Imperial College of Science, Technology and Medicine
Department of Computing

Capturing, Modeling and Reconstructing Photorealistic Avatars

Alexandros Lattas
May 2023

Supervised by
Prof Stefanos Zafeiriou and Prof Abhijeet Ghosh

Submitted in part fulfilment of the requirements for the degree of PhD in Computing and the Diploma of Imperial College London. This thesis is entirely my own work, and, except where otherwise indicated, describes my own research.

Abstract

This Thesis introduces several methods and a dataset, aiming to improve the capturing and inference of human faces, in a form that can be photorealistically rendered by computer applications. The proposed methods present advances in facial reflectance acquisition, from capturing to modeling and inference.

In the first part of the Thesis, we present various improvements on a state-of-the-art facial capturing system, such as a Light Stage. More specifically, we work on an existing diffuse-specular separation method that does not require light polarization and further optimize it while making it applicable to multi-view capture. We also use this improved setup to capture the largest dataset of high-resolution facial reflectance to date, which we have opened to the research community.

In the second part, we present two novel facial capturing setups, and an appropriate facial reflectance separation method. They are the first such systems consisting solely of off-the-shelf devices, while being modular and portable, and thus enabling capturing vast datasets of facial reflectance, at a reduced cost. Supporting static and dynamic facial capture, these systems allow the capture of even larger datasets of facial reflectance that can fuel deep-learning research and create race, gender, and age-balanced datasets.

In the third part of this Thesis, we introduce the first method for high-resolution facial reflectance inference from a single arbitrary facial image. Our method AvatarMe and its extension, AvatarMe^{++}, use a combination of 3D Morphable Models for facial shape reconstruction, adversarial image-translation models and super-resolution models, to transform reconstructed facial textures to high-resolution facial reflectance. We show that such a method is capable of retrieving UV maps of spatially varying facial reflectance that is render-ready in typical rendering applications without requiring capturing setups.

Finally, we introduce the first deep learning-based 3D Morphable Model of facial reflectance, called FitMe. FitMe proposes a style-based Generative Adversarial Network, which is multi-modal and branched in both the discriminator and the generator, and is capable of concurrently generating high-quality facial diffuse albedo, specular albedo and normals. Moreover, by using an accurate diffuse-specular differentiable renderer, FitMe can retrieve state-of-the-art facial shape and reflectance with an iterative optimization.

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC). Under this license, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose. When reusing or sharing this work, ensure you make the license terms clear to others by naming the license and linking to the license text. Where a job has been adapted, you should indicate that the work has been changed and describe those changes. Please seek permission from the copyright holder for uses of this work that are not included in this license or permitted under UK Copyright Law.

Acknowledgments

I met my PhD supervisor, Prof Stefanos Zafeiriou, through a cold email, which led to my first visit to Imperial College London, back in 2015. At that time, I was an exchange student at University College London, and was starting to get fascinated by the field of machine learning. First and foremost, I would like to thank him, for enormously inspiring me, first with his course in Advanced Statistical Machine Learning, which led to a fantastic MSc project in Generative Adversarial Networks, and then to the beginning of this PhD. Stefanos' guidance in the beginning, and later his trust in my independence, forged my character and enabled me to complete successfully, and more importantly, enjoy completing this work. I would also like to thank my supervisor Prof Abhijeet Ghosh, who made me fall in love with the field of Computer Graphics, and who supported me attentively throughout my PhD studies. I will always cherish our talks during lunches, or conference dinners, and will always admire his attention to detail and caring spirit. I would also be remiss not to thank my BSc Thesis supervisor and mentor, Prof Diomidis Spinellis, whose supervision and guidance during my formative BSc years enhanced and solidified my interest in Computer Science and research.

Even more, I would like to thank my friend and colleague, Dr Stylianos Moschoglou, who was the closest to me throughout my research journey, from the moment I considered joining the lab to writing this Thesis. His guidance was critical, and his support and insight were genuinely invaluable throughout these years, while our collaboration and friendship made this journey exciting and enjoyable. I also thank Dr Stylianos Ploumpis, who taught me so much about 3D modeling, and was always by my side as a friend and a colleague. In the same manner, I thank Dr Jiankang Deng and Dr Baris Gecer, for their help and collaboration throughout my studies. Finally, I would like to thank all those with whom

we fruitfully collaborated, including Yuliya, Panagiotis, Shiya, Yiming, Jay, Ekin, Foivos and Stathis.

I would also like to thank Hellenic Petroleum sincerely, who awarded me with a substantial MSc scholarship, which enabled me to start my journey at Imperial College London.

Finally, I would like to thank these people dearest to me, who helped, each in their own way, my research work and studies. First and foremost, my girlfriend Andrea, who is also now writing her PhD Thesis, having shared this research journey with me. I could have never imagined finding someone to support me at that level, having spent countless days (and nights) by my side while producing this work. And I also thank my best friends, who also share the research journey: Fotis, Vangelis, Michalis, and Giorgos, and those who have made every bit of these years in London an enjoyable experience, including: Manos, Tatiana, Liza, Eleni, Konstantinos, Ioanna, Despoina, Giorgos, Paul, Alexis and Mochi. And from the bottom of my heart, I thank my parents, Froso and Giorgos, my brother Nikolas, my grandparents, and my godparents, Eleni and Giorgos, who have always supported me and have sacrificed a lot, for me to complete this journey. This work is dedicated to them.

Dedication

To Andrea, my parents, brother, and godparents.

Acronyms

GAN	Generative Adversarial Network
CNN	Convolutional Neural Network
PCA	Principal Component Analysis
SVD	Singular Value Decomposition
3DMM	3D Morphable Model
BRDF	Bidirectional Reflectance Distribution Function
SVBRDF	Spatially-Varying Bidirectional Reflectance Distribution Function

Notation

$\sum$	Summation
$\prod$	Product
$\mathbb{R}$	Field of Real Numbers
$\lvert x \rvert$	Absolute value of x
$\sqrt{x}$	Square root of x
$\mathcal{N}(\mu, \sigma^2)$	Normal Distribution with mean μ and variance σ^2
$\mathbf{x}$	Column-vector with components x_i
$\mathbf{X}$	Matrix with components X_{ij}
$\mathbf{X}^{-1}$	Inverse of $\mathbf{X}$
$\mathbf{X}^{\top}$	Transpose of $\mathbf{X}$
$\mathbf{X} \odot \mathbf{Y}$	Hadamard product of $\mathbf{X}$ and $\mathbf{Y}$

$\mathbf{S}$	Shape geometry
$\mathbf{T}$	Texture (shaded)
$\mathbf{A}_D$	Diffuse albedo
$\mathbf{A}_S$	Specular albedo
$\mathbf{N}$	Surface normals
$\mathbf{N}_D$	Diffuse normals
$\mathbf{N}_S$	Specular normals
α	Specular roughness term

Contents

Acronyms Table **vi**

Notation Table **ix**

1 Introduction **1**

1.1 Problem Scope and Challenges 1

1.2 Contributions and Thesis Overview 4

1.3 Impact and Applications 5

1.4 Publications 8

1.4.1 Relevant Publications 8

1.4.2 Other Publications 9

2 Background **11**

2.1 Photorealistic Facial Avatars Overview 12

2.2 Photorealistic Facial Rendering 13

2.2.1 Differentiable Rendering 13

2.2.2 Shading Models 15

2.2.3 Facial Rendering Overview 18

2.3 Facial Geometry and Reflectance Capturing 20

2.3.1 Geometry Acquisition 20
2.3.2 Active Illumination Facial Scanners 21
2.3.3 Passive Illumination Facial Scanners 22
2.3.4 Smaller Scale Systems 23
2.3.5 Non-skin capturing 24
2.4 Deep Learning Methods for Image Synthesis and Manipulation 25
2.4.1 Generative Adversarial Networks for Image Synthesis 25
2.4.2 Generative Adversarial Networks for Image Manipulation 27
2.5 3D Facial Synthesis and Reconstruction 31
2.5.1 3D Facial Geometry and Texture Modeling and Synthesis 31
2.5.2 3D Facial Geometry and Texture Reconstruction from a Single Image 33
2.6 Photorealistic 3DMMs 36
2.6.1 Image Translation-based Methods 37
2.6.2 Combinatory Methods 37
2.6.3 Planar Material Acquisition 38

3 Capturing Human Facial Assets at Scale 39

3.1 Diffuse-Specular Separation Optimization 40
3.1.1 Introduction 40
3.1.2 Diffuse-Specular Separation Overview 41
3.1.3 Processing Binary Gradients 43
3.1.4 Results 45
3.2 Capturing and Reconstruction Automation 47

3.2.1 The Capturing System 48
3.2.2 Capturing 48
3.2.3 Transfer and Pre-processing 49
3.2.4 Geometry Acquisition 49
3.2.5 Template Registration and UV Texturing 50
3.3 RealFaceDB Dataset 52
3.4 Conclusion 54

4 Practical and Scalable Desktop-based High-Quality Facial Capture 55

4.1 Introduction 57
4.2 Related Work 58
4.2.1 Active Illumination 58
4.2.2 Passive Capture 60
4.3 Desktop-based Capture System 61
4.3.1 Tablet-based Setup 61
4.3.2 Monitor-based Setup 63
4.4 Illumination 63
4.4.1 Modulated Binary Illumination 63
4.4.2 Spectral Multiplexing and Cross-Talk 64
4.4.3 Form-Factor Modulation 66
4.4.4 Multi-view Capture 66
4.5 Method 67
4.5.1 Acquisition using White Illumination 67

4.5.2 Color-Multiplexed Illumination 69
4.5.3 Specular Roughness Estimation 70
4.5.4 Dynamic Capture 70
4.5.5 Base Geometry Acquisition 71
4.6 Experiments 71
4.6.1 Evaluation 71
4.6.2 Static Capture 72
4.6.3 Dynamic Capture 72
4.7 Limitations 73
4.8 Conclusions 74

5 Facial Appearance Reconstruction from a Single-Image 77

5.1 Introduction 79
5.2 Related Work 82
5.2.1 Facial Geometry and Reflectance Capture 82
5.2.2 Facial Geometry and Texture Estimation 83
5.2.3 Image-to-Image Translation 84
5.2.4 Differentiable Rendering 84
5.2.5 Facial BRDF Estimation 85
5.3 Data Acquisition 86
5.3.1 Training Data Capturing Setup 86
5.3.2 Data Augmentation 87
5.4 Method 88

5.4.1 Overview 88

5.4.2 Initial Geometry and Texture Estimation 89

5.4.3 3DMM Capturing Environment Estimation 90

5.4.4 Super Resolution 91

5.4.5 Reflectance Inference with AvatarMe 91

5.4.6 Photorealistic Differentiable Facial Rendering 95

5.4.7 Reflectance Inference with AvatarMe^{++} 99

5.5 Experiments 102

5.5.1 Implementation Details 102

5.5.2 Evaluation 104

5.5.3 Ablation 106

5.5.4 Limitations 108

5.6 Conclusion 109

6 Facial Reflectance Generation and Fitting to Arbitrary Images 114

6.1 Introduction 115

6.2 Related Work 120

6.2.1 3D Morphable Models 120

6.2.2 Deep Generative Networks 121

6.2.3 Facial Reflectance Acquisition 121

6.3 Method 122

6.3.1 The FitMe Deep 3D Morphable Model 123

6.3.2 Dataset Acquisition and Augmentation 124

6.3.3 Fast Differentiable Photorealistic Rendering 126
6.3.4 Fitting by Inversion through Rendering 127
6.4 Experiments 130
6.4.1 Implementation Details 130
6.4.2 Single-Image Reconstruction 131
6.4.3 Multi-Image Reconstruction 132
6.4.4 Dataset Augmentation 133
6.5 Model Manipulation 134
6.6 Ablation Study 135
6.7 Additional Results 137
6.8 Implementation Details 139
6.9 Limitations 140
6.10 Conclusion 140

