\mathcal{N}$ à l'aide du modèle de caméra Pinhole. L'interpolation bi-cubique est utilisée pour obtenir la normale de référence correspondante à partir de $\mathcal{N}$ (qui est une grille discrète). Le processus de projection peut être exprimé comme suit :

$$\boldsymbol{a} = \frac{1}{z} KR(\boldsymbol{p} + \boldsymbol{t}) \tag{5.4}$$

avec :

- $\boldsymbol{a} \in \mathbb{R}^2$ est le point projeté.
- $R \in SO(3)$ est la matrice de rotation, paramétrée par les trois angles de rotation *yaw*, *pitch* et *roll*, et dénotée $\boldsymbol{r}$.
- $\boldsymbol{t} \in \mathbb{R}^3$ est un vecteur de translation.
- z au dénominateur représente la normalisation d'un point 3D pour l'amener au plan normal (z est pris après avoir multiplié le vecteur par R mais avant de le multiplier par K).
- La matrice de projection K contient trois paramètres intrinsèques de la caméra f, u_0, v_0. Nous considérons que ces paramètres sont suffisants étant donné que le calibrage de la caméra est approximatif. Ceci donc définit le modèle de projection suivant :

$$K = \begin{pmatrix} f & 0 & u_0 \\ 0 & f & v_0 \end{pmatrix} \tag{5.5}$$

5.5.1.3 Régression des paramètres

Notre mise en œuvre suit des pratiques standard comme celles de [40,42,44,140] où le modèle déformable 3DMM a été utilisé dans un processus d'ajustement. L'idée fondamentale est de former une image aussi proche que possible d'une image cible en trouvant la combinaison de paramètres la plus adaptée. Dans notre cas, les images cibles sont ($\mathcal{N}$ et $\mathcal{Z}$) produites à partir de notre réseau encodeur-décodeur décrit dans la section 5.4. Le problème peut être exprimé comme la minimisation d'une fonction d'énergie qui représente l'erreur entre les cartes produites et celles générées par le modèle déformable. Cette fonction énergie contient trois principales composants $E_{\mathcal{N}}$, $E_{\mathcal{Z}}$ et E_P :

$$E = \lambda_{\mathcal{N}} E_{\mathcal{N}} + \lambda_{\mathcal{Z}} E_{\mathcal{Z}} + \lambda_P E_P \tag{5.6}$$

Dans ce qui suit, nous allons décrire chaque terme de cette dernière équation.

Tout d'abord, $E_{\mathcal{N}}$ représente la différence entre les normales interpolées $\mathcal{N}(\boldsymbol{a})$ de la projection $2D$ du modèle déformable sur $\mathcal{N}$ et les normales de sommet $\boldsymbol{n}$ qui sont calculées à partir du modèle déformable en utilisant les équations (5.2) et (5.3). Ainsi $E_{\mathcal{N}}$ est défini par :

$$E_{\mathcal{N}} = \frac{1}{2} \sum_{i=1}^{N_X} ||\mathcal{N}(\boldsymbol{a}_i) - \boldsymbol{n}_i||^2 \tag{5.7}$$

Puis, $E_{\mathcal{Z}}$ est définie comme la distance entre les points de repère détectés $\boldsymbol{z}_j$ à partir de $\mathcal{Z}$ et les projections $\boldsymbol{b}_j$ du sous-ensemble correspondant des sommets 3D à partir du modèle déformable LYHM. L'ensemble des projections $\boldsymbol{b}_j$ est a été aussi utilisé dans la section 3.4 pour définir les points de repère qui constitue la carte $\mathcal{Z}$ de vértié terrain. Ceci est réalisé en utilisant le même modèle de projection décrit dans l'équation (5.4).

$$E_{\mathcal{Z}} = \frac{1}{2} \sum_{j=1}^{N_z} ||\boldsymbol{z}_j - \boldsymbol{b}_j||^2 \tag{5.8}$$

Enfin, E_P assure la vraisemblance des têtes reconstruites en supposant des a priori donnés par le modèle statistique de tête LYHM représenté par des valeurs singulières σ_k. Dans la plupart des méthodes de reconstruction 3DMM [1, 44, 50, 61, 63, 140], ce terme est utilisé pour éviter la

dégénérescence de la géométrie reconstruite :

$$E_P = \frac{1}{2}\boldsymbol{y}^T \mathcal{C}_y^{-1} \boldsymbol{y} \tag{5.9}$$

Avec $\mathcal{C}_y$ est la matrice de covariance, qui dans ce cas est une matrice diagonale contenant $\sigma_1^2, \sigma_2^2, ..\sigma_{N_y}^2$.

Le but de cette régression est de prédire les paramètres du modèle déformable $\boldsymbol{y}$ et ceux liés à la projection du modèle de l'espace 3D vers l'espace 2D. Cela peut être formulé comme ce problème de minimisation :

$$\boldsymbol{y}^*, \boldsymbol{t}^*, \boldsymbol{r}^*, K^* = \underset{\boldsymbol{y},\boldsymbol{r},\boldsymbol{t},K}{\operatorname{argmin}} E \tag{5.10}$$

où $\boldsymbol{y}$ est le vecteur de paramètre de forme de tête (voir l'équation (2.2)), $\boldsymbol{t}$ et $\boldsymbol{r}$ sont les paramètres de translation et de rotation définissant la pose de la tête par rapport à la caméra, K est la matrice des paramètres intrinsèque de la caméra. Il s'agit d'un problème des moindres carrés non-linéaire qui peut être résolu efficacement en utilisant l'algorithme de Levenberg-Marquardt. Dans notre résolution de ce problème, nous utilisons le solveur Ceres [138] comme système d'optimisation.

5.5.1.4 Pré-alignement de la tête

Si nous n'utilisons que le champ des normales pour ajuster le modèle de tête, nous pouvons rencontrer un certain nombre de problèmes de convergence. Tout d'abord, l'optimiseur peut mettre un certain temps à converger vers la position et l'orientation souhaitées. Deuxièmement, l'optimisation peut rester bloquée dans un minimum local. L'idéal est que tous les optimiseurs de recherche locale aient une bonne estimation initiale. Nous pouvons y parvenir en calculant une pose initiale plausible. Cela se fait en optimisant $E_{\mathcal{Z}}$ sur $\boldsymbol{t}_0, \boldsymbol{r}_0$, tout en fixant $\boldsymbol{y}$ à 0 et K à une certaine valeur a priori. En d'autres termes, nous utilisons dans cette étape le modèle de tête moyenne et les points de repère donnés par le réseau. Puisqu'il s'agit d'un problème d'optimisation purement géométrique, il converge presque toujours vers la solution optimale et le fait rapidement, compte tenu du faible nombre d'erreurs (jusqu'à 48 dans notre cas) et d'inconnues (6 DoF). Cette approche nous permet de rendre le système robuste par rapport à la pose exacte de la tête et à l'orientation dans l'image.

5.5.2 Reconstruction 3D multi-vues

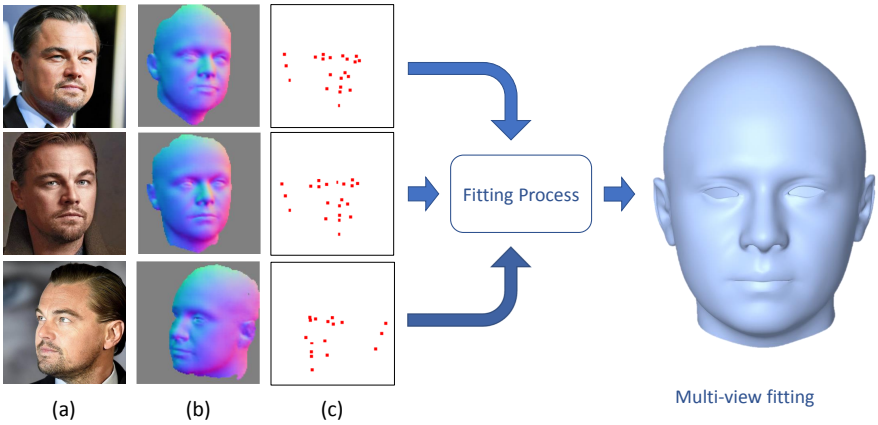

FIGURE 5.4 – Une illustration de notre méthode d'ajustement multi-vues. Nous estimons les deux cartes $\mathcal{N}$ (b) et $\mathcal{Z}$ (c) à partir de chaque image d'entrée I (a), puis nous les utilisons dans le même processus d'ajustement pour obtenir une reconstruction de tête 3D unique.