7 Conclusion **142**

7.1 Future Work 143

Bibliography **145**

List of Tables

3.1 RealFaceDB Demograhics 54

5.1 AvatarMe quantitative comparisons with state-of-the-art 104

5.2 AvatarMe^{++} method components quantitative ablation study 107

5.3 AvatarMe^{++} hyper-parameters quantitative ablation study 108

6.1 FitMe shape quantitative comparison with state-of-the-art 139

6.2 FitMe inversion hyperparamters 140

6.3 FitMe tuning hyperparameters 140

List of Figures

2.1 Example set of facial rendering assets 19

3.1 Geometry and rendered results 40

3.2 Polarized Spherical Gradient Illumination capture 41

3.3 Binary Spherical Gradient Illumination capture 42

3.4 Synthetic rotation of gradients 45

3.5 Comparison between our reflectance acquisition and the SOTA 45

3.6 Linear system solution for diffuse-specular separation 46

3.7 Comparison of mixed normals, SUV diffuse normals and spectral diffuse normals 47

3.8 Geometry and rendered comparison with state-of-the-art 47

3.9 The Imperial College Multispectral LightStage 48

3.10 Example stereo reconstruction 49

3.11 Reconstruction post-processing 50

3.12 Overview of the RealFaceDB dataset 53

4.1 Overview of practical capturing setup 56

4.2 Schematics of the two proposed desktop-based facial capture systems 61

4.3 Illumination patterns 64

4.4 SPD, sensitivity and spectral cross-talk 65

4.5 Comparison of our method with Kampouris et al. 2018 71

4.6 Comparison of tablet and monitor based setup 73

4.7 Results of specular roughness estimation 73

4.8 Completed UV maps 74

4.9 Diffuse albedo comparison with state-of-the-art 74

4.10 Additional results of our method 75

4.11 Additional side-view results of our method 75

4.12 Comparison of albedo measurement and separation 76

4.13 Comparison of our method and AvatarMe^{++} 76

4.14 Example results of dynamic capture 76

5.1 AvatarMe^{++} results summary 78

5.2 Example subject captured with our method 87

5.3 AvatarMe^{++} method summary 88

5.4 AvatarMe^{++} rendering ablation 98

5.5 AvatarMe^{++} self-occlusion autoencoder preview 99

5.6 AvatarMe^{++} training 99

5.7 AvatarMe^{++} reflectance qualitative comparison with state-of-the-art 104

5.8 AvatarMe qualitative comparisons with state-of-the-art 105

5.9 AvatarMe^{++} quantitative comparison with ground truth 106

5.10 Consistency of AvatarMe^{++} on varying input conditions 106

5.11 Generalization examples of AvatarMe^{++} 110

5.12 AvatarMe^{++} results rendered in different environments. 111

5.13 AvatarMe^{++} quantitative ablation study on method variations 112

5.14 AvatarMe^{++} qualitative ablation study on method variation 112

5.15 Comparison between AvatarMe and AvatarMe^{++} 112

5.16 AvatarMe^{++} qualitative comparison with state-of-the-art 113

5.17 AvatarMe^{++} quantitative ablation on rendering hyper-parameter 113

6.1 FitMe summary 116

6.2 FitMe method overview 117

6.3 BRDF-GAN architecture overview 123

6.4 FitMe dataset and augmentation overview 125

6.5 Comparison of albedo reconstruction on Digital Emily 128

6.6 Results of our method on challenging examples. 131

6.7 FitMe quantitative comparison based on cosine similarity 131

6.8 FitMe reconstruction qualitative comparison with state-of-the-art 132

6.9 FitMe reflectance qualitative comparison single-image reconstruction 133

6.10 FitMe fittings interpolation 133

6.11 Skin tone augmentation qualitative comparison 134

6.12 BRDF-GAN latent space manipulation 134

6.13 FitMe GAN-inversion qualitative ablation study 135

6.14 FitMe GAN-tuning qualitative ablation study 135

6.15 FitMe method-choices qualitative ablation study 136

6.16 FitMe additional multi-view results 137

6.17 FitMe additional multi-view results 138

6.18 FitMe additional multi-view results 138

Chapter 1

Introduction

Contents

1.1 Problem Scope and Challenges 1

1.2 Contributions and Thesis Overview 4

1.3 Impact and Applications 5

1.4 Publications 8

1.4.1 Relevant Publications 8

1.4.2 Other Publications 9

1.1 Problem Scope and Challenges

Human digitization has been a significant aspect of human culture since ancient times, as our ancestors sought to capture and depict other humans, by paintings on cave walls, vases, canvas, and various other surfaces. The invention of the camera revolutionized this endeavor, allowing for the preservation of accurate human likenesses in black and white, and then color, ultimately giving rise to a new form of artistic expression. As technology progressed, the introduction of motion pictures expanded the possibilities of representing humans, capturing not only their appearance but also their movements and actions. Recently, the increase in computing power and advances in computer graphics have enabled the creation of fully digital humans in three-dimensional space, providing unprecedented control and flexibility in their representation. Furthermore, the emergence of virtual and augmented reality technologies within the last decade, has opened up new

avenues for interaction with digital humans, seamlessly integrating them with our physical world. The need to capture and reproduce our own nature is ever-present, and the available tools continuously open up new possibilities.

The uncanny valley [1] is a phenomenon describing digital human representations that closely resemble real humans, which are inaccurate and can elicit discomfort and unease in certain viewers. This presents a significant challenge in the development of realistic digital humans, as achieving high levels of visual fidelity is crucial for various applications, such as virtual reality, film-making, and computer games. On the other hand, while cartoonish or stylized digital human representations may bypass the uncanny valley, they are perhaps not sufficient for scenarios that demand the authentic portrayal of human appearance. Therefore, researchers and developers must strive to overcome both the uncanny valley and the limitations of unrealistic digital humans, to create genuinely believable and engaging virtual human experiences.

The pursuit of realistic human digitization has been a long-standing problem in the fields of computer vision and graphics, with applications around entertainment, advertising, augmented and virtual reality, and even digital avatars for social media, telepresence, and the emerging metaverse. Over the past two decades, significant progress has been made in developing photorealistic rendering methods and accurate modeling of facial shape and appearance. This has been further improved by the advent of advanced acquisition techniques for high-quality 3D facial capture. However, the creation of lifelike digital characters often relies on sophisticated and costly capture systems, such as the Lightstage [2, 3, 4, 5]. As a result, the use of these acquisition techniques is typically limited to big-budget visual effects, games or organizations with the financial resources to access or construct such systems in-house. This restriction hinders the widespread adoption of 3D acquisition technologies and constrains the potential for broader applications in the realm of human digitization.

One of the most critical advancements has been the development of 3D Morphable Models (3DMM) [6]. These models employ a fitting algorithm that reconstructs facial texture and shape within a linear space, defined by a basis learned through Principal Component Analysis (PCA) [7]. While this approach has proven to be effective in representing coarse facial attributes, it cannot capture high-frequency details in both texture and geometry.

Additionally, PCA-based models face difficulties in accurately representing complex facial structures found in real-world scenarios, leading to sub-optimal performance when applied to "in-the-wild" images. Recent advancements in 3DMM fitting have attempted to address these issues (e.g. [8, 9, 10, 11]) but with limited success. While these efforts have yielded improvements in reconstructing 3D facial geometry and appearance from real-world images, challenges remain in accurately capturing and replicating the intricate details of facial appearance, especially when considering photorealistic rendering and relighting. As the field of human digitization continues to progress, overcoming these limitations is essential for generating highly realistic and believable digital human representations.

In recent decades, the fields of computer vision and machine learning have experienced significant advances, including the development of Convolutional Neural Networks (CNNs) [12]. These breakthroughs have paved the way for substantial improvements in image classification and generation tasks. In particular, the last decade has witnessed considerable progress in deep generative models, such as Generative Adversarial Networks (GANs) [13], which have achieved outstanding results in learning distributions of high-resolution 2D images of human faces. The introduction of style-based progressive generative networks [14, 15, 16, 17] has enabled the learning of meaningful latent spaces that can be traversed to reconstruct and manipulate various attributes of generated samples. Moreover, conditional GANs have enabled significant progress in image manipulation [18]. Some methods have even proven effective in learning a 2D representation of 3D facial attributes, such as UV maps [19]. By combining machine learning techniques, such as CNNs and GANs, with 3DMMs, the potential for digitizing humans can be further explored and enhanced. This fusion of technologies offers exciting new possibilities for the future of human digitization, paving the way for more realistic and accessible virtual human representations.

While the advancements in machine learning have brought significant progress in various domains, the need for large datasets remains a crucial aspect of the development process. In the context of human digitization, capturing photorealistic humans presents unique challenges due to the size and cost of high-quality 3D scanners, coupled with the inherent diversity of human appearances. Moreover, human skin is a complex, multi-layered structure that interacts with light in intricate ways, adding an additional difficulty to the data acquisition process. The scarcity of comprehensive datasets hinders the development

of machine learning methods that could generate truly lifelike digital humans. This lack of data, along with the difficulties involved in capturing the nuances of human appearance and skin properties, makes the digital human acquisition a challenging and ongoing problem to be addressed by researchers of the community.

Thus, the problem of acquiring photorealistic digital human avatars is two-fold. On the one hand, progress is needed in the relevant algorithms at the intersection of computer vision, machine learning, and computer graphics. On the other hand, datasets of large amounts of appropriate data are required. In this PhD Thesis, we introduce solutions attempting to solve both problems. The first and second chapters focus on the data, specifically on simplifying the existing capturing mechanisms and using them to capture a sizeable facial dataset. The third chapter introduces a method that transforms existing 3DMM approaches to higher quality and relightable digital human assets, while, finally, the fourth chapter fuses the latest GAN models with 3DMMs, for photorealistic human avatar generation and reconstruction.

1.2 Contributions and Thesis Overview

In the following section, we describe the contributions of this Thesis, in greater detail and per chapter:

- **Chapter 3. Capturing Human Facial Assets at Scale:** In this chapter, we describe the improvements implemented in an existing facial scanning apparatus, in order to make it more robust and automate the capture and processing while adding multi-view capabilities to its diffuse-specular separation algorithm. Moreover, we describe an automated pipeline for transforming the captured facial data to aligned meshes and texture maps. Finally, we introduce a captured dataset using the above methods, which is crucial for the work shown in the following chapters. The dataset has also been made public to the community in the form of facial reflectance patches, to preserve the anonymity of the captured subjects.

- **Chapter 4. Practical and Scalable Desktop-based High-Quality Facial Capture:** In this chapter, we introduce two novel facial scanners, that are portable,

faster, and cheaper than the existing scanners, and thus enable the acquisition of larger and more diverse datasets. Moreover, we propose a novel photometric acquisition pipeline for diffuse-specular separation and a novel design of spectral multiplexing for binary illumination. Finally, we show that these devices are capable of multi-view static and also dynamic facial appearance capture.

- **Chapter 5. Facial Appearance Reconstruction from a Single-Image:** In this chapter we introduce *AvatarMe* and its extension *AvatarMe*$^{++}$, two methods capable of transforming a facial reconstruction acquired with existing 3DMM methods or simple facial scanners, to a set of high-resolution and relightable facial textures that can be used as-is in common rendering algorithms. In this manner, we introduce several image-to-image translation networks that transform facial textures with baked illumination to diffuse and specular albedo and normals. Finally, we introduce a differentiable rendering pipeline that is capable of photorealistic and differentiable UV-space relighting, which is used to guide the model training.

- **Chapter 6. Capturing Human Facial Assets at Scale:** In the final chapter, we introduce *FitMe*, the first 3DMM capable of high-resolution facial reflectance generation, that can be photorealistically rendered by common algorithms. FitMe is based on our novel branched multi-modal style-based generator, of diffuse and specular albedo and normals, and an appropriate multi-branched discriminator. Apart from generating novel subjects from its distribution, FitMe can be optimized on unconstrained facial images, using our proposed pipeline of generator projection and tuning, and an accurate diffuse and specular rendering pipeline.

1.3 Impact and Applications

Scanning virtual humans or reconstructing them from unconstrained images, enables the unhindered use of human avatars in computer graphics applications, with a significant impact on *Human-Computer interaction*. The ability to easily reconstruct humans without the need for specialized hardware, enables people to portray themselves in virtual and augmented reality applications, and engage more directly while managing to cross the uncanny valley. Telepresence applications are poised to benefit significantly from advancements in

digital human representation. By providing realistic and immersive experiences, users can interact with others in a virtual environment as if they were physically present, unlocking new possibilities for remote communication, collaboration, and shared experiences. Compared to the current literature, the work proposed in this Thesis simplifies the capturing process and improves the avatar likeness, while the reconstruction of facial reflectance parameters enables the placement of these assets in scenes with arbitrary illumination, with accurate relighting.

Moreover, this work can be applied as-is or with minor modifications to a range of other fields, including:

- **Gaming and Social Media:** Character customization from real people or even the user themselves is a crucial aspect that benefits from these advancements. By capturing accurate and detailed 3D representations of individuals, players can create personalized avatars that closely resemble their real-world appearance, greatly enhancing the immersive experience. Moreover, the ability to generate genuine characters more easily using advanced human digitization techniques streamlines the development process for game designers. This not only saves time and resources but also allows for the creation of more diverse and visually engaging virtual worlds. As the concept of the metaverse gains traction, the role of human digitization becomes increasingly important. Integrating lifelike avatars within these virtual worlds promises to redefine online social interaction and collaboration, ultimately transforming the way people connect, work, and play.

- **Forensics:** Human digitization can significantly impact the field of forensics by providing valuable tools for criminal investigations. Generating photorealistic faces from police sketches allows for more accurate visualizations of suspects, aiding law enforcement in their identification efforts. Additionally, the ability to manipulate digital human models in terms of age, pose, and expression can provide critical insights and explore various scenarios during investigations. Advancements in human digitization can also improve face-matching algorithms by incorporating highly detailed 3D facial models.

- **Health and Skin Analysis:** By accurately capturing albedo information, re-

searchers and healthcare professionals can gain insights into an individual's skin characteristics, such as melanin levels. This information can be valuable for understanding skin health and guiding personalized skincare recommendations or treatments. Furthermore, comparing facial characteristics through detailed 3D representations can aid in the assessment of various health-related factors. By analyzing these digital models, healthcare professionals can identify patterns or anomalies, potentially leading to the early detection of medical conditions or a deeper understanding of an individual's unique characteristics

- **Fashion:** In the fashion industry, human digitization opens up new avenues for enhancing the shopping experience and promoting customer engagement. By allowing users to create accurate 3D representations of themselves, they can visualize how various fashion items, such as makeup, glasses, or hats, would look on them. This personalized approach helps customers make informed decisions but also fosters a more immersive and enjoyable shopping experience, ultimately driving customer satisfaction and loyalty.

- **Datasets:** The creation and availability of datasets play a vital role in advancing machine learning algorithms and their applications. Human digitization can contribute to the development of privacy-free datasets of human representations, which can be used to train algorithms without infringing on personal privacy rights. These anonymized yet realistic models can help researchers and developers refine their techniques while adhering to ethical guidelines and data protection regulations. Additionally, human digitization allows for placing digital humans in controlled virtual environments with adjustable parameters. This capability enables the generation of diverse and comprehensive datasets that simulate various scenarios, lighting conditions, or interactions. Such datasets can be instrumental in enhancing the performance and adaptability of machine learning algorithms across a broad range of applications and industries.

1.4 Publications

1.4.1 Relevant Publications

The majority of the work included in this thesis is based on published work in peer-reviewed conference proceedings and journals. The text of each chapter is based on these works as described below, and is kept as close to the original publication as possible to aid the reader, while maintaining a coherent structure for the Thesis. In summary, the following chapters contain:

- **Chapter 3:**
  - *A. Lattas, M. Wang, S. Zafeiriou, and A. Ghosh, "Multi-view facial capture using binary spherical gradient illumination," in* ACM SIGGRAPH 2019 Posters*, pp. 1–2, 2019*,
  - The extensive work carried out to automate the Imperial College Light Stage and improve its robustness, to produce the vast dataset RealFaceDB, used in the following chapter.
- **Chapter 4:**
  - *A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera, G. Chawla, and A. Ghosh, "Desktop-based high-quality facial capture for everyone," in* SIGGRAPH Talks*, 2022*,
  - *A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera, G. Chawla, and A. Ghosh, "Practical and scalable desktop-based high-quality facial capture," in* Proceedings of the European Conference on Computer Vision*, 2022*
- **Chapter 5:**
  - *A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "AvatarMe: Realistically renderable 3d facial reconstruction" in-the-wild"," in* Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 760–769, 2020*,

  - *A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, A. Ghosh, and S. Zafeiriou, "AvatarMe++: Facial shape and BRDF inference with photorealistic rendering-aware GANs,"* IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2021.*

- **Chapter 6:**

  - *A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, J. Deng, and S. Zafeiriou, "Fitme: Deep photorealistic 3D morphable model avatars," in* Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8629–8640, June 2023*

1.4.2 Other Publications

In addition, during these studies, the author participated and co-authored the following works, many of which expand on the subjects of Chapters 4,5, and 6. They are listed in chronological order:

- *P. Tzirakis, A. Papaioannou, A. Lattas, M. Tarasiou, B. Schuller, and S. Zafeiriou, "Synthesising 3D facial motion from "in-the-wild" speech," in* 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pp. 265–272, IEEE, 2020*

- *B. Gecer, A. Lattas, S. Ploumpis, J. Deng, A. Papaioannou, S. Moschoglou, and S. Zafeiriou, "Synthesizing coupled 3D face modalities by trunk-branch generative adversarial networks," in* European conference on computer vision*, pp. 415–433, Springer, 2020*

- *Y. Gitlina, G. C. Guarnera, D. S. Dhillon, J. Hansen, A. Lattas, D. Pai, and A. Ghosh, "Practical measurement and reconstruction of spectral skin reflectance," in* Computer Graphics Forum*, pp. 75–89, Wiley Online Library, 2020*

- *J. Deng, J. Guo, J. Yang, A. Lattas, and S. Zafeiriou, "Variational prototype learning for deep face recognition," in* Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11906–11915, 2021*

- *J. Guo, J. Deng, A. Lattas, and S. Zafeiriou, "Sample and computation redistribution for efficient face detection," in* International Conference on Learning Representations*, 2021*

- *S. Li, U. Waheed, M. Bahshwan, L. Z. Wang, L. M. Kalossaka, J. Choi, F. Kundrak, A. Lattas, S. Ploumpis, S. Zafeiriou,* et al.*, "A scalable mass customisation design process for 3D-printed respirator mask to combat covid-19,"* Rapid Prototyping Journal*, 2021*

- *Y. Miao, A. Lattas, J. Deng, J. Han, and S. Zafeiriou, "Physically-based face rendering for NIR-VIS face recognition," in* Advances in Neural Information Processing Systems *(A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, eds.), 2022*

- *X. Ren, A. Lattas, B. Gecer, J. Deng, C. Ma, X. Yang, and S. Zafeiriou, "Facial geometric detail recovery via implicit representation," in* 2023 17th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2023)*, IEEE, 2023*

- *J. Guo, J. Yu, A. Lattas, and J. Deng, "Perspective reconstruction of human faces by joint mesh and landmark regression,"* European Conference on Computer Vision (ECCV) Workshops*, 2022*

- *S. Galanakis, B. Gecer, A. Lattas, and S. Zafeiriou, "3DMM-RF: Convolutional radiance fields for 3D face modeling," in* Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3536–3547, 2023*

- *F. P. Papantoniou, A. Lattas, S. Moschoglou, and S. Zafeiriou, "Relightify: Relightable 3D faces from a single image via diffusion models," in* Proc. IEEE International Conference on Computer Vision*, 2023*

Finally, two patent applications were published, with the first being accepted, while the second is pending at the current date:

- *A. Ghosh, C. Kampouris, A. Lattas, and S. Zafeiriou, "Image processing for diffuse-specular separation," 2021. US Patent 10,964,099*

- *S. Zafeiriou, A. Lattas, S. Moschoglou, S. Ploumpis, and B. Gecer, "Three-dimensional facial reconstruction." UK Patent App GB2593441A (Pending)*

Chapter 2

Background

Contents

2.1 Photorealistic Facial Avatars Overview 12

2.2 Photorealistic Facial Rendering 13

2.2.1 Differentiable Rendering 13

2.2.2 Shading Models 15

2.2.3 Facial Rendering Overview 18

2.3 Facial Geometry and Reflectance Capturing 20

2.3.1 Geometry Acquisition 20

2.3.2 Active Illumination Facial Scanners 21

2.3.3 Passive Illumination Facial Scanners 22

2.3.4 Smaller Scale Systems 23

2.3.5 Non-skin capturing 24

2.4 Deep Learning Methods for Image Synthesis and Manipulation 25

2.4.1 Generative Adversarial Networks for Image Synthesis 25

2.4.2 Generative Adversarial Networks for Image Manipulation 27

2.5 3D Facial Synthesis and Reconstruction 31

2.5.1 3D Facial Geometry and Texture Modeling and Synthesis 31

2.5.2 3D Facial Geometry and Texture Reconstruction from a Single Image 33

2.6 Photorealistic 3DMMs 36

2.6.1 Image Translation-based Methods 37

2.6.2 Combinatory Methods 37
2.6.3 Planar Material Acquisition 38

2.1 Photorealistic Facial Avatars Overview

The chapters of this work explore the acquisition of photorealistic human facial assets, which are also increasingly referred to as *Avatars* in the scientific community. The scope of this analysis is facial assets, whose acquisition and rendering are challenging, with a multitude of works published in the span of 3 decades. For a single subject, such facial assets typically consist of a shape representation that captures the geometry of the given subject and a set of textures that describe spatially varying facial attributes, such as skin color. Such assets can be used by rendering algorithms (Sec. 2.2) to generate images of the assets in arbitrary environments, or used in applications such as games, virtual and augmented reality, etc. Assets can be captured with large and expensive facial scanners (Sec. 2.3), manually created by graphic artists, or inferred using machine learning models (Sec. 2.5). This Thesis contains work relating to the simplification of the capturing process, and the improvement of algorithms for machine learning inference of facial assets, from unconstrained facial images.

Both sides of this 3D reconstruction problem belong to the well-studied field of Inverse Rendering, in the intersection of Graphics and Computer Vision. *Inverse Rendering* describes the problem of analyzing images, to acquire the geometry of a scene, the properties of its illumination sources and the reflectance properties of its objects' materials. Following the standard definitions [39], Inverse Rendering problems can be classified into (a) *Inverse Lighting* problems, where the properties of the illumination sources are unknown, (b) *Inverse Geometry*, where the geometry of an object is unknown (c) *Inverse Reflectance* problems, where the texture or the reflectance distribution function of an object is unknown. Photorealistic 3D reconstruction of faces incorporates a combination of Inverse Geometry and Inverse Reflectance and often Inverse Lighting.

In the following sections, we describe the required background around inverse rendering and machine learning, while keeping the scope around facial assets. Sec. 2.2 introduces the reader to the basics of facial rendering, and differentiable rendering, which is required

for optimization, Sec. 2.3 describes the prior work in facial assets capturing using scanning systems, Sec. 2.4 provides an overview of the basics of machine and deep learning used in facial reconstruction, Sec. 2.5.1 describes the prior work in 3D facial asset synthesis and finally Sec. 2.5.2 analyses the prior work relating to photorealistic 3D facial reconstruction.

2.2 Photorealistic Facial Rendering

Rendering comprises a set of functions that synthesize an image from 3D assets. These 3D assets are commonly described by a group of vertices with polygonal faces and associated textures. The main steps are a) geometry processing, b) rasterization, and c) shading. Through *Geometry processing*, a 3D model's coordinates are transformed from its own *model space* to a shared *world space*, and from it to a desired *camera space*, based on a camera's parameters. Both transformations can be described with matrix multiplication [40]. The transformation to the camera space can either be *orthographic*, a parallel projection, or *perspective*. Then, through *rasterization*, the visible polygons to the camera are selected, and fragments of those are associated with each camera's pixels [40]. Finally, through *shading*, a function transforms each pixel's fragment to the final color of the rendered image, while overlapping fragments are blended based on their distance from the camera [40].

2.2.1 Differentiable Rendering

The geometry processing and most common pixel shaders are by definition differentiable. However, the rasterization and fragment blending (typically using z-buffering) steps are discontinuous functions and cannot be differentiated. Works on *differentiable rendering* are either simplifying or changing the above rendering pipeline, so it can be incorporated into an iterative optimization pipeline.

Multiple works in the past have incorporated differentiable rendering models to solve specific problems, such as geometry inference or reflectance acquisition, with limited success [39]. Nevertheless, the advent of deep neural networks has sparked the interest in fully differentiable rendering, that overcomes the existing obstacles, mainly that of rasterization.

OpenDR [41] was the first such complete framework in Python for differentiable rendering, built upon an auto-differentiation framework. It supports per-pixel intensity with spherical harmonics or point light sources, triangular mesh representation and a pinhole camera model. For rasterization, they use a filtering kernel for all boundary pixels and a first-order Taylor expansion for all interior pixels. Neural 3D Mesh Renderer [42] introduces a rasterization approximation which enables differentiation for non-occluded gradients, using linear interpolation to replace the discontinuous triangle borders with gradual and smooth changes from background to foreground. This method can be directly incorporated into neural networks for rendering, but does not model pixel shading and cannot render meshes of different topologies.

For the first time, TF Mesh Renderer [43] used differentiable rendering to acquire a reconstruction loss of a 3DMM reconstruction from "in-the-wild" images. They introduced a differentiable rasterizer which interpolates per-vertex attributes in view space, using positive (inside triangle) and negative (outside triangle) barycentric coordinates to overcome discontinuity. Tf Mesh Renderer includes a deferred pixel shader which successfully incorporates Phong shading. SoftRasterizer [44] introduced a fully differentiable rendering framework using a probabilistic rasterizing function with an aggregation function for z-buffering, which significantly improves the gradient flow over OpenDR [41] and [42]. A different approach by Chen et al. [45] separate the rasterization for the foreground (in triangle) and background pixels (outside triangle) and uses re-defined differentiable barycentric coordinates to propagate the gradient only for the foreground pixels. The above methods can successfully use differentiable rendering and shading models in deep neural networks and can be used to solve inverse rendering problems. However, their rasterization approximation still lags behind traditional methods and the local illumination models cannot deal with occlusion and shadows.

Ray Tracing is a rendering technique which traces the path of light reflected from the light source to an object and then to the camera. Ray Tracing is more realistic to local illumination modeling and can accurately capture reflections and shadows. However, it comes with significant performance costs and the result is commonly sampled with Monte Carlo-based Ray Tracing, which is also discontinuous on object edges. Li et al. [46] introduced the first differentiable ray tracer, which uses a novel edge-sampling-based method to design a

continuous rendering function while using importance sampling to improve performance. Finally, Loubet et al. [47] introduced spherical rotations that remove the discontinuities w.r.t. the parameters of visibility, cameras, lights, and geometry. This method improves its robustness compared to [46] but is still suffering from biased gradients and high computational costs. However, the current methods have yet to produce realistic rendering of human faces.

Recently, several complete frameworks combined deep learning with differentiable rendering. TF Mesh Renderer [43] can be easily integrated with Tensorflow which also includes an integrated library for graphics and differentiable rendering [1]. Kaolin [48] is library of Pytorch implementations including [49] and SoftRasterizer [44]. Finally, Pytorch3D [50] is a complete modular differentiable rendering framework, loosely based on SoftRasterizer, with many additional modules for shading, performance and compatibility with data structures.

2.2.2 Shading Models

Shading refers to the computation of the visible color on a surface, and is based on the environment illumination and the surface's properties. Following the definitions by Pharr et al. 2016 [51], *Radiance* $L(p, \omega)$ is defined as the power radiated at a point $\mathbf{p}$, in a direction ω per unit of the projected area, perpendicular to that direction, per unit solid angle for a given frequency. When incident light hits a surface, part is reflected, and part is transmitted. The *Bidirectional Reflectance Distribution Function* (BRDF), $f_r(\mathbf{p}, \mathbf{v}, \mathbf{l})$, is a function describing the radiance coming from a surface at point $\mathbf{p}$ and viewing direction $\mathbf{v}$, when light is emitted at p from a direction $\mathbf{l}$. A *Bidirectional Transmittance Distribution Function* (BTDF), $f_t(\mathbf{p}, \mathbf{v}, \mathbf{l})$, is a function describing the transmitted light at a surface point $\mathbf{p}$ and viewing direction $\mathbf{v}$, with light emitted at $\mathbf{p}$ from a direction $\mathbf{v}$. To measure the total radiance at the point $\mathbf{p}_o$, we can integrate these functions over the sphere of incident directions $\mathcal{S}^2$:

$$L_o(\mathbf{p}_o, \mathbf{v}) = \int_{\mathcal{S}^2} f(\mathbf{p}_o, \mathbf{v}, \mathbf{l}) L_i(\mathbf{p}_o, \mathbf{l}) \, |\cos \theta_i| \, d\mathbf{l}, \tag{2.1}$$

[1] `www.tensorflow.org/graphics/overview`

where $L_i(\mathbf{p}_o, \mathbf{l})$ is the incident radiance at point $\mathbf{p}_o$ from direction $\mathbf{l}$, and $\mathbf{n}$ is the surface normal $\mathbf{p}_o$. As frequently used in literature, f denotes both the BRDF and BTDF, and is called the *Bidirectional Scattering Distribution Function* (BSDF).