Notre approche peut être utilisée pour régresser les paramètres 3DMM à partir de plusieurs images faciales I_i de la même personne dans différentes vues. Elle peut être généralisée à n'importe quel nombre d'images d'entrée. Nous produisons les cartes $\mathcal{N}_i$ et $\mathcal{Z}_i$ pour toutes les images d'entrée, puis nous utilisons ces cartes dans le même processus d'ajustement.

Les poses et les paramètres de la caméra sont calculés indépendamment pour chaque image tandis que le vecteur de paramètres de forme $\boldsymbol{y}$ est partagé pour toutes les images. Dans la figure 5.4, nous montrons un exemple d'ajustement 3DMM à partir de plusieurs images de la même personne.

5.6 Évaluation

Dans cette section, nous évaluons la qualité de nos résultats de reconstruction de tête 3D obtenus en utilisant notre méthode proposée. Pour minimiser la fonction d'énergie de l'équation (5.6), nous réglons les paramètres aux valeurs suivantes : $\lambda_{\mathcal{N}} = 1$, $\lambda_{\mathcal{Z}} = 0.8$, et $\lambda_P = 0.4$.

Tout d'abord, nous analysons la qualité des cartes produites par notre modèle *Normal-Landmark-Net* par rapport à celles issues du modèle présenté dans le chapitre précédent dans la section 5.6.1, puis nous montrons les résultats de reconstruction sur des images de quelques célébrités dans la section 5.6.2. Par la suite, nous évaluons quantitativement notre système sur la base de données BU-3DFE [10] que nous avons présenté au chapitre 4 face à certaines méthodes de l'état de l'art.

5.6.1 Évaluation de l'entraînement

Pour effectuer l'entraînement de notre modèle, nous générons 60 000 images faciales à la taille 128×128 (réparties équitablement entre hommes et femmes) ainsi que leurs cartes $\mathcal{N}$ et $\mathcal{Z}$ correspondantes. Nous formons le modèle pour environ 2500 époques avec un pas d'apprentissage de $1e-5$, 32 comme taille de lot, et nous utilisons *RMSprop* comme optimiseur. Comme déjà réalisés pour l'entraînement du réseau de la méthode précédente, nous ajoutons un effet de flou aléatoire et du bruit gaussien ainsi qu'une image d'arrière-plan tirée au hasard de l'ensemble de données COCO [139]. L'objectif de ces augmentations est de rendre le réseau plus résistant aux images du monde réel.

Dans le but d'évaluer la performance de notre réseau à prédire les cartes $\mathcal{N}$ et $\mathcal{Z}$ à partir d'une image faciale d'entrée I, nous utilisons une base de données synthétiques qui se compose de 200 images avec les différentes cartes de vérités terrain ($\mathcal{N}_{\text{GT}}$ et $\mathcal{Z}_{\text{GT}}$). Pour générer cette base, nous suivons la même procédure décrite dans la section 4.7.1.1.

Ces expériences d'évaluations ont pour but d'examiner l'effet des changements que nous avons apportés à notre architecture (adaptée de base du modèle proposé par Su *et al.* [46]) et donc l'évolution de la précision des cartes prédites. Nous montrons dans la figure 5.5 quelques exemples d'images faciales I (première colonne) avec les cartes de vérité terrain $\mathcal{N}_{\text{GT}}$ (deuxième colonne) et les cartes produites $\mathcal{N}$ par notre modèle *Normal-Landmark-Net* (troisième colonne). Dans la quatrième colonne, nous marquons les points de repère de vérité terrain avec des étoiles vertes et celles prédites avec des étoiles rouges.

Avec cette nouvelle architecture, nous remarquons que notre encodeur-décodeur proposé réussit à segmenter le visage à partir de l'image d'entrée et produit des cartes de normales avec un degré de netteté plus élevée que celles produites par *Face-Normal-Net* (Voir section 4.7.2). Avec ceci, les points de repère produits sont assez précis et parfois bien estimés dans des endroits qui ne contiennent pas de points de repère de vérité terrain (Voir les exemples de la première et deuxième ligne de la figure 5.5).

Pour l'analyse quantitative, nous reportons dans le tableau 5.1, les résultats de prédiction obtenus par nos deux réseaux sur les données synthétiques de test. Nous remarquons tout d'abord une légère diminution du taux de précision contre un rappel plus élevé. Nous pensons que ceci est dû à la difficulté considérable de la tâche de ce réseau par rapport à celle de *Face-Normal-Net*. En effet, contrairement à ce dernier réseau qui a été entraîné pour prédire le champ de normales

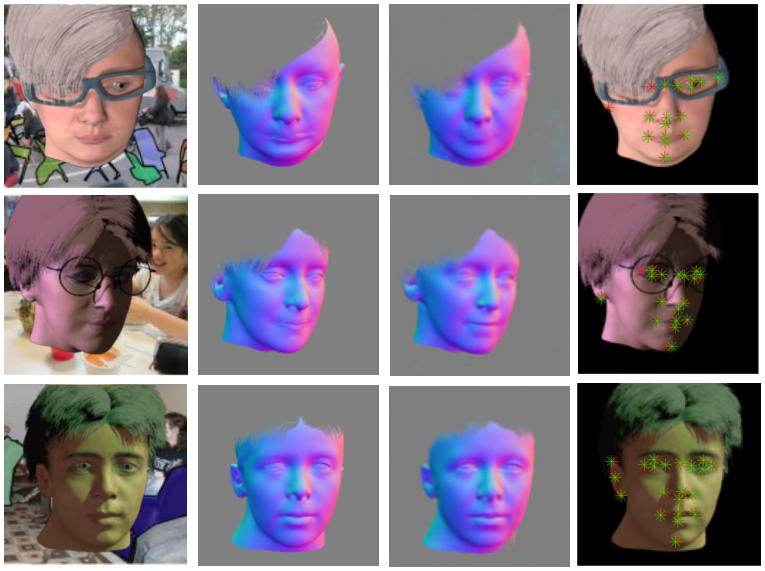

FIGURE 5.5 – Comparaison entre la vérité terrain et les estimations des cartes $\mathcal{N}$ et $\mathcal{Z}$. La première colonne contient l'image d'entrée I, la deuxième et la troisième colonne contiennent respectivement des cartes de normales de vérité terrain $\mathcal{N}_{\text{GT}}$ et estimées $\mathcal{N}$ et la quatrième colonne contient l'image du visage avec les points de repère sous forme d'étoiles vertes pour la vérité terrain $\mathcal{Z}_{\text{GT}}$ et rouges pour la prédiction $\mathcal{Z}$.

du visage et deviner aussi une forme approximative continue (sans trous) de ce qui est caché sous les cheveux, l'objectif de *Normal-Landmark-Net* est de détecter seulement les valeurs des normales aux pixels qui appartiennent au visage. La tâche de segmentation de ce réseau est plus compliquée vu la forme irrégulière des cheveux qui peuvent couvrir le visage partiellement et perturber donc la visibilité de certaines zones. Tout ceci a donc provoqué l'augmentation des pixels faux positifs et par conséquent la diminution de la précision.

TABLE 5.1 – Comparaison quantitative entre les deux réseaux de neurones (*Face-Normal-Net* et *Normal-Landmark-Net*) sur la base de données synthétiques. À gauche : les valeurs élevées de la précision et du rappel pour les résultats de segmentation indiquent les meilleures performances. À droite : les meilleures performances pour la précision des cartes de normales sont indiquées par les faibles valeurs de la moyenne et l'écart-type de l'erreur angulaire d'une part et par les valeurs élevées pour les pourcentages d'erreurs inférieures à différents seuils.

	Mask Evaluation	
CNN	*précision*	*rappel*
Face-Normal-Net	93.72 %	98.40 %
Normal-Landmark-Net	90.06 %	99.43 %

Normals Evaluation				
Mean	*Std*	$< 10°$	$< 20°$	$< 30°$
10.01°	12.45°	67.50 %	92.65 %	97.13 %
8.67°	9.18°	72.96 %	94.56 %	97.86 %

En utilisant l'équation d'erreur angulaire (4.13), nous calculons la moyenne (*Mean*), l'écart-type (*Std*) et le pourcentage de pixels avec une erreur angulaire inférieure à 10°, 20° et 30°. D'après ses différentes mesures, nous remarquons qu'il y a une amélioration significative de la qualité des cartes de normales prédites avec notre réseau *Normal-Landmark-Net*. Nous estimons

que de telles performances sont dues à l'augmentation de la profondeur (nombre des filtres) des couches intermédiaires (5 à 11) de notre encodeur décodeur, en plus des connexions de saut qui peuvent transférer les détails fins (voir figure 5.5).

5.6.2 Évaluation qualitative

Des résultats de reconstruction 3D à partir de quelques photos de certaines célébrités qui ont été capturées dans différentes poses, sont illustrés dans la figure 5.6. Cette évaluation montre comment notre réseau a appris à représenter des structures de tête de personnes réelles. À partir de la deuxième ligne de la figure 5.6, nous remarquons que notre réseau produit des cartes du champ des normales de haute qualité ainsi que des détections de points de repère précises (représentés par les points rouges dans la 3^{e} ligne), bien qu'il ait été formé avec des données entièrement synthétiques. Nous pouvons voir que la tête est bien séparée des cheveux et de l'arrière-plan. Notre méthode produit des résultats de haute qualité, qui correspondent bien à la structure globale. Comme nous utilisons le modèle déformable LYHM [5] qui inclut la partie crânienne, notre méthode nous permet de récupérer le modèle 3D de la tête. La région crânienne se rapproche des zones cachées par les cheveux alors que nous estimons seulement les parties visibles qui incluent la peau du visage.

Une comparaison entre les processus monoculaire et multi-vues utilisant un exemple de l'ensemble de données BU-3DFE [10] est illustrée dans la figure 5.7. Nous pouvons remarquer une ressemblance visuelle significative entre la reconstruction 3D et la vérité terrain en utilisant le processus multi-images—la reconstruction 3D avec plus d'images permet de mieux reconstituer la forme du visage. Par contre, nous remarquons aussi que pour quelques autres exemples de la base BU-3DFE [10], les résultats sont moins précis.

5.6.3 Évaluation quantitative

Pour l'évaluation quantitative, nous démontrons l'efficacité de notre approche en utilisant l'ensemble de données BU-3DFE [10]. Dans notre évaluation, nous utilisons à la fois des images frontales et de profil de tous les sujets pour évaluer les performances de notre algorithme avec une reconstruction mono et multi-vues. Puisque notre 3DMM ne contient pas d'expressions, nous n'utilisons que des visages d'expressions neutres. L'évaluation se produit comme ceci : tout d'abord, nous effectuons un processus de pré-alignement approximatif entre le modèle reconstruit et la vérité terrain en utilisant six sommets présélectionnés. Ensuite, le processus d'alignement

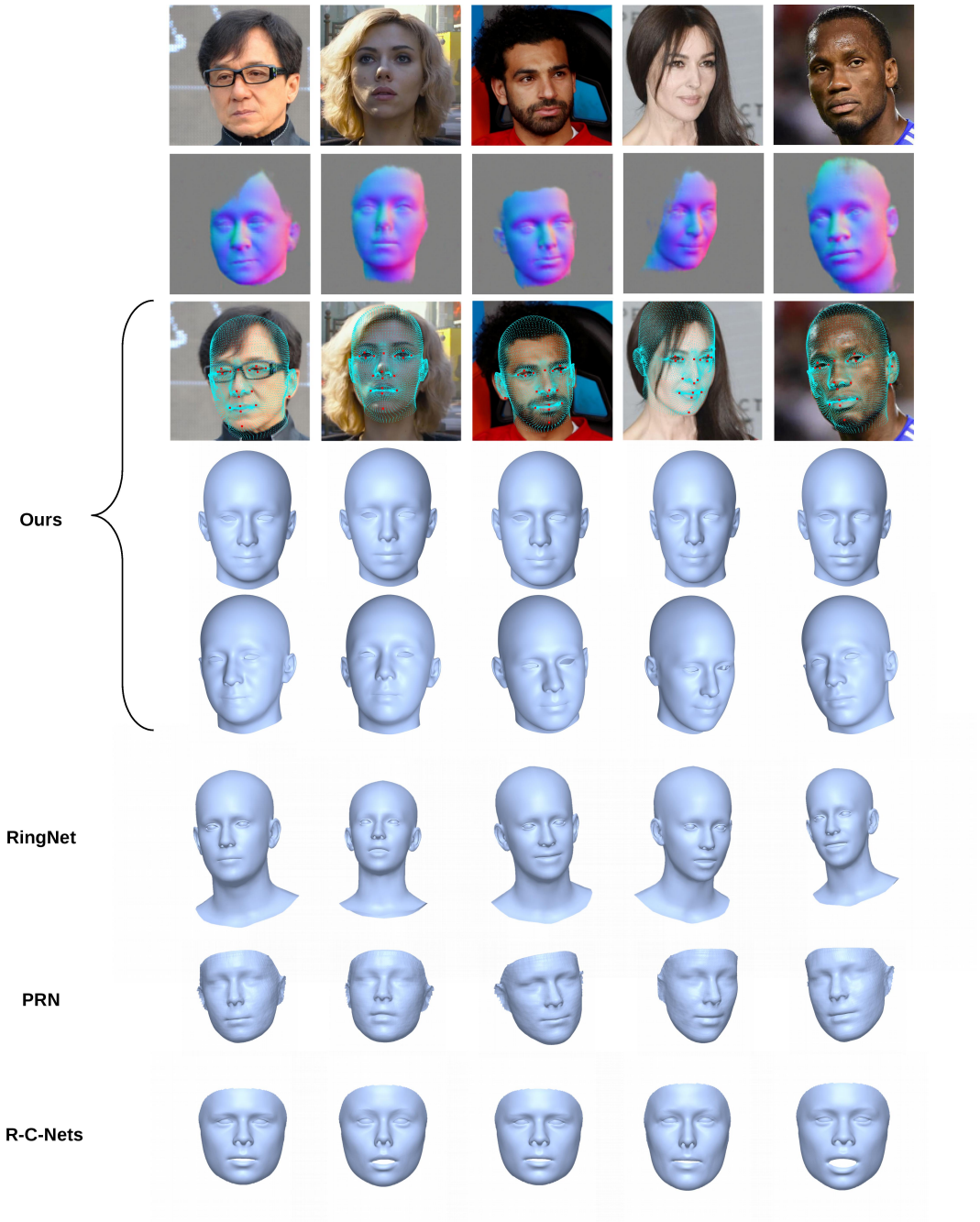

FIGURE 5.6 – Comparaison visuelle avec des méthodes de l'état de l'art en utilisant des images faciales de certaines célébrités. Les lignes contiennent dans l'ordre ; image d'entrée I, carte $\mathcal{N}$ prédite, image d'entrée avec les points de repère prédits (points rouges) et résultats d'alignement dense (sommets projetés du modèle déformable produit par notre processus d'ajustement en bleu), notre méthode (vue frontale), notre méthode (vue aligné), RingNet [7], PRN [8] et RC-Nets [9].

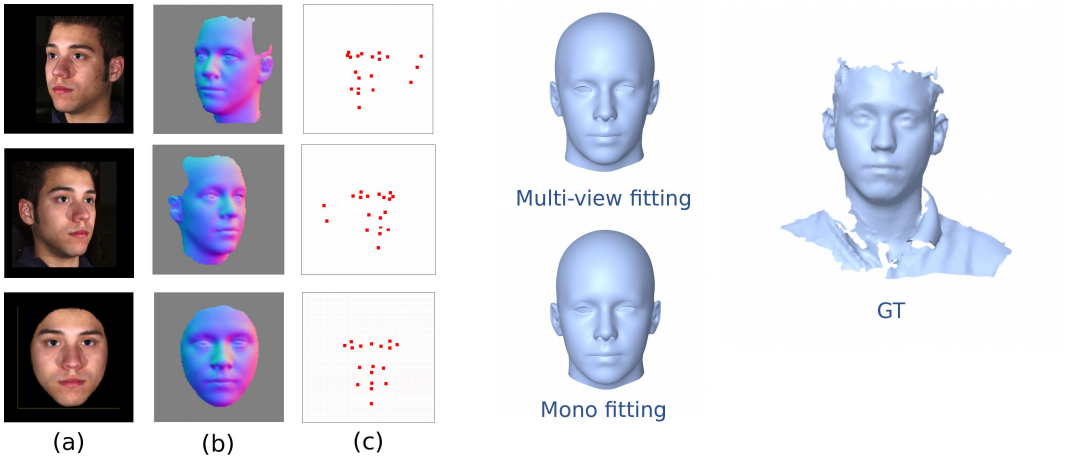

FIGURE 5.7 – Comparaison entre les deux processus d'ajustement (mono et multi-images) en utilisant un exemple de jeu de données BU-3DFE [10]. Images d'entrée I (a), carte de champ des normales $\mathcal{N}$ (b), carte des points de repère $\mathcal{Z}$ (c). (Multi-view fitting) : reconstruction de la tête 3D en utilisant toutes les images dans le même processus d'ajustement. (Mono fitting) : tête 3D reconstruite en utilisant uniquement l'image frontale dans le processus d'ajustement (troisième ligne). (GT) : la vérité terrain du maillage de la tête 3D.

avec le modèle de vérité terrain est effectué à l'aide du solveur itératif du point le plus proche (ICP). Une fois que le solveur converge, nous calculons les distances point à plan et les erreurs de profondeur absolue entre les modèles 3D reconstruits et les maillages 3D de vérité terrain. Nous éliminons les exemples lorsque le processus d'alignement échoue (ce qui représente environ 3-4 %).