Materials such as human skin exhibit significant subsurface scattering (SSS) [52]. The *Bidirectional Scattering Surface Reflectance Distribution Function* (BSSRDF) [51], $S(\mathbf{p}_o, \mathbf{v}, \mathbf{p}_i, \mathbf{l})$, describes the ratio of exitant differential radiance at a point $\mathbf{p}_o$ in direction $\mathbf{v}$, to the incident differential radiance at another point $\mathbf{p}_i$ from another direction $\mathbf{l}$. In that case, to measure the total radiance at point $\mathbf{p}_o$, we can integrate this function over the sphere of incident directions $\mathcal{S}^2$ and over the surface area A:

$$L_o(\mathbf{p}_o, \mathbf{v}) = \int_A \int_{\mathcal{S}^2} S(\mathbf{p}_o, \mathbf{v}, \mathbf{p}_i, \mathbf{l}) L_i(p_i, \mathbf{l})(\mathbf{n} \cdot \mathbf{l}) \, d\mathbf{l} \, dA. \tag{2.2}$$

Physically-based rendering, as defined above, considers the *Global Illumination* of a scene and measures the radiant incidence and exitance on each point from light sources and between objects in a scene, based on Light Transport Theory. The integrals can be estimated using Monte Carlo Estimators [53]. Although Global Illumination BSDFs achieve the most photorealistic rendering results, it is difficult to differentiate and slow to calculate. It is thus inefficient with iterative optimization and outside the scope of this work.

On the contrary, *Local Illumination* models measure the local shading at each surface point by simplifying certain light interactions. Multiple shading models have been suggested, that achieve realistic results and have low computational needs, even though they are not physically based. Moreover, they are typically fast and differentiable. Under local illumination, L_i is given, and a light source is defined as a distant area light with direction $\mathbf{l_i}$ and color $\mathbf{c_i}$ [40]. In that case, the BRDF is simplified to

$$L(\mathbf{p}_o, \mathbf{v}) = \pi \sum_{i=1}^{n} f(\mathbf{l_i}, \mathbf{v}) \mathbf{c_i} (\mathbf{n} \cdot \mathbf{l_i})^+, \tag{2.3}$$

where $()^+$ indicates clipping to positive numbers and n is the number of light sources.

A basic local shading model is the *Lambertian* [40], which calculates a view-independent reflectance value based on the light direction $\mathbf{l}$ and the surface normal $\mathbf{n}$, for a given albedo

color $\mathbf{a}$ at a surface point $\mathbf{p}_o$:

$$L(\mathbf{p}_o, \mathbf{v}) = \mathbf{a} \sum_{i=1}^{m} (\mathbf{l_i} \cdot \mathbf{n}) \mathbf{c_i} \tag{2.4}$$

Essentially, the above models the *diffuse* reflectance, but not the specular term, which models the refracted incident illumination that gets reflected back to the environment. According to the Fresnel equations [40], in most mediums and also in human skin, part of the incident illumination results in part diffuse reflectance, and part *specular* reflectance, where the light is reflected towards the reflection vector $\mathbf{r}$:

$$\mathbf{r} = 2(\mathbf{n} \cdot \mathbf{l})\mathbf{n} - \mathbf{l} \tag{2.5}$$

Usually, such an effect is affected by the mediums' refractive indexes. The Schlick approximation [54] for the Fresnel effect can be used to account for reflections with various refractive indexes.

Effective models of specular reflectivity account for micro-geometry, that is, surface details smaller than the pixel and the normal used in the calculations. Such micro-facet BRDF models [55, 56] assume a distribution of micro-facet surface normals and typically account for the shadowing and scattering effects such details cause at the specular reflection.

An early empirical model by Phong [57] captures both the specular and diffuse reflection and is plausible on facial skin and other dielectric materials. Under the Phong model, the reflection is composed of a diffuse (Lambertian), a specular, and an ambient component, for a set of m directional light sources, with $\mathbf{l_i}$ direction and $\mathbf{c}_{d_i}$ the diffuse, $\mathbf{c}_{s_i}$ the specular and $\mathbf{c}_a$ the ambient light intensities. The material properties are defined by the albedo, which defines the intensity of the reflected light for the diffuse component $\mathbf{a}_d$, specular component $\mathbf{a}_s$ and ambient component $\mathbf{a}_a$. For a specular roughness term $\alpha \in \mathbb{R}$ and a viewing direction $\mathbf{v}$, since specular reflection is view-dependent, the Phong model is defined as:

$$\mathbf{u} = \mathbf{c_a}\mathbf{a}_a + \sum_{i}^{m} \left(\mathbf{a}_d \left(\mathbf{l}_i \cdot \mathbf{n} \right) \mathbf{c}_{d_i} + \mathbf{a}_s \left(\mathbf{r_i} \cdot \mathbf{v} \right)^{\alpha} \mathbf{c}_d \right), \tag{2.6}$$

where $\mathbf{r}$ is the reflection vector defined by Eq. 2.5.

An extension to the Phong model, termed the Blinn-Phong model [58], modifies the re-

flection vector, as the half vector $\mathbf{h}$ between the view direction and light direction. The Blinn-Phong model assumes that the micro-facet BRDF is non-zero where the micro-facet normals are aligned with the vector pointing halfway between $\mathbf{l}$ and $\mathbf{v}$. The half vector is defined as:

$$\mathbf{h} = \frac{\mathbf{l} + \mathbf{v}}{||\mathbf{l} + \mathbf{v}||} \tag{2.7}$$

and the Blinn-Phong model is defined as:

$$\mathbf{u} = \mathbf{c_a a}_a + \sum_{i=1}^{m} \left(\mathbf{a}_d \left(\mathbf{l}_i \cdot \mathbf{n} \right) \mathbf{c}_{d_i} + \mathbf{a}_s \left(\mathbf{h_i} \cdot \mathbf{v} \right)^{\alpha} \mathbf{c}_d \right). \tag{2.8}$$

A multitude of other shading models has been designed, aiming at more realistic shading or modeling different material properties. Notable are the Cook-Torrance [56] and GGX [59], which enjoy widespread use up to today.

2.2.3 Facial Rendering Overview

There are multiple different setups that can accomplish photorealistic facial rendering. However, in the scope of this research, a certain paradigm is common, based on the requirements of photorealistic rendering, explicit mesh and texture parameterization, and differentiable rendering equations, which we expand below. Typically, the shape of the face is represented by a triangular mesh $\mathbf{S}$, with N vertices $S_i \in \mathbb{R}^3$ $i = 1, \ldots, N$ and a number of faces defining their connectivity. Each vertex can be assigned a color $S_{i_c} \in \mathbb{R}^3$ (*colored mesh*), or a set of UV coordinates $S_{i_{uv}} \in \mathbb{R}^2$ (*textured mesh*), which map each vertex to a position in a texture map $\mathbf{T} \in \mathbb{R}^{W \times H \times 3}$, where W, H are the dimensions of the texture map. When rendering a textured mesh, multiple pixels per face can be interpolated on the rasterized image and provide a higher-resolution rendering, than when interpolating between the colored vertices.

Many facial reconstruction works rely on either a colored (e.g. [6, 10]) or textured mesh (e.g. [19, 60]), which are not adequate for photorealistic facial rendering, and often contain baked environment illumination from the training data. Most importantly, for facial relighting, the facial texture must be an *albedo*, representing only the diffuse response of the skin and devoid of any external illumination. Moreover, for a dielectric material such



$\mathbf{A}_D$ $\mathbf{A}_S$ $\mathbf{N}_D$ $\mathbf{N}_S$ $\mathbf{S}$ mesh render

Figure 2.1: Example set of facial rendering assets (diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ diffuse normals $\mathbf{N}_D$, specular normals $\mathbf{N}_S$) as defined in this chapter, including a mesh with rasterized surface and a rendering.

as skin, both diffuse and specular terms are required to produce photorealistic results [61]. The specular term can also be defined by a *specular albedo*, which defines the skin's spatially varying specular response. Depending on the shading model used, a constant or spatially varying *specular roughness* term may be required, which defines the microfacet distribution.

Meshes of high polygonal count can model facial details accurately, but become increasingly expensive to render and manipulate. Therefore, it is common practice to reduce the number of vertices in the mesh, and create a texture map with normal information $\mathbf{N}$. Moreover, [4, 5, 62] have shown that skin exhibits different intensity for the specular and diffuse component and that due to the skin's subsurface scattering, detailed surface normals can be used to model the specular term, while much smoother diffuse normals should be used to model the diffusion.

In the following chapters, we consider two types of facial assets: A) the simpler combination of a triangular mesh $\mathbf{S}$ and a single texture $\mathbf{T}$, and B) the photorealistically renderable assets consisting of a triangular mesh $\mathbf{S}$ and a set of textures $\mathbf{R} = \{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_D, \mathbf{N}_S\}$, where $\mathbf{A}_D$ is the diffuse albedo, $\mathbf{A}_S$ is the specular albedo, $\mathbf{N}_D$ are the diffuse normals and $\mathbf{N}_S$ the specular normals. Of note, in some works, the diffuse and specular normals $\mathbf{N}_D$ may not be explicitly defined, as surface normals $\mathbf{N}$ can be computed from $\mathbf{S}$.

Finally, on a different paradigm, neural rendering approaches are also becoming increasingly popular for facial and human rendering. Neural rendering for facial assets [63] typically uses a mesh and neural material representations. Such materials can be rendered using an optimizable neural network, conditioned on the assets and in a specific scene. Such approaches achieve excellent rendering quality from even incomplete facial assets, as the neural network learns to correct rendering imperfections, but require huge training

data and lengthy training, while being challenging to incorporate into typical rendering applications.

Recently, Neural Radiance Fields (NERFs) [64], proposed a set of MLPs that learn a radiance field of a static scene, that can be rendered using volumetric Rendering [64]. NERFs require tens to hundreds of 2D images of a static scene for training and suffer from the drawbacks mentioned above of neural rendering, additionally to slow rendering times. However, multiple follow-up works add faster rendering [65], dynamic scenes [66], and controllable illumination [67]. Finally, recent works [68] combine powerful generative models with NERFs with applications on facial modeling and rendering.

2.3 Facial Geometry and Reflectance Capturing

Facial geometry and reflectance acquisition is a widely explored field in the literature, and can be accomplished in different ways, depending on the hardware available and the required precision. Facial geometry (Sec. 2.3) is usually represented as a point cloud or meshes, and can be acquired from multi-view images or using structured light (e.g. laser scanners) (Sec. 2.3.1). Facial reflectance acquisition, can be acquired with active illumination devices (Sec. 2.3.2) or passive illumination devices (Sec. 2.3.3). Finally, this chapter also explores relevant smaller-scale scanners and handheld devices (Sec. 2.3.4) and non-facial capturing (Sec. 2.3.5).

2.3.1 Geometry Acquisition

Structured Light methods employ structured signals, which are projected on the subject under a controlled environment. Such methods include devices that utilize time-of-flight or structured-light systems, which provide high-accuracy (i.e. laser scanners) or easy-of-use (i.e. mobile phone sensors). An example of a laser scanner usable for facial reconstruction is Arctec3D[2]. Moreover, time-of-flight sensors with structured light projections, provide rough shape estimation combined with ease of use, examples of which are Kinect[3] and ARkit[4]. A thorough analysis of these methods is beyond the scope of this work.

[2] www.artec3d.com/
[3] developer.microsoft.com/en-us/windows/kinect
[4] developer.apple.com/augmented-reality/

Passive capturing systems also provide an easy way of acquiring geometry, as they only require several photographs of the subject under unconstrained illumination. Multiple commercial stereo-reconstruction frameworks (i.e. Agisoft Metashape[5], Reality Capture[6]) and the open-source (i.e. COLMAP[7] and Meshroom[8]) can be used. It's important to note that passive capturing methods usually result in rough and noisy estimations. Geometry-enhancing techniques can mediate this, such as Nehab's [69], by calculating the displacement map (also known as a height map) from captured normals and using it to emboss the details on the geometry.

Finally, 3D Morphable Models are powerful statistical models widely used for facial shape modeling and inference and are covered in detail in Sec. 5.2.2. Li et al. [31] compare Arctec3D, structure-from-motion, and 3DMM fitting on facial reconstruction.

2.3.2 Active Illumination Facial Scanners

Photometric stereo systems have traditionally been popular in computer vision for estimating surface normals, in conjunction with color-multiplexed illumination for dynamic capture [70][71]. Debevec et al. [72] first proposed a specialized light stage setup to acquire a dense reflectance field of a human face for photo-realistic image-based relighting applications. They also fit microfacet-BRDF based reflectance using the acquired data. Weyrich et al. [73] instead employed an LED sphere and multiple cameras to densely record facial reflectance and computed view-independent estimates of spatially varying facial reflectance from the acquired data for driving a microfacet BRDF model-based rendering. Subsequently, Ma et al. [74] introduced polarized spherical gradient illumination (using an LED sphere) for efficient acquisition of the separated diffuse and specular albedos and photometric normals of a face, and demonstrated high-quality facial geometry, including skin mesostructure as well as realistic rendering with the acquired data.