Similairement à la première approche, nous utilisons l'erreur quadratique moyenne (RMSE) point à plan comme mesure de performance. Nous pensons que c'est un bon moyen d'évaluer les performances qui n'est sensible ni au nombre de sommets du modèle ni à la manière dont ils sont distribués. Cette erreur est calculée selon l'équation 4.14.

TABLE 5.2 – Comparaison quantitative sur l'ensemble de données BU-3DFE [10]. Les faibles valeurs indiquent les meilleures performances.

Méthode	μ	σ	$\widetilde{m}$	$\delta_{90\%}$	RMSE
Ours (Mono)	2.21	1.08	2.09	3.61	1.74 ± 0.44
Ours (Multi)	2.17	**1.04**	2.05	3.53	1.67 ± 0.43
RingNet [7]	3.42	1.58	3.23	5.60	1.90 ± 0.49
PRN [8]	1.83	1.70	1.43	3.46	1.86 ± 0.47
R-C-Nets [9]	**1.64**	1.69	**1.27**	**3.00**	**1.60 ± 0.41**

Dans le tableau 5.2, nous rapportons les résultats numériques sous la forme de mean ± std pour l'erreur du point à plan (RMSE) et pour les erreurs de profondeur absolue, nous rapportons la moyenne (μ), l'écart-type (σ), la médiane ($\widetilde{m}$) et la moyenne de 90 % des erreurs les plus

significatives ($\delta_{90\%}$). Nous avons comparé notre système en configuration monoculaire (Mono) et multi-vues (Multi) aux différentes approches de l'état de l'art qui ont un code disponible en ligne [7–9]. Nous avons utilisé exactement la même procédure pour tous les systèmes testés afin de nous assurer qu'ils sont tous dans les mêmes conditions. Nous pouvons voir que l'erreur moyenne de l'approche multi-vue est inférieure à celle que nous avons avec la configuration monoculaire. Dans la plupart des cas, la morphologie de la tête entière est mieux capturée en utilisant plusieurs images. Dans certains autres cas, cette approche en vues multiples donne de moins bons résultats. Et comme l'évaluation ne se fait que sur la partie faciale, elle ne contribue pas à la précision. On peut voir que la méthode proposée est performante au niveau de l'état de l'art, étant clairement surpassé uniquement par les R-C-Nets. La précision que nous avons obtenue pour ce dernier est en quelque sorte pire que celle rapportée dans la publication correspondante [9], qui est de 1.40 ± 0.31 pour l'erreur du point à plan (RMSE). Ceci est peut être dû à certaines différences dans la technique d'évaluation. Cependant, contrairement à notre approche hybride qui combine les réseaux de neurones avec une technique de géométrie contrôlable, R-C-Nets [9] utilisent un seul réseau formé de bout-en-bout pour régresser les paramètres 3DMM à partir d'une image d'entrée. À notre connaissance, ce type d'architecture ignore les informations précieuses qui peuvent décomposer le problème en plusieurs modules. D'une manière générale, il est difficile d'améliorer ou de contrôler ce type de système. Si un changement structurel doit être appliqué (par exemple, l'augmentation des dimensions d'entrée en ajoutant plus de caractéristiques), l'ancien modèle n'est plus utilisé et le réseau doit être remplacé et entraîné à nouveau. En contrepartie, pour notre approche, il est possible d'intégrer d'autres informations issues de l'image d'entrée pendant l'ajustement pour affiner la reconstruction.

5.7 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle approche composée à la fois de méthodes d'apprentissage en profondeur et de géométrie visuelle, dont le but est d'estimer la forme complète de la tête humaine en 3D à partir d'une ou de plusieurs images. Notre méthode utilise un réseau encodeur-décodeur qui transforme l'image d'entrée I à une carte de champ des normales $\mathcal{N}$ et une carte de points de repère $\mathcal{Z}$. Ces cartes sont ensuite utilisées dans un processus d'ajustement pour régresser les paramètres 3DMM de l'identité du visage à partir du modèle LYHM. Similairement à ce que nous avons utilisé pour le réseau de notre méthode décrite dans le chapitre précédent, cette

nouvelle architecture d'encodeur décodeur a été entraînée uniquement à partir d'un ensemble de données faciales synthétiques. Et encore une fois, ceci produit de bons résultats en termes de précision et de généralisation sur des images du monde réel.

En plus de la reconstruction 3D de la tête, la carte de points de repère $\mathcal{Z}$ peut être directement utilisée pour le suivi du visage et aussi pour l'estimation de la pose qui est une partie essentielle de notre pipeline de reconstruction. Dans notre pipeline, les points de repère sont utilisés pour trouver une bonne estimation initiale de la pose avant l'ajustement final qui est plus précis. Cela nous permet d'améliorer le taux de convergence du processus de régression et de minimiser les chances d'atteindre un minimum local.

Nous avons réalisé des expériences quantitatives et qualitatives pour évaluer la performance de notre approche. Nous démontrons que notre approche proposée atteint de hautes performances dans la reconstruction de visage 3D pour les configurations à vue unique et en multi-vues.

Chapitre

6 Conclusion et perspectives

Dans ce chapitre, nous procédons au bilan de nos travaux décrit dans ce manuscrit. Tout d'abord, nous présentons nos principales contributions puis nous examinons leurs limites et nous décrivons les futures perspectives en indiquant les idées d'amélioration les plus intéressantes.

6.1 Contributions principales

Dans le cadre de cette thèse CIFRE effectué au sein de la start-up *MCQ-Scan*, nous avons étudié la reconstruction 3D de visage à partir d'une image 2D. Cette technique peut être appliquée dans divers domaines, tels que la sécurité, la santé physique et mentale, le divertissement et dans des applications de réalité augmenté/virtuelle. Cependant, ce problème est sous-déterminé et il nécessite donc des connaissances préalables pour pouvoir le résoudre sans ambiguïté.

Nous avons commencé par un résumé des différentes approches qui sont en relation avec nos méthodes proposées, spécifiquement, les méthodes à base de photométrie, ajustement d'un modèle déformable et les réseaux de neurones profonds. Dans le même contexte, nous avons mis l'accent sur les modèles déformables (3DMM) qui ont fait l'objet de connaissances supplémentaires dans nos processus de reconstruction de visage 3D à partir d'une image 2D et nous avons présenté les plus connues d'entre eux.

Avant de s'attaquer au problème de reconstruction de visage, nous avons abordé le sujet de manque de données nécessaires à l'entraînement des réseaux de neurones. Pour éviter cet écueil, nous avons proposé d'utiliser des images synthétiques de tête humaine de haute qualité qui visent à imiter les images réelles capturées par les différents appareils photo. Contrairement aux méthodes qui utilisent seulement des visages synthétisés à partir du modèle BFM [2], notre principale contribution est de combiner différents composants afin de synthétiser une quantité illimitée d'images faciales avec des cartes riches en informations géométriques. Particulièrement, notre générateur est composé de :

— Modèle déformable LYHM [5] pour synthétiser aléatoirement des géométries de tête crâniofacial.

— Plusieurs textures faciales de haute résolution (également distribuées entre homme et femme).

— Modèle de yeux avec plusieurs textures pour changer la couleur d'iris.

— La base de données de coiffures *USC-HairSalon* [108] que nous avons prétraitées pour fournir des maillages faciles à manipuler, légers et donc moins coûteux en matière de temps de chargement.

— Plusieurs textures de cheveux synthétisés pour assurer la diversité des modèles de coiffures rendues en matières : couleur, longueur et aspect visuel.

— Un moteur de rendu graphique 3D avec différentes techniques d'affichage (*shader*) qui permettent d'assembler les éléments cités ci-dessus pour produire des têtes humaines synthétiques 3D ayant une apparence réaliste avec des géométries connues.

Après ceci, nous avons présenté nos deux approches hybrides permettant de reconstruire un visage en 3D seulement à partir d'une image 2D. Elles sont complètement automatiques et basées à la fois sur une approche d'apprentissage machine et sur une approche géométrique. Dans les deux approches, nous avons utilisé des réseaux de neurones qui ont été uniquement formés sur un ensemble de données faciales synthétiques fournies par notre générateur de données synthétiques. Les deux CNN permettent de prédire deux différentes cartes complètement alignées, au niveau du pixel, en ayant une image à l'entrée.

- **Intégration robuste des normales :** la première méthode génère une carte de champ des normales et une carte du module de gradient de la profondeur du visage à partir du réseau *Face-Normal-Net*. Elles sont par la suite utilisées dans un processus d'intégration robuste des normales basée sur les moindres carrés pondérés qui permet de reconstruire la profondeur faciale. Pour faire face aux discontinuités de profondeur lors de l'étape d'intégration, nous avons utilisé la deuxième carte estimée (module de gradient de la profondeur) qui assure la robustesse de la reconstruction en fournissant un poids important pendant l'apparition des discontinuités. Pendant l'évaluation de notre méthode, nous avons démontré l'efficacité de cette carte pour améliorer la reconstruction et gérer les artefacts.
- **Ajustement d'un modèle déformable :** Cette approche est considérée la plus importante dans cette thèse. Similairement à l'architecture du réseau de la première méthode,

Normal-Landmark-Net permet d'obtenir pour chaque image d'entrée, une carte de champ des normales et une carte contenant des points de repère répartis sur différentes parties du visage. Avec le modèle paramétrique LYHM, ces deux cartes sont utilisées dans un processus d'ajustement pour régresser les paramètres d'identité du visage à partir du modèle LYHM. Les points de repère ont permis de trouver une bonne estimation initiale de la pose avant l'ajustement final. Cela améliore le taux de convergence du processus de régression et minimise les chances d'atteindre un minimum local.

Nous avons réalisé plusieurs expériences pour évaluer les performances de nos deux approches. Pour ce faire, nous avons utilisé quelques images de célébrité pour évaluer nos méthodes qualitativement, puis, en utilisant la base de données BU-3DFE [10], nous avons pu mesurer l'efficacité de la reconstruction de nos méthodes quantitativement. Ces expériences d'évaluation ont montré que nos deux réseaux de neurones génèrent des cartes de normales de haute qualité et se généralisent bien sur des images du monde réel. Nous avons aussi prouvé que nos deux approches sont capables de produire des reconstructions plausibles et atteignent des performances de pointe face à d'autres méthodes de l'état de l'art.

Nos deux méthodes de reconstruction (intégration des normales et ajustement d'un 3DMM) sont génériques et ils peuvent donc éventuellement être étendus pour reconstruire d'autres types d'objets 3D si le bon générateur de données est disponible. Pour la deuxième approche, il nous faut aussi le bon modèle déformable 3DMM en relation avec la catégorie d'objets à reconstruire. En comparant nos deux méthodes entre eux, il est bien clair que la méthode à base d'ajustement de 3DMM est meilleure que celle à base d'intégration des normales tant pour la configuration à vue unique que pour celle à multiples vues. Alors que l'intégration des normales produit seulement un nuage de point 3D qui représente la surface faciale, notre deuxième méthode à base d'ajustement de modèle déformable permet d'obtenir un maillage sémantique qui définit une vraie morphologie d'une tête humaine crâniofacial grâce au modèle LYHM. À l'opposé de la première méthode qui prend une seule image en entrée, la méthode à base d'ajustement peut être utilisée pour reconstruire un visage 3D à partir de plusieurs images faciales de la même personne prises dans différentes vues. Elle peut être donc généralisée à n'importe quel nombre d'images d'entrée ce qui améliore la qualité de reconstruction dans certains cas. Grâce à la carte de champ des normales prédite par notre réseau, la procédure d'ajustement comprend un nombre important de points qui sont répartis sur toute la surface faciale visible dans l'image d'entrée et leurs correspondants sur le maillage du 3DMM. Ceci permet de mettre l'accent sur des détails spécifiques

du visage contrairement aux autres méthodes d'ajustement classiques [15, 16, 21, 26, 37–39] qui utilisent les points de repère détectés uniquement. En plus de la reconstruction 3D de la tête, le réseau *Normal-Landmark-Net* produit la carte des points de repère $\mathcal{Z}$ qui peut être directement utilisée pour le suivi du visage et l'estimation de la pose.

Surpassant les performances de la première approche en matière de reconstruction de visage en 3D, le processus complet de reconstruction de notre deuxième approche est moins rapide que le premier en raison de la nature itérative du solveur d'optimisation. Le tableau 6.1 montre le résultat de temps d'exécution pour chaque phase des deux méthodes. Nous utilisons pour cette évaluation un *GPU* NVIDIA GeForce GTX 1650 et un CPU Intel(R) Core(TM) $i5-4210$ @ 1.70 GHz.

TABLE 6.1 – Temps d'exécution (en secondes) de chaque phase pendant le test de nos deux méthodes.

Méthode	Intégration des normales	Ajustement d'un 3DMM	
Nombre d'images I	1	1	3
CNN	*Face-Normal-Net* ($\mathcal{N}$ et $\mathcal{W}$)	*Normal-Landmark-Net* ($\mathcal{N}$ et $\mathcal{Z}$)	
	1.55	0.93	2.79
Reconstruction	0.08	2.89	9.56
Total	1.63	3.82	12.35

6.2 Limitations et perspectives

En dépit des performances robustes dans de nombreux cas, nos méthodes présentent un certain nombre de limites. Dans ce sens, Il existe différentes directions de recherche intéressantes qui peuvent être abordées.

— Les résultats de reconstruction que nous avons obtenus dépendent de la diversité et de la qualité des cartes produites à partir de nos réseaux de neurones qui à leur tour dépendent de la base de données qui peut introduire certains biais pendant l'apprentissage. Dans cette perspective, une amélioration du générateur de données synthétiques peut être réalisée. Cette limitation peut être surmontée en utilisant un meilleur générateur de données synthétiques, il ne s'agit donc pas d'une limitation fondamentale de nos méthodes de reconstruction proposées. Par exemple, notre modèle LYHM utilisé n'inclut pas d'expressions faciales et a une plage d'âge limitée. C'est pourquoi il est difficile de reconstruire les détails les plus fins du visage, car la précision de la géométrie récupérée est limitée à la flexibilité de

modèle. Cette restriction peut être surmontée en adoptant un modèle déformable plus expressif [141] pour le générateur de données synthétiques et pour le processus d'ajustement. D'autres éléments peuvent aussi être intégrés dans le générateur pour créer plus de diversité et rendre l'aspect final plus naturel (moustache, barbe et d'autres types d'accessoires : chapeaux, écharpes, etc.). Une autre limitation est que les données synthétiques générées par combinaison de divers éléments peuvent avoir des caractéristiques irréalistes, ce qui peut introduire certains biais dans le processus d'apprentissage. Pour améliorer la qualité et le réalisme des données synthétiques, nous pouvons envisager l'utilisation d'architecture du type GAN [142, 143] en combinaison avec un rendu 3D classique pour transformer les images synthétiques en images photoréalistes tout en conservant les géométries faciales intactes.

— Alors que nos deux CNN produisent des cartes de normales plausibles, certaines caractéristiques faciales générées sont encore légèrement floues. Les travaux futurs pourraient inclure le test d'autres fonctions coûts et l'amélioration de l'architecture en augmentant la complexité des réseaux tout en modifiant les couches pour obtenir des détails précis similaires à [99, 132].