Ghosh et al. [75] further extended the acquisition method to acquire layered facial reflectance using a combination of polarization and structured lighting and further extended the view-dependent solution of Ma et al. for multi-view facial acquisition with polarized

[5] `www.agisoft.com/` previously called Agisoft Photoscan
[6] `www.capturingreality.com`
[7] `colmap.github.io`
[8] `www.alicevision.org/`

spherical gradient illumination [76]. These techniques have had a significant impact on film VFX. Fyffe et al. [77] employed the method of [78] for temporal alignment of spherical gradient illumination for dynamic performance capture. However, their method involves a heuristic-based separation of reflectance, which is inaccurate, and the capture process requires high-speed cameras. Fyffe & Debevec [79] further proposed a single-shot method using a complex setup of a polarized RGB LED sphere for color-multiplexed spherical gradients that are further polarized using the method of [76] for diffuse-specular separation. Finally, Kampouris et al. [80] employed binary spherical gradient illumination using an LED sphere for facial appearance capture including diffuse-specular separation of albedo and photometric normals.

An alternate approach that does not require an LED sphere is the work of Fyffe et al. [81] on static facial appearance capture employing off-the-shelf photography hardware. However, the approach does not extend to dynamic facial appearance capture. Still, it requires quite a complex hardware setup consisting of 24 cameras and six flashes triggered in sequence within a few milliseconds. These previous works inspire our approach.

2.3.3 Passive Illumination Facial Scanners

Researchers have also investigated approaches for passive acquisition of faces and entire body. Such acquisition is particularly well suited for performance capture since active approaches usually require time-multiplexed illumination, imposing requirements of high frame rate acquisition, lighting control and synchronization. A popular approach has been to employ uniform constant illumination for multi-view facial capture [82][83]. Such an approach enables the estimation of an albedo texture under flat-lit illumination for rendering purposes besides facial geometry reconstruction based on the multi-view stereo. High-frequency mesostructure is further embossed on the base-geometry using a high-pass filter on the diffuse texture [82]. The approach has been extended for reconstructing facial performances with drift-free tracking over long sequences using anchor frames [84]. While producing excellent qualitative results for facial geometry, the estimated albedo is not completely diffuse and contains a small amount of specular reflectance baked into the texture, and the embossed mesostructure is plausible rather than accurate.

More recent works have extended passive capture systems for achieving facial reflectance estimation using inverse rendering in conjunction with view-multiplexing [85][86]. The latter demonstrates a more practical and inexpensive apparatus. However, while achieving impressive results for passive capture, the quality of acquired reflectance maps is not quite at par with active illumination systems. Gotardo et al. [85] are also able to acquire dynamic facial appearance, however, their method requires the initialization of a neutral facial appearance through dense multi-view capture in conjunction with measurement of person-specific change in skin color due to blood flow constraining the optimization under passive illumination.

2.3.4 Smaller Scale Systems

Several systems have been suggested for 3D reconstruction from hand-held scanners. Such methods provide excellent portability and enable the acquisition of vast datasets, however, the existing methods only reconstruct medium-quality geometry and do not capture reflectance. Bellus3D (now discontinued) used a propertiary camera system to capture the subject's face from different viewpoints with RGB and depth and estimate a colored facial mesh. However, they were limited to facial and head reconstructions and did not capture skin reflectance but a texture with baked illumination. Scandy Pro uses a similar approach, but capture with a TrueDepth-equipped iPhone [9] and still suffers from the aforementioned drawbacks. Therefore, other types of scanners that do not capture photorealistic appearance, like laser scanners, are not in the scope of this study. Nevertheless, the subject of handheld facial scanning is of great interest, because the ability to capture photorealistic facial meshes using a cheap and portable device would significantly increase the amount of data available for learning-based methods. On this matter, Ichim et al. [87] demonstrate a combination of photometric-stereo from an arbitrary mobile phone camera video input, which they combine with a statistical model to easily acquire medium-quality facial scans.

[9] TrueDepth is a camera system by Apple that captures RGB, depth, and infrared data. Additional technical details are available at `https://support.apple.com/en-gb/HT208108`

2.3.5 Non-skin capturing

All previously described work focus on facial skin acquisition, since other facial features require different approaches. Human eyes exhibit high specularity and are usually partially occluded in scans. The state-of-the-art method on eye acquisition [88] can capture the individuality of each eye and realistically render it. The sclera is captured using a passive multi-view system, the cornea is reconstructed using a photometric LED system and the iris is captured by a path tracing multi-view system.

The teeth and tongue also exhibit different material properties than the skin and are also primarily occluded in facial photographs. Dentistry teeth scanners with structured lighting (i.e. iTero [10] or 3Shape [11]) are specialized scanners that have to be inserted orally to capture. Wu et al. [89] reconstruct teeth from normal photographs, using a parametric teeth model created from 3D teeth models acquired with plaster molds. Recently, Velinov et al. [90] implicitly model the reflectance of teeth and capture the surface reflectance, transmission, and sub-surface scattering, using intra-oral scanning and path tracing.

Finally, hair is another particular category in reconstruction, as it exhibits different reflectance properties and is sub-millimeter in width, which complicates representation. Early volumetric approaches with stereo reconstruction fail to capture both the shape and reflectivity of the hair. Beeler et al. [91] jointly reconstruct skin and facial hair, by reconstructing bulky facial hair using stereo reconstruction and adding synthesized facial fibers where detected on the skin. Luo et al. [92] use unconstrained images of hair to capture plausible hair 3D orientation fields whose occluded areas are completed to form full-length colored wisps that can be fitted on a head model. Hu et al. [93] reconstruct hair from single image by simulating possible hairstyles and then picking example hair strands from the best simulation. Hu et al. [94] also suggest curating a dataset of artist-made hairstyles, which are then matched to an input image and a few example hair strokes.

[10] www.itero.com
[11] www.3Shape.com

2.4 Deep Learning Methods for Image Synthesis and Manipulation

Deep learning algorithms model high-level abstractions in data, drawing ideas from the biological nervous system of interconnected neurons. The norm in training such systems from data is Backpropagation [95], which describes the adjustment of the neuron's weights, by passing along the network, the gradient of a loss function. The seminal of Convolutional Neural Networks (CNNs) [96, 97] introduced weight sharing between nodes and learnable feature maps on 2D data, which enabled object recognition in images. Kingma et al. [98] introduced the Variational AutoEncoder, a network that learns the approximate latent distribution of input data, by compressing and reconstructing them using CNNs, which highlighted the potential of Deep Learning in image generation. Even though their results were impressive, they were still low in quality and resolution for any meaningful application. During the last few years, Generative Adversarial Networks (GANs) [13] truly unlocked the power of image generation, by implicitly modeling the distribution of given data and reconstructing realistic unseen images, using a pair of CNNs in a game-theoretic setting, as described in the following section.

2.4.1 Generative Adversarial Networks for Image Synthesis

GANs [13] are neural network models, that learn to implicitly estimate the probability distribution of training data by using two differentiable components, the generator and the discriminator. The generator $G(\mathbf{z}|\theta_\mathbf{G})$ with parameters $\theta_\mathbf{G}$ produces a sample from the approximated distribution of the model p_{model}, given a prior $\mathbf{z}$. The discriminator $D(\mathbf{X}|\theta_\mathbf{D})$ with parameters $\theta_\mathbf{D}$ predicts if a sample $\mathbf{x}$ comes from the actual data distribution p_{data} or from the generator's learned distribution p_{model}. Both agents are jointly trained with the target of finding the parameters $\theta_\mathbf{G}$ and $\theta_\mathbf{D}$, such as the generator's $p_{model} = p_{data}$. Therefore, in the value function $V(\theta_\mathbf{D}, \theta_\mathbf{G})$, we aim to maximize the discriminator's ability to recognize samples from p_{data}, and to minimize its ability to recognize them, when they are coming from the generating agent:

$$\min_G \max_D V(\theta_\mathbf{D}, \theta_\mathbf{G}) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_\mathbf{z}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \tag{2.9}$$

The Kullback-Leibler (KL) divergence measures the divergence of a distribution P from another distribution Q:

$$KL(P||Q) = -\int_{\mathbf{X}} \log \frac{P}{Q} dP \tag{2.10}$$

Using the KL divergence, the Jenshen-Shannon divergence measures the similarity between two distributions as:

$$JD(P||Q) = \frac{1}{2} KL\left(P \middle|\middle| \frac{P+Q}{2}\right) - \frac{1}{2} KL\left(Q \middle|\middle| \frac{P+Q}{2}\right) \tag{2.11}$$

Then, the cost for the discriminator $J_D(\theta_{\mathbf{D}}, \theta_{\mathbf{G}})$ is derived from the value function as follows. Minimizing it allows to find the p_{model} that is similar to p_{data}:

$$J_D(\theta_{\mathbf{D}}, \theta_{\mathbf{G}}) = \frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \tag{2.12}$$

Training both agents resembles a zero-sum game-theoretic setting [13], meaning that the sum of both player's cost is zero:

$$J_D + J_G = 0 \tag{2.13}$$

Then the value function V (Eq. 2.9) can be used to devise the optimal parameters for the generator as:

$$\theta^*_{\mathbf{G}} = \arg\min_{\theta_{\mathbf{G}}} \max_{\theta_{\mathbf{D}}} V(\theta_{\mathbf{D}}, \theta_{\mathbf{G}}) \tag{2.14}$$

However, in practice the heuristic approach by [99] is used for the generator's cost J_G, which flips the target on the discriminator's cost, thus maximizing its probability of being mistaken:

$$J_G = -\frac{1}{2}\mathbb{E}_{\mathbf{z}} \log D(G(\mathbf{z})) \tag{2.15}$$

The Deep Convolutional GAN (DCGAN) [100] introduced several variations from the original model, that made training more stable and reached higher quality results. By using fully convolutional networks with strided convolutions, DCGAN achieves higher-quality results. Moreover, eliminating the fully connected layers and connecting the convolutional layers directly to the vectors, enhanced the stability of the network while training. Finally, by normalizing each batch to zero mean and dividing it by the batch's standard deviation (*Batch Normalization* [101]), was shown to speed-up learning and help avoid mode

collapse.

The Wasserstein GAN (WGAN) [102] improved on GAN training stability by altering the distance measure between p_{data} and p_{model}. Instead of the Jensen-Shannon divergence (Eq 2.9), the distance between the distributions is defined in a limited weight space $\mathbf{w} \in [-0.01, 0.01]$ to ensure continuity:

$$W(p_{data}, p_{model}) = \inf_{\gamma \in \Pi(p_{data}, p_{model})} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\gamma} \left[||\mathbf{x} - \mathbf{y}||\right] \tag{2.16}$$

Limiting WGAN weight space results in exploring more straightforward functions and thus limiting the network's capacity. Gulrajani et al. [103] introduced the WGAN with Gradient Penalty (WGAN-GP), which penalizes the gradient of the discriminator. The objective then is defined as:

$$\max_G \min_D \mathbb{E}_{\mathbf{x}\sim p_{model}}[D(G(\mathbf{z}))] - \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}[D(\mathbf{x})] - \lambda \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}} \left[\left(||\nabla_{\mathbf{x}} D(\mathbf{x})||_2 - 1 \right)^2 \right] \tag{2.17}$$

All previous methods showcased sharp and realistic results, but were constrained by low-resolution outputs (up to 256×256). Karras et a. [14] introduced a progressive GAN architecture (PROG-GAN) that enabled generating realistic images of up to 1024×1024 pixels, while achieving stable training in a short training time. The network starts with a 4×4 input to fully convolutional layers, and after training to an equilibrium, adds an extra convolutional layer with double the size of the previous one. The process continues until the maximum supported network size, and all weights remain trainable until the end. They also introduce a layer for increased standard deviation between features, to increase the output's variability and replace batch normalization with dynamic weight scaling.

2.4.2 Generative Adversarial Networks for Image Manipulation

Image translation refers to the adaptation of an input image to a different domain (i.e. artistic style or illumination conditions). The traditional GAN value function V (Eq. 2.9) can be conditioned to a label so that the generated data correspond to a specific domain.

Mirza and Osinero [104] first proposed the conditional GANs, where both the generator

and the discriminator are conditioned on class labels $\mathbf{y}$. The objective of the GAN is defined as:

$$\min_G \max_D V(\theta_{\mathbf{D}}, \theta_{\mathbf{G}}) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})} [\log D(\mathbf{x}|\mathbf{y})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z}|\mathbf{y})))] \quad (2.18)$$

Following the conditional GAN paradigm, Isola et al. [105] introduced *pix2pix*, which learns to translate images between two domains (i.e. sketch and photograph or day and night). Instead of conditioning both agents on a label $\mathbf{y}$, the generator G is conditioned on the source domain input $\mathbf{x}$. Then discriminator is now assessing the validity of the pair of $\mathbf{x}$ and either the generators output or the actual representation of $\mathbf{x}$ in the *target* domain, $\mathbf{y}$:

$$L_{Adv}(G, D) = \mathbb{E}_{\mathbf{x},\mathbf{y}} [\log D(\mathbf{x}, \mathbf{y})] + \mathbb{E}_{\mathbf{x},\mathbf{z}} [\log (1 - D(\mathbf{x}, G(\mathbf{x}, \mathbf{z})))] \quad (2.19)$$

The generator's architecture is an encoder-decoder network (*U-Net*), which comprises convolutional layers that downsample the input into a lower-dimensional space before generating the output from it, while enabling information to pass between the encoder and decoder layers of the same size (*skip-connections*). Moreover, the translation is improved when the generator is forced to produce a result near the ground truth of the target domain image, which can be achieved with an additional L_1 loss between them:

$$L_{L1}(G) = \mathbb{E}_{\mathbf{x},\mathbf{y},\mathbf{z}} [||\mathbf{y} - G(\mathbf{x}, \mathbf{z})||_1] \quad (2.20)$$

The complete objective comprises both losses:

$$G^* = \arg \min_G \max_D L_{Adv}(G, D) + \lambda L_{L1}(G) \quad (2.21)$$

The main drawback of the pix2pix framework is the need for a vast dataset of paired images from both domains. *CycleGAN* [106] addresses this need, by enabling image translation between 2 domains $(\mathbf{Y}, \mathbf{X})$, using unpaired training data. The network comprises two mappings for the generator $G : \mathbf{X} \rightarrow \mathbf{Y}$, $F : \mathbf{Y} \rightarrow \mathbf{X}$ and their respective discriminators D_Y, D_X. Then, the adversarial loss for the network is defined as:

$$L_{GAN}(G, D_Y, \mathbf{X}, \mathbf{Y}) = \mathbb{E}_{\mathbf{y}\sim p_{data}(\mathbf{y})} [\log D_Y(\mathbf{y})] + \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})} [\log (1 - D_Y(G(\mathbf{x})))] \quad (2.22)$$

$$L_{GAN}(G, D_X, \mathbf{X}, \mathbf{Y}) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\log D_X(\mathbf{x})\right] + \mathbb{E}_{\mathbf{y}\sim p_{data}(\mathbf{y})}\left[\log\left(1 - D_X(G(\mathbf{y}))\right)\right] \quad (2.23)$$

CycleGAN learns to translate from one to the other domain and vice versa, which not only increases utility, but constrains each generator to produce results, that that can be translated back to the source domain, using the second generator. To do so, Zhu et al. [106] suggest the addition of another loss, L_{cyc}, which measures the distance between the input image $\mathbf{x}, \mathbf{y}$ and the twice-translated result, $F(G(\mathbf{x})), G(F(\mathbf{y}))$ for both mappings:

$$L_{cyc} = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[||F(G(\mathbf{x})) - \mathbf{x}||_1\right] + \mathbb{E}_{\mathbf{y}\sim p_{data}(\mathbf{y})}\left[||G(F(\mathbf{y})) - \mathbf{y}||_1\right] \quad (2.24)$$

Then, the total loss L is defined as:

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, \mathbf{X}, \mathbf{Y}) + L_{GAN}(G, D_X, \mathbf{Y}, \mathbf{X}) + \lambda L_{cyc}(G, F) \quad (2.25)$$

Both pix2pixHD and CycleGAN can only be trained to translate from one domain to another for each generator. Choi et al. [107] introduced the Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation (STARGAN). It features a single conditional GAN generator $G(\mathbf{x}, \mathbf{c})$, that can translate an input image $\mathbf{x}$ between multiple domains, given a domain label c. Here, the discriminator features an adversarial loss D_{adv} (similarly to (Eq. 2.19) and an additional auxiliary classifier, which judges whether the given image $\mathbf{x}$ belongs in the given domain c). The *domain classification loss* is defined on both the real (Eq. 2.26) and fake (Eq. 2.27) images.

$$L_{label}^{real} = \mathbb{E}_{\mathbf{x},c}\left[-\log D_{label}(c|\mathbf{x})\right] \quad (2.26)$$

$$L_{label}^{fake} = \mathbb{E}_{\mathbf{x},c}\left[-\log D_{label}(c|G(\mathbf{x}, c))\right] \quad (2.27)$$

Moreover, Choi et al.[107] use the cycle consistency loss from CycleGAN [106], so that the translated to a domain c image, can be translated back to its original domain c':

$$L_{cycle} = \mathbb{E}_{\mathbf{x},c,c'}\left[\left|\left|\mathbf{x} - G\left(G(\mathbf{x}, c), c'\right)\right|\right|_1\right] \quad (2.28)$$

Finally, the losses for training STARGAN are defined as:

$$L_D = -L_{adv} + \lambda_{label} L_{label}^{real}, \quad L_G = L_{adv} + \lambda_{label} L_{label}^{fake} + \lambda_{cycle} L_{cycle} \quad (2.29)$$

All aforementioned works have been groundbreaking but were still constrained in mid-low resolution data, i.e. 256×256 for pix2pix [108]. The seminal work by Wang et al. [18], pushed the boundaries in image translation by generating realistic images of 2048×1024 pixels from semantic labels. Based on the architecture of pix2pix [108], a 2D input of semantic labels per pixel (i.e. car, road or sky) is concatenated with a 2D per pixel boundary (1 on the boundary between objects, 0 elsewhere) $\mathbf{s} : [\mathbf{S}_{label}, \mathbf{s}_{boundary}]$ and passed on the conditional generator G. The generator is comprised of 2 components: the *global generator* G_1, which performs the actual translation between the two domains at a down-sampled version of the input and labels, and the *local generator-enhancer* G_2, which down-samples the input before passing it to G_1 and then up-samples and enhances the details of the output from G_2. Both consist of a convolutional front-end (denoted with F), a set of residual blocks (R), and a transposed convolutional back-end (B). This architecture enables both high-resolution results and stable training. Wang et al. use a set of 3 discriminators D_1, D_2, D_3, which have the same architecture but work on the image in different scales $1\times, \frac{1}{2}\times, \frac{1}{4}\times$. This enables the network to optimize both, with a global view of the result, while also picking up the fine details of the image. Finally, they add an additional loss to the learning problem, termed *feature matching loss*, which stabilizes the network and guides the generator to meaningful statistics, by penalizing the difference of multiple layers of the discriminator, between the real and fake image:

$$L_{FM}(G, D_k) = \mathbb{E}_{(\mathbf{s},\mathbf{x})} \sum_{i=1}^{T} \frac{1}{N_i} \left[\left|\left| D_k^{(i)}(\mathbf{s}, \mathbf{x}) - D_k^{(i)}(\mathbf{s}, G(\mathbf{s})) \right|\right|_1 \right] \tag{2.30}$$

Then, the objective for pix2pixHD is:

$$\min_G \left(\left(\max_{D_1,D_2,D_3} \sum_{k=1,2,3} \mathcal{L}_{GAN}(G, D_k) \right) + \lambda \sum_{k=1,2,3} \mathcal{L}_{FM}(G, D_k) \right) \tag{2.31}$$

2.5 3D Facial Synthesis and Reconstruction

2.5.1 3D Facial Geometry and Texture Modeling and Synthesis

Statistical Methods

Over the years, numerous methods have been introduced in the literature that tackle the problem of 3D facial reconstruction from a single input image. Early methods required a statistical 3D Morphable Model (3DMM) [6, 9] both for shape and appearance, usually encoded in a low-dimensional space constructed by PCA. A thorough review of such methods is compiled by Egger et al. [109].

The 3DMM is defined as a generator function $c : \mathbb{R}^d \rightarrow \mathbb{R}^{3n}$, for a d-dimensional parameter space to a mesh of n vertices, where $d << n$. Blanz and Vetter [6] in their seminal work, first used PCA to disentangle a number of captured 3D facial meshes to their lower-dimensional representation w.r.t. identity, expression, and appearance.

Identity: Before training, all 3D shapes should be in correspondence, so that the shape $\mathbf{s} = \left[\mathbf{x}_1^\top, \ldots, \mathbf{x}_N^\top\right]^\top = [x_1, y_1, z_1, \ldots, x_N, y_N, z_N]^\top$. Blanz and Vetter [6] calculate the mean of the training shapes $\bar{\mathbf{c}}$ and the matrix with the d dominant eigenvectors of the covariance $\mathbf{E}^s$ of the difference of each shape from the mean. The training data have been captured with neutral expression and pose and have been densely aligned so that PCA will only capture identity variation. Then, for an instance of the low-parameter vector $\mathbf{w}^s \in \mathbb{R}^d$ a new shape can be generated as:

$$c(\mathbf{w}) = \bar{\mathbf{c}} + \mathbf{E}^s \mathbf{w}^s \tag{2.32}$$

Additionally, given an existing shape $\mathbf{c}_0$ in the same topology and in dense correspondence with the model, the optimal shape vector that produces such a shape can be recovered as:

$$\mathbf{w} = \mathbf{E}^{s\top}(\mathbf{c}_0 - \bar{\mathbf{c}}) \tag{2.33}$$

Expression: Blanz et al. [6] also model expressions using PCA on reconstructions with different expressions. Here, the dominant eigenvectors for expression $\mathbf{E}^e$ are found on the covariance of the difference between a reconstruction with expression and the neutral

shape of the same subject. The eigenvectors for identity $\mathbf{E}^s$ are calculated as in Eq. 2.32, using only the reconstructions with a neutral expression. Thus, to generate a shape with expression, both models are combined:

$$c(\mathbf{w^s}, \mathbf{w^e}) = \bar{\mathbf{c}} + \mathbf{E^s w^s} + \mathbf{E^e w^e} \tag{2.34}$$

Multiple variations of this framework have been suggested. Given pre-defined expression blendshapes, few examples of expression face scans can be used to adapt the blendshapes to that subject [110]. Vlasic et al. [111] introduced a 3DMM based on a multi-linear model using a higher-order tensor decomposition, which models the correlation between identity and expression. Recent work has also shown that a similar approach can be used to drive speech instead of expression [26].

Appearance: 3DMM can also capture variations in appearance as well, either as per-vertex color values or 2D texture maps. Blanz and Vetter [6] unwrap cylindrically the appearance of the reconstruction in a two-dimensional texture representation (UV space) which is mapped to the vertices using texture-coordinates $tc : \mathbb{R}^{3n} \rightarrow \mathbb{R}^{u \times v}$. Again, using PCA on the covariance of the difference of each texture and the mean $\bar{\mathbf{T}}$, the generative model $t : \mathbb{R}^d \rightarrow \mathbb{R}^{u \times v}$ is defined as:

$$t(\mathbf{w}^t) = \bar{\mathbf{t}} + \mathbf{E}^t \mathbf{w}^t \tag{2.35}$$

Appearance can also be modeled in the same dimensions as the vertices (RGB instead of XYZ) with $tc : \mathbb{R}^{3n} \rightarrow \mathbb{R}^{3n}$. Booth et al. [9, 10] build such a model from SIFT features, which are invariant to illumination changes. However, working on UV space grants several benefits, such as compatibility with CNNs and the ability to work with a geometry of a different resolution (usually lower). Booth and Zafeiriou [112] offer an analysis on the subject. On another note, Moschoglou et al. [113] propose a robust matrix factorization for facial uv-maps that can handle missing information and outliers. Finally, the linear biophysical 3DMM [114] models appearance based on melanin and hemoglobin pigments.

The above framework is defined as a *global model*, as the full mesh is generated altogether, which is straightforward to model, but may not reproduce high-frequency details. Different

local model versions have been suggested [115, 116] which segment the face into different parts and model them separately, which improves reconstruction fidelity.

The framework introduced by Blanz and Vetter [6] has evolved and most current state-of-the-art 3DMMs are based on it, i.e. the Basel Facial Model (BFM) [117] and the Large-Scale Facial Model (LSFM) [9]). Recently, variations on the 3DMM have been introduced, that improve the identity, expression, or appearance modeling, their accuracy and generalization. The DL-3DMM [118] is using a learned dictionary of bases for the linear model, which improves the reconstruction error for different poses and ethnicities. Wu et al. [116] use a bone-structure local PCA model that accurately captures extreme motion deformations. FLAME [119] combines bone articulation with the above linear model, effectively making it a non-linear model with more realistic expressions. More non-linear approaches based on deep learning are described in Section 2.5.2.

All previously described methods are focused on the facial area, since it contains most identity-specific characteristics and details. A prominent work focusing on details whole-head 3DMM is the *Liverpool-York* head model [120], which uses over 1200 subjects and transfers the texture directly from the input captures for higher-resolution texturing. Moreover, Ploumpis et al. [121] introduced the *Unified Head Model*, a 3DMM of head, eyes, teeth, oral cavity, and ears. Instead of combining all components in a single PCA model, they introduce a Gaussian Process method for fusing them together.

2.5.2 3D Facial Geometry and Texture Reconstruction from a Single Image

Fitting Methodology

The most prominent application of a 3DMM is its fitting to an arbitrary 2D image $\mathbf{I}_0$ of a human face. The standard process, originally introduced by Blanz and Vetter [6] uses an *analysis by synthesis* approach, where the 3DMM recreates a natural image that is compared against the input. Specifically, the statistical shape is projected to 2D-space, in order to match the pose of the input face and the texture is rendered in the input's illumination environment to produce the synthesized face $\mathbf{I}(\mathbf{w})$. For the fitting, the unknown parameters $\mathbf{p}$ are the weights for the 3DMM model $\mathbf{w^s}, \mathbf{w^e}, \mathbf{w^t}$ and the rendering

parameters for the camera $\mathbf{w^c}$ and the illumination $\mathbf{w^l}$ such as: $\mathbf{w} = \left[\mathbf{w}^s, \mathbf{w}^e, \mathbf{w}^t, \mathbf{w^c}, \mathbf{w^l}\right]$. Then, an energy-minimization method is used for the optimal parameters. A regularizer $Reg(\mathbf{w})$ may also be used for the parameters:

$$\min_{\mathbf{w}} \mathcal{E}(\mathbf{w}) = ||\mathbf{I}_0 - \mathbf{I}(\mathbf{w})||_2^2 + Reg(\mathbf{w}) \tag{2.36}$$

More recent approaches are still based on this framework [9, 122]. Fitting can also be based on landmarks, curves and contours [118]. Another approach is optimizing in feature space and not RGB space [9, 123].