Bibliographie

[1] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pp. 187–194, ACM Press/Addison-Wesley Publishing Co., 1999. viii, 11, 12, 13, 14, 15, 19, 27, 31, 33, 83

[2] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 296–301, Ieee, 2009. viii, 13, 14, 27, 93

[3] P. Huber, G. Hu, R. Tena, P. Mortazavian, P. Koppen, W. J. Christmas, M. Ratsch, and J. Kittler, "A multiresolution 3d morphable face model and fitting framework," in *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2016. viii, 13, 14, 27

[4] J. Booth, A. Roussos, A. Ponniah, D. Dunaway, and S. Zafeiriou, "Large scale 3d morphable models," *International Journal of Computer Vision*, vol. 126, no. 2-4, pp. 233–254, 2018. viii, 13, 14, 27, 32

[5] H. Dai, N. Pears, W. A. P. Smith, and C. Duncan, "A 3D morphable model of craniofacial shape and texture variation," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. viii, xi, 14, 27, 28, 33, 78, 81, 88, 94

[6] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse : A 3d facial expression database for visual computing," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 3, pp. 413–425, 2013. viii, 13, 14, 27, 32

[7] S. Sanyal, T. Bolkart, H. Feng, and M. J. Black, "Learning to regress 3D face shape and expression from an image without 3D supervision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7763–7772, 2019. xii, 22, 89, 90, 91

[8] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3d face reconstruction and dense alignment with position map regression network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 534–551, 2018. xii, 15, 24, 32, 60, 89, 90, 91

[9] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong, "Accurate 3D face reconstruction with weakly-supervised learning : From single image to image set," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019. xii, 22, 89, 90, 91

[10] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato, "A 3D facial expression database for facial behavior research," in *7th international conference on automatic face and gesture recognition (FGR06)*, pp. 211–216, IEEE, 2006. xii, xiii, 6, 26, 31, 69, 73, 75, 76, 78, 85, 88, 90, 95

[11] **O.Bouafif**, B. Khomutenko, and M. Daoudi, "Monocular 3d head reconstruction via prediction and integration of normal vector field," in *15th International Conference on Computer Vision, Theory and Applications.*, 2020. xvi

[12] **O.Bouafif**, B. Khomutenko, and M. Daoudi, "Hybrid approach for 3d head reconstruction : Using neural networks and visual geometry," in *25th International Conference on Pattern Recognition (ICPR2020)*, 2020. xvi

[13] C. Siegl, V. Lange, M. Stamminger, F. Bauer, and J. Thies, "Faceforge : Markerless non-rigid face multi-projection mapping," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 11, pp. 2440–2446, 2017. 2

[14] H. Drira, B. B. Amor, A. Srivastava, M. Daoudi, and R. Slama, "3d face recognition under expressions, occlusions, and pose variations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2270–2283, 2013. 2

[15] Y. Hu, D. Jiang, S. Yan, L. Zhang, *et al.*, "Automatic 3d reconstruction for face recognition," in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, pp. 843–848, IEEE, 2004. 2, 3, 15, 18, 19, 27, 29, 96

[16] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 787–796, 2015. 2, 3, 15, 18, 19, 27, 31, 75, 96

[17] F. Liu, R. Zhu, D. Zeng, Q. Zhao, and X. Liu, "Disentangling features in 3d face shapes for joint face reconstruction and recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5216–5225, 2018. 2, 32

[18] P. Hammond, "The use of 3d face shape modelling in dysmorphology," *Archives of disease in childhood*, vol. 92, no. 12, p. 1120, 2007. 2

[19] A. Bottino, M. De Simone, A. Laurentini, and C. Sforza, "A new 3-d tool for planning plastic surgery," *IEEE transactions on biomedical engineering*, vol. 59, no. 12, pp. 3439–3449, 2012. 2

[20] K. Anis, H. Zakia, D. Mohamed, and C. Jeffrey, "Detecting depression severity by interpretable representations of motion dynamics," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pp. 739–745, IEEE, 2018. 2

[21] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face : Real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2387–2395, 2016. 2, 3, 15, 18, 19, 27, 96

[22] A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "Avatarme : Realistically renderable 3d facial reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 760–769, 2020. 2

[23] Q. Zhang, Y. Guo, P.-Y. Laffont, T. Martin, and M. Gross, "A virtual try-on system for prescription eyeglasses," *IEEE computer graphics and applications*, vol. 37, no. 4, pp. 84–93, 2017. 3

[24] K. Scherbaum, T. Ritschel, M. Hullin, T. Thormählen, V. Blanz, and H.-P. Seidel, "Computer-suggested facial makeup," in *Computer Graphics Forum*, vol. 30, pp. 485–492, Wiley Online Library, 2011. 3

[25] T. Beeler, B. Bickel, P. Beardsley, B. Sumner, and M. Gross, "High-quality single-shot capture of facial geometry," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 1–9, 2010. 3, 9

[26] A. Bas, W. A. Smith, T. Bolkart, and S. Wuhrer, "Fitting a 3d morphable model to edges : A comparison between hard and soft correspondences," in *Asian Conference on Computer Vision*, pp. 377–391, Springer, 2016. 3, 15, 18, 20, 27, 29, 96

[27] P. Huber, Z.-H. Feng, W. Christmas, J. Kittler, and M. Ratsch, "Fitting 3d morphable face models using local features," in *Image Processing (ICIP), 2015 IEEE International Conference on*, pp. 1195–1199, IEEE, 2015. 3, 15, 18, 27, 29

[28] I. Kemelmacher-Shlizerman and R. Basri, "3d face reconstruction from a single image using a single reference face shape," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 2, pp. 394–405, 2010. 3, 15, 16, 27, 29, 75

[29] R. Dovgard and R. Basri, "Statistical symmetric shape from shading for 3d structure recovery of faces," in *European Conference on Computer Vision*, pp. 99–113, Springer, 2004. 3, 15, 16

[30] Y. Li, L. Ma, H. Fan, and K. Mitchell, "Feature-preserving detailed 3d face reconstruction from a single image," in *Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*, pp. 1–9, 2018. 3, 15, 16

[31] I. Kemelmacher-Shlizerman and S. M. Seitz, "Face reconstruction in the wild," in *2011 international conference on computer vision*, pp. 1746–1753, IEEE, 2011. 3, 15, 17, 27, 29

[32] S. Suwajanakorn, I. Kemelmacher-Shlizerman, and S. M. Seitz, "Total moving face reconstruction," in *European conference on computer vision*, pp. 796–812, Springer, 2014. 3, 15, 17, 27

[33] J. Roth, Y. Tong, and X. Liu, "Unconstrained 3d face reconstruction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2606–2615, 2015. 3, 15, 17, 27

[34] X. Cao, Z. Chen, A. Chen, X. Chen, S. Li, and J. Yu, "Sparse photometric 3d face reconstruction guided by morphable models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4635–4644, 2018. 3, 15, 17

[35] J. Roth, Y. Tong, and X. Liu, "Adaptive 3d face reconstruction from unconstrained photo collections," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4197–4206, 2016. 3, 15, 17, 18

[36] M. Piotraschke and V. Blanz, "Automated 3d face reconstruction from multiple images using quality measures," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3418–3427, 2016. 3, 15, 18, 19, 27

[37] F. Shi, H.-T. Wu, X. Tong, and J. Chai, "Automatic acquisition of high-fidelity facial performances using monocular videos," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, pp. 1–13, 2014. 3, 15, 19, 27, 96

[38] H. Jin, X. Wang, Z. Zhong, and J. Hua, "Robust 3d face modeling and reconstruction from frontal and side images," *Computer Aided Geometric Design*, vol. 50, pp. 1–13, 2017. 3, 15, 19, 27, 96

[39] G. Hu, F. Yan, J. Kittler, W. Christmas, C. H. Chan, Z. Feng, and P. Huber, "Efficient 3d morphable face model fitting," *Pattern Recognition*, vol. 67, pp. 366–379, 2017. 3, 15, 20, 27, 96

[40] A. Tuan Tran, T. Hassner, I. Masi, and G. Medioni, "Regressing robust and discriminative 3d morphable models with a very deep neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5163–5172, 2017. 3, 15, 21, 26, 32, 83

[41] P. Dou, S. K. Shah, and I. A. Kakadiaris, "End-to-end 3D face reconstruction with deep neural networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 21–26, 2017. 3, 15, 21, 26, 31, 33, 34

[42] E. Richardson, M. Sela, R. Or-El, and R. Kimmel, "Learning detailed face reconstruction from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1259–1268, 2017. 3, 9, 15, 24, 31, 33, 34, 75, 83

[43] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, "Large pose 3d face reconstruction from a single image via direct volumetric cnn regression," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1031–1039, 2017. 3, 15, 24, 32

[44] M. Sela, E. Richardson, and R. Kimmel, "Unrestricted facial geometry reconstruction using image-to-image translation," *arxiv*, 2017. 3, 15, 25, 31, 33, 34, 75, 83

[45] E. Richardson, M. Sela, and R. Kimmel, "3d face reconstruction by learning from synthetic data," in *2016 fourth international conference on 3D vision (3DV)*, pp. 460–469, IEEE, 2016. 3, 15, 26, 31, 33, 34

[46] W. Su, D. Du, X. Yang, S. Zhou, and H. Fu, "Interactive sketch-based normal map generation with deep neural networks," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 1, no. 1, p. 22, 2018. 5, 57, 58, 61, 62, 64, 80, 86

[47] O. Ronneberger, P. Fischer, and T. Brox, "U-net : Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015. 6, 24, 25, 58, 62, 63, 77, 80

[48] M. S. Knighton, D. S. Agabra, V. C. Cardei, J. A. Millers, M. A. Feeney, and W. D. McKinley, "Multiple laser scanner," Aug. 9 2011. US Patent 7,995,834. 9

[49] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE multimedia*, vol. 19, no. 2, pp. 4–10, 2012. 9

[50] R. Wang, C.-F. Chen, H. Peng, X. Liu, O. Liu, and X. Li, "Digital twin : Acquiring high-fidelity 3D avatar from a single image," *arXiv preprint arXiv :1912.03455*, 2019. 9, 15, 26, 83

[51] X. Zeng, X. Peng, and Y. Qiao, "Df2net : A dense-fine-finer network for detailed 3D face reconstruction," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2315–2324, 2019. 9, 15, 24, 32

[52] B. Egger, W. A. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, *et al.*, "3d morphable face models—past, present, and future," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 5, pp. 1–38, 2020. 9, 14

[53] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt, "State of the art on monocular 3d face reconstruction, tracking, and applications," *Computer Graphics Forum*, vol. 37, pp. 523–550, 05 2018. 9

[54] G. Stylianou and A. Lanitis, "Image based 3d face reconstruction : a survey," *International Journal of Image and Graphics*, vol. 9, no. 02, pp. 217–250, 2009. 9

[55] A. Tuan Tran, T. Hassner, I. Masi, E. Paz, Y. Nirkin, and G. Medioni, "Extreme 3d face reconstruction : Seeing through occlusions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3935–3944, 2018. 10, 15, 26, 32

[56] J. Booth, A. Roussos, E. Ververas, E. Antonakos, S. Ploumpis, Y. Panagakis, and S. Zafeiriou, "3d reconstruction of "in-the-wild" faces in images and videos," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 11, pp. 2638–2652, 2018. 10

[57] J. R. Tena, M. Hamouz, A. Hilton, and J. Illingworth, "A validated method for dense non-rigid 3d face registration," in *2006 IEEE International Conference on Video and Signal Based Surveillance*, pp. 81–81, IEEE, 2006. 13

[58] B. Amberg, S. Romdhani, and T. Vetter, "Optimal step nonrigid icp algorithms for surface registration," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2007. 13, 14

[59] A. Myronenko and X. Song, "Point set registration : Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010. 14

[60] H. Yi, C. Li, Q. Cao, X. Shen, S. Li, G. Wang, and Y.-W. Tai, "Mmface : A multi-metric regression network for unconstrained face reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7663–7672, 2019. 15, 21, 32

[61] F. Wu, L. Bao, Y. Chen, Y. Ling, Y. Song, S. Li, K. N. Ngan, and W. Liu, "Mvf-net : Multi-view 3D face morphable model regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 959–968, 2019. 15, 21, 83

[62] X. Tu, J. Zhao, M. Xie, Z. Jiang, A. Balamurugan, Y. Luo, Y. Zhao, L. He, Z. Ma, and J. Feng, "3d face reconstruction from a single image assisted by 2d face images in the wild," *IEEE Transactions on Multimedia*, 2020. 15, 23

[63] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, "Unsupervised training for 3D morphable model regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8377–8386, 2018. 15, 25, 31, 83

[64] Y. Zhou, J. Deng, I. Kotsia, and S. Zafeiriou, "Dense 3d face decoding over 2500fps : Joint texture & shape convolutional mesh decoders," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1097–1106, 2019. 15, 25

[65] G. Trigeorgis, P. Snape, S. Zafeiriou, and I. Kokkinos, "Normal Estimation For "in-the-wild" Faces Using Fully Convolutional Networks," in *Computer Vision and Pattern Recognition (CVPR)*, 2017. 15, 26, 32, 59, 61, 70

[66] M. Feng, S. Zulqarnain Gilani, Y. Wang, and A. Mian, "3D face reconstruction from light field images : A model-free approach," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 501–518, 2018. 15, 26, 73, 75

[67] B. K. Horn and M. J. Brooks, *Shape from shading*. MIT press, 1989. 15

[68] J.-D. Durou, M. Falcone, and M. Sagona, "Numerical methods for shape-from-shading : A new survey with benchmarks," *Computer Vision and Image Understanding*, vol. 109, no. 1, pp. 22–43, 2008. 15

[69] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical engineering*, vol. 19, no. 1, p. 191139, 1980. 15, 17

[70] J. Ackermann and M. Goesele, "A survey of photometric stereo techniques," *Foundations and Trends® in Computer Graphics and Vision*, vol. 9, no. 3-4, pp. 149–254, 2015. 15

[71] Y. Quéau, J.-D. Durou, and J.-F. Aujol, "Normal integration : a survey," *Journal of Mathematical Imaging and Vision*, vol. 60, no. 4, pp. 576–593, 2018. 16

[72] M. Bähr, M. Breuß, Y. Quéau, A. S. Boroujerdi, and J.-D. Durou, "Fast and accurate surface normal integration on non-rectangular domains," *Computational Visual Media*, vol. 3, no. 2, pp. 107–129, 2017. 16

[73] W. Y. Zhao and R. Chellappa, "Illumination-insensitive face recognition using symmetric shape-from-shading," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 1, pp. 286–293, IEEE, 2000. 16, 27

[74] J. J. Atick, P. A. Griffin, and A. N. Redlich, "Statistical approach to shape from shading : Reconstruction of three-dimensional face surfaces from single two-dimensional images," *Neural computation*, vol. 8, no. 6, pp. 1321–1340, 1996. 16

[75] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel, "Laplacian surface editing," in *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pp. 175–184, 2004. 16

[76] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, 1975. 19, 20, 31, 35

[77] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 21, 23

[78] I. Masi, A. T. Tran, T. Hassner, J. T. Leksut, and G. Medioni, "Do we really need to collect millions of faces for effective face recognition ?," in *European conference on computer vision*, pp. 579–596, Springer, 2016. 21

[79] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*, pp. 2642–2651, PMLR, 2017. 22

[80] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017. 22

[81] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017. 22

[82] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv :1406.2661*, 2014. 23

[83] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv :1701.07875*, 2017. 23, 58

[84] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, "Recent progress on generative adversarial networks (gans) : A survey," *IEEE Access*, vol. 7, pp. 36322–36333, 2019. 23

[85] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, "How generative adversarial networks and their variants work : An overview," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–43, 2019. 23

[86] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *arXiv preprint arXiv :1606.03498*, 2016. 23

[87] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European conference on computer vision*, pp. 483–499, Springer, 2016. 24

[88] R. Or-El, G. Rosman, A. Wetzler, R. Kimmel, and A. M. Bruckstein, "Rgbd-fusion : Real-time high precision depth recovery," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5407–5416, 2015. 26

[89] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, and H.-Y. Shum, "Mesh editing with poisson-based gradient field manipulation," *ACM Transactions on Graphics*, vol. 23, 08 2004. 26

[90] J. D'Errico, "Surface fitting using gridfit," *MATLAB central file exchange*, vol. 643, 2005. 26

[91] R. T. Frankot and R. Chellappa, "A method for enforcing integrability in shape from shading algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 4, pp. 439–451, 1988. 26, 59

[92] B. K. Horn and M. J. Brooks, "The variational approach to shape from shading," *Computer Vision, Graphics, and Image Processing*, vol. 33, no. 2, pp. 174–208, 1986. 28, 66

[93] J. Han, S. Karaoglu, H.-A. Le, and T. Gevers, "Improving face detection performance with 3d-rendered synthetic data," *arXiv preprint arXiv :1812.07363*, 2018. 30

[94] F. Kuhnke and J. Ostermann, "Deep head pose estimation using synthetic images and partial adversarial domain adaption for continuous label spaces," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 10164–10173, 2019. 30