Landmark Localization

Facial landmarks can be automatically localized in a 2D image, after a face has been detected. The standard methodology revolves around fitting an Active Appearance Model (AAM) [124]. AAMs are generative deformable statistical models that can be used to bring an object into an image into registration with a deformed reference template of shape and appearance. In the case of landmark localization, the AAM is built with a set of landmarks, i.e. iBug Facial Point Annotations [125]. Instead of the input RGB image, a more recent robust and accurate AAM is applied on a dense Histogram of Oriented Gradients (HOG) of the input image [8], using a gradient descent-based optimization. RetinaFace [126], a recent work, uses a fully convolutional network to extract features for a multi-task regression, that in one pass jointly localizes a bounding box, 2D facial landmarks and a 3D mesh for the input image.

Registration

Another form of fitting, termed *registration*, describes the transformation of a given 3D mesh to the topology of the 3DMM, bringing them in *dense correspondence*. This means that they will have the same number of vertices, same triangulation and same semantic and anatomic meaning for all vertices. It is common also to transform the input, so that it has the same center, scale, and pose as the 3DMM mean.

The seminal work on registration is the Non-Rigid Iterative Closest Point algorithm [127]

which describes an incremental deformation of a template source to a target mesh. For a set of per-vertex stiffness weights α, Amberg et al. [127], find the nearest correspondence for each vertex and calculate a deformation, in a loop, until a multi-target error is minimized. For a template $\mathcal{S} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V}$ and edges $\mathcal{E}$, and a target mesh $\mathbf{X}$, the error E is comprised of a) a *distance term*: between the source $\mathbf{X_i u_i}$ and the target nearest vertices $\mathbf{T}$, weighted by the match reliability w_i, b) the *stiffness term*: that regularizes the deformation by penalizing the difference of neighboring transformations, using a weighting matrix $\mathbf{G}$ and c) the *landmark term*: which regularizes the registration based on predefined landmark points $\mathbf{l}$. In each loop, both stiffness and landmarks terms are assigned decreasing weights, α_i and β_i:

$$E(\mathbf{X}) = \sum_{\mathbf{u_i} \in V} w_i dist^2(\mathbf{T}, \mathbf{X_i u_i}) + \alpha_i \sum_{\{i,j\} \in \mathcal{E}} ||(\mathbf{X_i} - \mathbf{X_j})\mathbf{G}||_F^2 + \beta_i \sum_{(\mathbf{u_i}), \mathbf{l} \in \mathcal{L}} ||\mathbf{X_i u_i} - \mathbf{l}||^2 \quad (2.37)$$

A more recent state-of-the-art approach [128] achieves landmark-free registration, which regularizes the deformation, based on the entropy levels on different facial areas. Finally, a state-of-the-art flexible commercial approach is offered by Wrap [12].

GAN-based Methods:

Lately, many approaches have tried to leverage the power of Deep Learning to either regress the latent parameters of a PCA model or utilize a 3DMM to synthesize images and formulate an image-to-image translation problem using CNNs. Tran and Liu [129] introduced a non-linear 3DMM which uses a set of CNN networks to infer the model's shape and albedo parameters, as well as the illumination, pose, and camera parameters, which is optimized using a differentiable rendering layer that reconstructs the input image from the fitting. Cole et al. [130] use a differentiable image warping technique to regenerate an input image in a registered 2D space, from which an instance of a 3DMM can be derived.

Deep learning based 3DMM reconstructions have also been able to alleviate the need for large labeled training datasets, by combining a *differentiable renderer* with self-supervised training using arbitrary images [129, 131, 132, 133] and video [134] for training. Moreover, Genova et al. [43] introduced a similar approach, which also utilizes a *facial recognition*

[12] `https://www.russian3dscanner.com/`

loss to enforce the retention of identity features, a *batch distribution loss* that encourages generalization, a *loopback loss* which ensures self-interpretable results and *multi-view identity loss* which improves the reconstruction from multiple view-points. Such approaches seem very promising in generalizing to unseen data, however, their results are currently of limited resolution and quality. More details on the differentiable renderers used here are provided in section 2.2.1.

Arguably, the first generic method that demonstrated that it is possible to reconstruct high-quality texture and shape from single "in-the-wild" images is the recently proposed GANFIT method [19]. GANFIT can be described as an extension of the original 3DMM fitting strategy and differentiable rendering [43], but with the following differences: (a) instead of a PCA texture model, it uses a progressively growing GAN [14] trained on thousands of high-resolution UV-maps, and (b) in order to preserve the identity in the reconstructed texture and shape, it uses features from a state-of-the-art face recognition network [135]. However, the reconstructed texture and shape is not rendering-ready due to (a) the texture containing baked illumination, and (b) not being able to reconstruct high-frequency normals or specular reflectance.

2.6 Photorealistic 3DMMs

The statistical paradigm on which 3DMMs are based makes them a robust facial shape and appearance inference model, but hinders their ability to reconstruct high-resolution textures, which would make the reconstruction appear realistic. Some of deep-learning methods (i.e. GANFIT [19]) described in Section 2.5.2 can generate textures of up to 576×384 pixels. However, they do not model facial reflectance, which is required for photorealistic rendering and for most models, the learned textures incorporate shadows and environmental illumination.

An extension to 3DMM for photorealistic relighting is the SHBMM [136], which constructs a 3DMM on 3D spherical harmonics bases. Recently, Smith et al. [11] extended the Basel Face Model [117] to model appearance with two PCA models for the diffuse albedo and specular albedo. Their approach can be used for state-of-the-art analysis by synthesis and the reconstruction can be physically rendered, however, the resolution and sharpness of

the generated appearance are still constrained by the use of a PCA model.

2.6.1 Image Translation-based Methods

Early applications on reflectance acquisition of human faces from "in-the-wild" images [49, 137] use an encoder-decoder architecture to infer reflectance in a low-resolution texture, illumination and shape, by changing the domain of the input image. Neural Face Editing [49] trains a GAN on unlabeled arbitrary images and adds a reconstruction loss based on the rendering of the results. In a similar manner, SfSNet [137] train a network with residual blocks, firstly supervised on labeled synthetic data from 3d reconstructions, and secondly semi-supervised on the labeled data and on "in-the-wild" images, with an added reconstruction loss. Both methods achieve exciting results, but are achieving very low-resolution results and they use a simplified Lambertian shading model that cannot accurately model and render the nature of human skin.

2.6.2 Combinatory Methods

During the last few years, some attempts have been made on the acquisition of high resolution geometry and reflectance acquisition of human faces. They combine a linear or non-linear 3DMM geometry reconstruction with a deep learning approach for reflectance, similar to the image-translation (Sec. 2.6.1) and material methods (Sec. 2.6.3).

A recent approach [138] uses a 3DMM fitting which retains the input image details and incorporates facial normal and displacement mappings derived from an image-translation deep network, resulting in representations with high-frequency details. Although this method demonstrates impressive results in geometry inference, it tends to fail in conditions with harsh illumination and extreme head poses, and does not produce re-lightable results, as the environment illumination is baked in the resulting albedo. Saito et al. [139] proposed a deep learning approach for data-driven inference of high-resolution albedo texture map of an entire face for realistic rendering, using an input of a single low-resolution face image with partial facial coverage. This has been extended to inference of facial mesostructure, given a diffuse albedo [140], which, however, needs to be captured in a constrained environment. Yamaguchi et al. [141] use a facial image as input to infer complete facial reflectance

and displacement maps besides albedo texture, which can be photo-realistically rendered and achieves impressive results.

2.6.3 Planar Material Acquisition

Reflectance acquisition for materials is a similar problem that shares methods with reflectance acquisition from faces. In its common form, a single photo of a planar surface is analyzed to recover characteristics of materials, such as wood, metal, stone, or plastic. The planar geometry of such materials simplifies the problem, however, the materials in question can have different reflectance properties (i.e. light interacts differently with dielectric or metallic surfaces).

Aittala et al. [142] capture flat materials with head-lit flash illumination, normalize the image mean and contrast and jointly predict the SVBRDF (diffuse and specular albedo, normals, and roughness), using an iterative optimization process with a differentiable renderer. Capturing the material with a flash removes shadows for the image and ensures the appearance of specular highlights. *Materials for the Masses* [143] achieved similar results with a simpler setup, using a single multi-channel encoder-decoder CNN architecture, which also classifies the material group. Deschaintre et al. [144] worked on a similar paradigm which also combines local and global features, while using a vast training material dataset by procedurally generating synthetic labeled data that can be photo-realistically rendered. Another approach [145] is using a similar encoder-decoder architecture without the need for an input constrained under flash illumination.

Chapter 3

Capturing Human Facial Assets at Scale

Contents

3.1 Diffuse-Specular Separation Optimization 40

3.1.1 Introduction 40

3.1.2 Diffuse-Specular Separation Overview 41

3.1.3 Processing Binary Gradients 43

3.1.4 Results 45

3.2 Capturing and Reconstruction Automation 47

3.2.1 The Capturing System 48

3.2.2 Capturing 48

3.2.3 Transfer and Pre-processing 49

3.2.4 Geometry Acquisition 49

3.2.5 Template Registration and UV Texturing 50

3.3 RealFaceDB Dataset 52

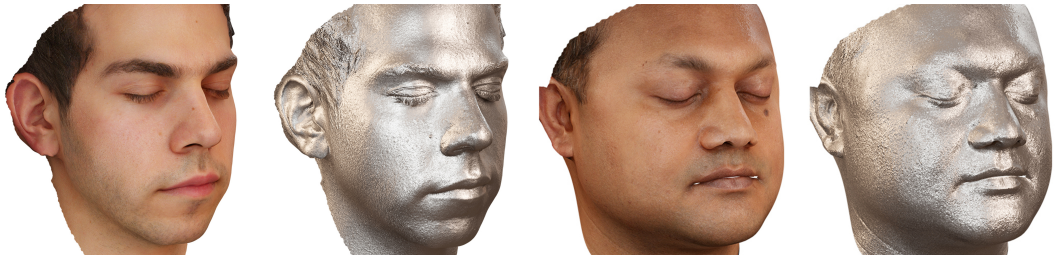
3.4 Conclusion 54

In this chapter, we describe the work done to improve the Imperial College Light Stage [146], specifically the diffuse-specular separation algorithm, in order to optimize it for facial shape and texture capture at scale (Sec 3.1) and increase its automation and robustness (Sec. 3.2). Finally, we introduce RealFaceDB, a public dataset captured with the optimized Light Stage (Sec. 3.3).

3.1 Diffuse-Specular Separation Optimization

Here we introduce the optimizations done on the diffuse-specular separation of [147], in order to make the approach multi-view and faster to compute. Published in *A. Lattas, M. Wang, S. Zafeiriou, and A. Ghosh, "Multi-view facial capture using binary spherical gradient illumination," in* ACM SIGGRAPH 2019 Posters*, pp. 1–2, 2019.*

3.1.1 Introduction



(a) Render (b) Geometry (c) Render (d) Geometry

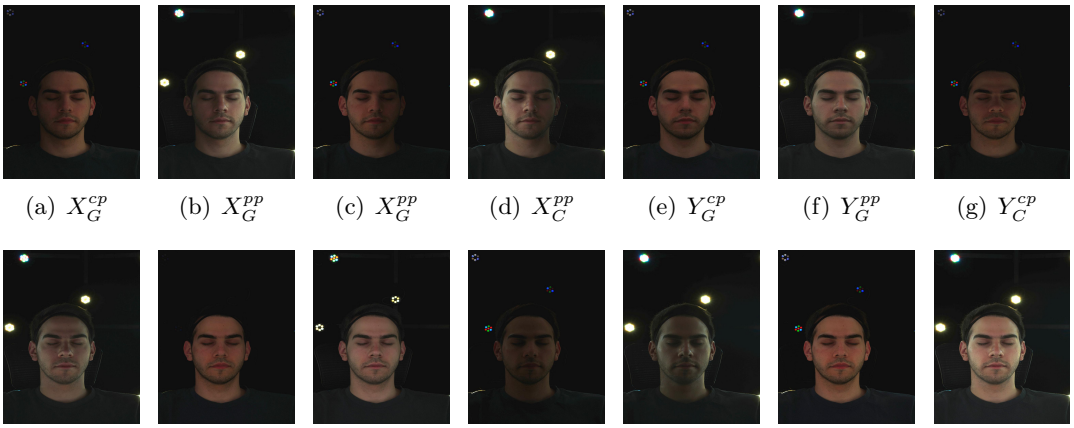
Figure 3.1: Renders of two subjects captured with our method. Left: full rendering, right: geometry-only rendering.

High-resolution facial capture has received a lot of attention in computer graphics due to its application in the creation of photorealistic digital humans for various applications ranging from film VFX, games, VR, etc. Here, the state-of-the-art method for high-quality acquisition of facial geometry and reflectance employs polarized spherical gradient illumination [74, 76]. The technique has had a significant impact on facial capture for film VFX, recently receiving a Technical Achievement award from the Academy of Motion Picture Arts and Sciences [148]. The method, however, imposes a few constraints due to the employment of polarized illumination, and requires the camera viewpoints to be located close to the equator of the LED sphere for appropriate diffuse-specular separation for multiview capture [76]. The employment of polarization for reflectance separation also reduces the amount of light available for exposures and requires double the number of photographs (in cross and parallel polarization states), increasing the capture time and the number of pictures required for each face scan.

In this work, we instead adapt our recently proposed diffuse-specular separation technique using binary spherical gradient illumination [147] for multiview face capture. Instead of

relying on polarized illumination, diffuse-specular separation using binary spherical gradients relies on color-space separation of reflectance (assuming dichromatic reflectance of a dielectric material such as skin). Besides requiring the acquisition of fewer images for facial capture (with higher light efficiency) than polarized spherical gradients, another advantage of the method is that it can be employed with LEDs that can only switch between a binary on-off state and does not require modulation of intensities to create gray levels. The high-resolution facial normal map acquired using binary gradients exhibits sharper skin mesostructure details (obtained from specular reflectance) compared to polarized spherical gradients, while achieving high-quality reflectance separation for a realistic rendering of skin appearance. In the following, we discuss some specific modifications to the processing of data acquired with binary spherical gradient illumination proposed by Kampouris et al. [147] that we found to be useful for multiview face capture.