[95] A. Kortylewski, A. Schneider, T. Gerig, B. Egger, A. Morel-Forster, and T. Vetter, "Training deep face recognition systems with synthetic data," *arXiv preprint arXiv :1802.05891*, 2018. 30

[96] I. Abbasnejad, S. Sridharan, D. Nguyen, S. Denman, C. Fookes, and S. Lucey, "Using synthetic data to improve facial expression analysis with 3d convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1609–1618, 2017. 30

[97] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from synthetic humans," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 109–117, 2017. 30

[98] J. Piao, C. Qian, and H. Li, "Semi-supervised monocular 3d face reconstruction with end-to-end shape-preserved domain transfer," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9398–9407, 2019. 31

[99] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs, "Sfsnet : Learning shape, reflectance and illuminance of facesin the wild'," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6296–6305, 2018. 31, 59, 60, 70, 97

[100] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses : A 3d solution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 146–155, 2016. 31

[101] S. Romdhani and T. Vetter, "Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior," in *2005 IEEE Computer Society*

Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, pp. 986–993, IEEE, 2005. 31

[102] P. S. Heckbert, "Survey of texture mapping," *IEEE computer graphics and applications*, vol. 6, no. 11, pp. 56–67, 1986. 34

[103] B. Lévy and J.-L. Mallet, "Paramétrisation des surfaces triangulées," *Revue Internationale de CFAO et d'informatique graphique et d'informatique graphique*, vol. 15, no. 1, pp. 25–42, 2000. 34

[104] B. O. Community, *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 35

[105] M. Pharr, W. Jakob, and G. Humphreys, *Physically based rendering : From theory to implementation*. Morgan Kaufmann, 2016. 35

[106] J. F. Blinn, "Models of light reflection for computer synthesized pictures," in *Proceedings of the 4th annual conference on Computer graphics and interactive techniques*, pp. 192–198, 1977. 35

[107] S. Du, N. Zheng, S. Ying, Q. You, and Y. Wu, "An extension of the icp algorithm considering scale factor," in *2007 IEEE International Conference on Image Processing*, vol. 5, pp. V–193, IEEE, 2007. 38

[108] E. ARTS., "The sims resource.," 2015. https://www.thesimsresource.com/. 40, 94

[109] Y. Zhou, L. Hu, J. Xing, W. Chen, H.-W. Kung, X. Tong, and H. Li, "Hairnet : Single-view hair reconstruction using convolutional neural networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 235–251, 2018. 41

[110] W. Liang and Z. Huang, "An enhanced framework for real-time hair animation," in *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.*, pp. 467–471, IEEE, 2003. 41, 42, 44

[111] J. Xing, K. Nagano, W. Chen, H. Xu, L.-y. Wei, Y. Zhao, J. Lu, B. Kim, and H. Li, "Hairbrush for immersive data-driven hair modeling," in *Proceedings of the 32Nd Annual ACM Symposium on User Interface Software and Technology*, pp. 263–279, 2019. 41, 44, 45

[112] C. K. Koh and Z. Huang, "Modeling and animation of human hair in strips," in *SIGGRAPH*, 2000. 41, 44, 45

[113] C. K. Koh and Z. Huang, "A simple physics model to animate human hair modeled in 2d strips in real time," in *Computer Animation and Simulation 2001*, pp. 127–138, Springer, 2001. 41, 44, 45, 49

[114] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern recognition*, vol. 36, no. 2, pp. 451–461, 2003. 41, 42

[115] S. Laroze and J.-J. Barrau, *Mécanique des structures.* Eyrolles, 1988. 42

[116] M. Koster, J. Haber, and H.-P. Seidel, "Real-time rendering of human hair using programmable graphics hardware," in *Proceedings Computer Graphics International, 2004.*, pp. 248–256, IEEE, 2004. 42

[117] T. Scheuermann, "Practical real-time hair rendering and shading," in *ACM SIGGRAPH 2004 Sketches*, SIGGRAPH '04, (New York, NY, USA), p. 147, Association for Computing Machinery, 2004. 44, 45

[118] J. T. Kajiya and T. L. Kay, "Rendering fur with three dimensional textures," in *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '89, (New York, NY, USA), p. 271–280, Association for Computing Machinery, 1989. 46, 48

[119] Y. Yu, "Modeling realistic virtual hairstyles," in *Proceedings Ninth Pacific Conference on Computer Graphics and Applications. Pacific Graphics 2001*, pp. 295–304, IEEE, 2001. 46

[120] L. Li, R. Li, and J. Yu, "A mass spring based 3d virtual hair dynamic system for straight and curly hair," in *2016 35th Chinese Control Conference (CCC)*, pp. 6982–6987, IEEE, 2016. 46

[121] T.-Y. Kim and U. Neumann, "Interactive multiresolution hair modeling and editing," *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 620–629, 2002. 46

[122] T. Scheuermann, "Hair rendering and shading," *ShaderX3-Advanced Rendering with DirectX and OpenGL*, pp. 239–250, 2004. 48

[123] T. Kim *et al.*, "Algorithms for hardware accelerated hair rendering," in *The 30th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2003), San Diego, CA*, 2003. 48

[124] L. Wang, Y. Yu, K. Zhou, and B. Guo, "Example-based hair geometry synthesis," in *ACM SIGGRAPH 2009 Papers*, SIGGRAPH '09, (New York, NY, USA), Association for Computing Machinery, 2009. 48

[125] K. Klasing, D. Althoff, D. Wollherr, and M. Buss, "Comparison of surface normal estimation methods for range sensing applications," in *2009 IEEE International Conference on Robotics and Automation*, pp. 3206–3211, IEEE, 2009. 52, 62

[126] Z. Lun, M. Gadelha, E. Kalogerakis, S. Maji, and R. Wang, "3d shape reconstruction from sketches via multi-view convolutional networks," in *2017 International Conference on 3D Vision (3DV)*, pp. 67–77, IEEE, 2017. 59

[127] M. Hudon, M. Grogan, R. Pagés, and A. Smolić, "Deep normal estimation for automatic shading of hand-drawn characters," in *European Conference on Computer Vision*, pp. 246–262, Springer, 2018. 59

[128] A. Bansal, B. Russell, and A. Gupta, "Marr revisited : 2d-3d alignment via surface normal prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5965–5974, 2016. 59

[129] J. Qiu, Z. Cui, Y. Zhang, X. Zhang, S. Liu, B. Zeng, and M. Pollefeys, "Deeplidar : Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3313–3322, 2019. 59

[130] X. Wang, D. Fouhey, and A. Gupta, "Designing deep networks for surface normal estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 539–547, 2015. 59

[131] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras, "Neural face editing with intrinsic image disentangling," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5541–5550, 2017. 60

[132] V. F. Abrevaya, A. Boukhayma, P. H. Torr, and E. Boyer, "Cross-modal deep face normals with deactivable skip connections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4979–4989, 2020. 60, 70, 97

[133] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv :1411.1784*, 2014. 62

[134] R. T. Frankot and R. Chellappa, "A method for enforcing integrability in shape from shading algorithms," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 10, no. 4, pp. 439–451, 1988. 66

[135] Y. Quéau and J.-D. Durou, "Intégration d'un champ de gradient rapide et robuste aux discontinuités application à la stéréophotométrie," in *Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014*, 2014. 66

[136] Y. Quéau, J.-D. Durou, and J.-F. Aujol, "Variational methods for normal integration," *Journal of Mathematical Imaging and Vision*, vol. 60, no. 4, pp. 609–632, 2018. 66

[137] M. Harker and P. O'Leary, "Least squares surface reconstruction from measured gradient fields," in *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–7, IEEE, 2008. 66

[138] S. Agarwal, K. Mierle, and Others, "Ceres solver." http://ceres-solver.org. 67, 84

[139] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco : Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014. 69, 86

[140] Y. Guo, J. Cai, B. Jiang, J. Zheng, *et al.*, "Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 6, pp. 1294–1307, 2018. 83

[141] S. Ploumpis, E. Ververas, E. O'Sullivan, S. Moschoglou, H. Wang, N. Pears, W. Smith, B. Gecer, and S. P. Zafeiriou, "Towards a complete 3d morphable model of the human head," *IEEE transactions on pattern analysis and machine intelligence*, 2020. 97

[142] S. Bi, K. Sunkavalli, F. Perazzi, E. Shechtman, V. G. Kim, and R. Ramamoorthi, "Deep cg2real : Synthetic-to-real translation via image disentanglement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2730–2739, 2019. 97

[143] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019. 97