3.1.2 Diffuse-Specular Separation Overview



(a) X_G^{cp} (b) X_G^{pp} (c) X_G^{pp} (d) X_C^{pp} (e) Y_G^{cp} (f) Y_G^{pp} (g) Y_C^{cp}

(h) Y_C^{pp} (i) Z_G^{cp} (j) Z_G^{pp} (k) Z_C^{cp} (l) Z_C^{pp} (m) F^{cp} (n) Z^{pp}

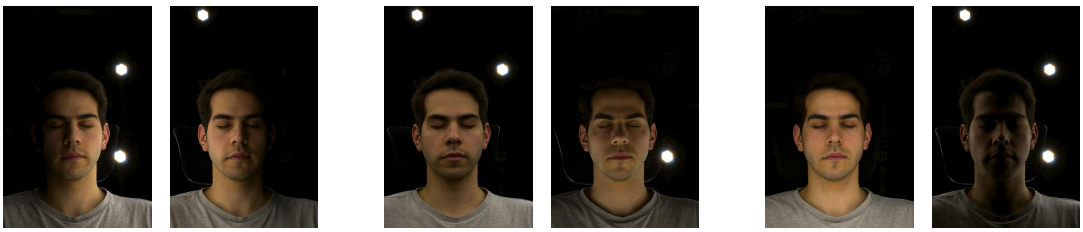
Figure 3.2: Example captured image set $\mathcal{C}^{pol}$ of images with polarized spherical gradients, used in [76] and our extension. Central camera images are shown.

The state-of-the-art reconstruction method of Ghosh et al. [76] requires capturing a set of 14 polarized images $\mathcal{C}^{pol}$ per camera:

$$\mathcal{C}^{pol} = \{X_G^{cp}, X_G^{pp}, X_C^{cp}, X_C^{pp}, \quad Y_G^{cp}, Y_G^{pp}, Y_C^{cp}, Y_C^{pp}, \quad Z_G^{cp}, Z_G^{pp}, Z_C^{cp}, Z_C^{pp}\} \tag{3.1}$$

where X, Y, Z are the principal direction for the gradients, G is the gradient and C its complement along the principal direction, cp denotes a cross-polarized capture and pp

and parallel-polarized capture. When using current commercial cameras for the capture, in burst mode, Ghosh et al. [76] requires an estimated 3.5 seconds, which may result in subtle subject motion that shifts the pixel alignment and results in smoothed details in the reconstruction.



(a) X_G (b) X_C (c) Y_G (d) Y_C (e) Z_G (f) Z_C

Figure 3.3: Example captured image set $\mathcal{C}^{bin}$ of images with unpolarized binary spherical gradients, used in [147] and our extension. Central camera images are shown.

On the contrary, the alternative base method of [147] approximates the diffuse-specular separation based on a color space analysis. This requires a set of only six images $\mathcal{C}^{bin}$ per camera and does not require polarisers on the cameras or the light sources:

$$\mathcal{C}^{bin} = \{X_G, X_C, \quad Y_G, Y_C, \quad Z_G, Z_C\} \tag{3.2}$$

where X, Y, Z are the principal direction for the gradients, G is the gradient and C its complement along the principal direction. Capturing with this method, not only requires a simpler acquisition setup (no polarizers), but the capture requires 2 seconds in burst mode, minimizing subject motion. Moreover, since we aim at mass capturing, we expect our subjects to be sensitive to the flashing lights and more prone to move during capture. Therefore, we choose this method for our capturing experiments. However, Kampouris et al. [147] method is restricted to frontal view specular-diffuse separation, requires significant computational time and produces noise on the sides of the face. In this section, we address these shortcomings by proposing multiple modifications, which simplify the setup without sacrificing the quality of the results.

3.1.3 Processing Binary Gradients

Diffuse Normals

The data acquired using binary spherical gradient illumination directly compute a mixed photometric normal (due to a diffuse and specular response mix under unpolarized illumination). Hence, Kampouris et al. first proposed employing rotation of the RGB colors into *suv* space [149] and computing a pure diffuse normal using the magnitude of the chroma components (u, v). Instead, we observe that under white illumination, the different color channels of measured skin reflectance have the same amount of specular reflectance mixed with a spectrally varying amount of diffuse reflectance. Hence, we can simply compute a pure diffuse normal using the difference between the brightest and darkest color channels of reflectance, which for human skin are the red and blue channels, respectively. This spectral differencing step completely removes the specular signal, isolating a pure diffuse component.

Specular Albedo and Diffuse Albedo

Our baseline here is the diffuse-specular separation obtained by Kampouris et al. [147] using their proposed analytic solution obtained from HSV color transformation analysis. The quality of separation is quite good at the sides of the face but suffers from a poor signal-to-noise ratio, of the separated specular component near the frontal areas of the face. This results in sub-optimal separation and some specular pollution near the frontal regions of the face in the separated diffuse albedo.

Instead, we employ a linear system to solve for the specular component based on the observation that under the binary gradient and complement pairs, the brightly lit condition consists of diffuse plus specular while the darker condition consists of pure diffuse component [147]. Without loss of generality, assuming a surface appears brighter under gradient G and darker under complement C, we can formulate the linear system solution as follows:

$$\begin{bmatrix} G \\ C \end{bmatrix} = \begin{bmatrix} N_d & 1 \\ (1 - N_d) & 0 \end{bmatrix} \begin{bmatrix} D \\ S \end{bmatrix} \tag{3.3}$$

Here, D is the diffuse component and S, the specular component of the signal in a color channel, and N_d is the corresponding component of the diffuse normal (shifted to $[0, 1]$) for that principal direction. We solve this linear system for the green channel since it has the best signal-to-noise ratio.

We separately solve Eq. 3.3 twice, once each for the X and Y gradient - complement pairs, to obtain two different estimates of D and S. Assuming monochromatic specular reflection, we just compute the final estimate of specular reflectance as the mean of the two solutions for S:

$$S = \frac{S_X + X_Y}{2} \tag{3.4}$$

Finally, we simply subtract the monochromatic specular component S from the mixed albedo (gradient G plus complement C for a principal direction) to obtain the separated diffuse albedo:

$$D = G + C - S \tag{3.5}$$

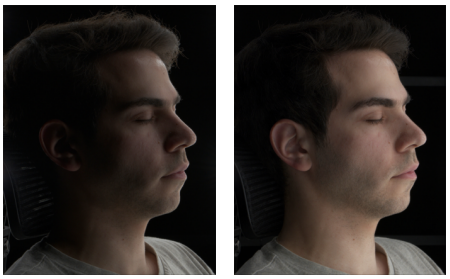
As can be seen in (Figure 3.6) we obtain a much higher quality of reflectance separation when using the linear system solution.

Specular Normals

Kampouris et al. [147] estimate a linear combination weight α for the diffuse and specular component of the normals, using the blue channel of the captured gradients. We follow the same practice for the estimation of the specular normal. However, we empirically found that employing the green channel of the data to compute the linear mixing weight α gave slightly higher quality results with less noise for the specular normal estimation step. Here, the mixing weight α is employed to compute the specular normal as a linear combination of the mixed normal and the pure diffuse normal.

Synthetic Rotation of Gradients

As mentioned above, in the multi-view setting, we find that the best estimates are calculated from the local X and Y gradient-complement pairs. Here, local refers to gradients and complements aligned with respect to the camera viewing direction. However, given the

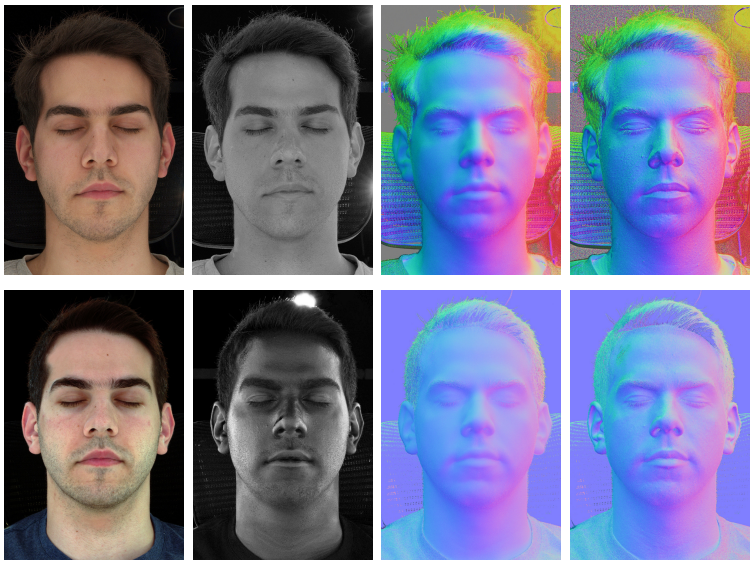


(a) X_G (b) Q_G

Figure 3.4: Synthetic rotation of the X_G gradient captured from a side camera, using our method (Eq. 3.2), resulting in a Z-axis gradient in the space of that camera (termed Q_G).

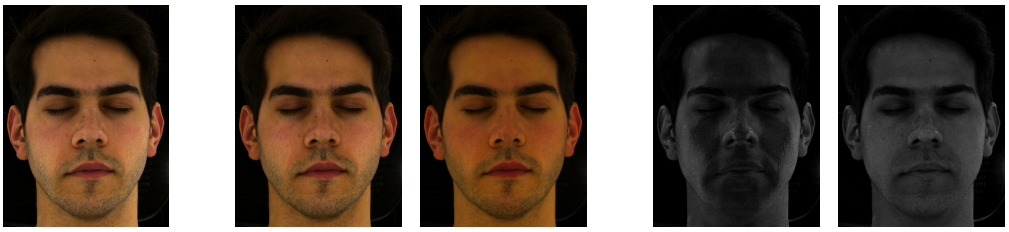
multi-view capture setup (we employ nine cameras in our setup), the acquired binary gradients and complements are initially axis-aligned with respect to the global coordinates of the LED sphere. Hence, before we can compute diffuse-specular separation of the albedo, we need to rotate the binary gradients X computationally, and their complements into the local coordinate frame of each camera (termed Q) and then employ the local Q and Y pairs accordingly to estimate the specular albedo. Note that this rotation of the acquired gradients and complements can be done afterwards, as together they form a steerable basis.

3.1.4 Results



(a) $\mathbf{A}_D$ (b) $\mathbf{A}_S$ (c) $\mathbf{N}_D$ (d) $\mathbf{N}_S$

Figure 3.5: Comparison of reflectance and normal maps acquired using polarized spherical gradients (top-row) [76], and our binary spherical gradients (bottom-row). Note that the diffuse albedo $\mathbf{A}_D$ differs slightly in hue, due to different color space, while the polarized capture specular albedo $\mathbf{A}_S$, exhibits fresnel reflection on the sides.



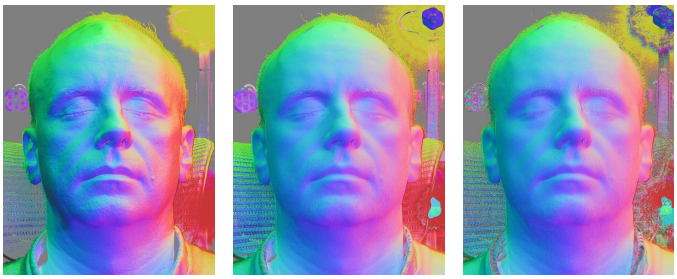
(a) Mix. Albedo (b) $\mathbf{A}_D$, [147] (c) $\mathbf{A}_D$, ours (d) $\mathbf{A}_S$, [147] (e) $\mathbf{A}_S$, ours

Figure 3.6: Results of the linear system solution for diffuse, specular separation (Eq. 3.3) compared to the color-space separation employed by [147].

Fig. 3.8 shows a comparison of the estimated diffuse and specular albedo and normals using polarized spherical gradient illumination [76], compared to those estimated using binary spherical gradient illumination with our modifications. There are qualitative similarities, as well as some noticeable differences in the estimated reflectance and normal maps when using these two techniques. Interestingly, the specular normals estimated using binary spherical gradient illumination is sharper than that estimated using polarized gradients [76]. Part of the reason is that the polarized capture requires double the number of photographs and is hence more susceptible to minor subject motion during capture. While the diffuse-specular separation of albedo is of slightly higher quality when employing polarized illumination, the quality of reflectance separation with binary spherical gradient illumination is of sufficiently high quality to enable realistic rendering of the face scan (Fig. 3.8). Note that the diffuse albedo acquired by the two methods exhibits slightly different color tones due to the different types of LEDs used and the polarization filters. We can correct this by color transformation to sRGB. Also, the polarized capture [5] specular albedo $\mathbf{A}_S$ exhibits Fresnel gain on the sides. Although it can be compensated in post-processing [150], our methods suffer from such a problem.

Compared to [147], the linear system solution (Eq. 3.3) achieves a more accurate diffuse albedo, since the predicted specular albedo is less biased when using the two brighter estimates from X, Y and Z axes. This is visualized in comparison with [147] in Figure 3.6.

We compare our novel spectral differencing diffuse normals separation in Fig. 3.7, with the mixed normals and the *suv*-based approach [147]. Our simple $red - blue$ difference-based diffuse normal exhibits very similar qualitative softness and blur exhibited by the diffuse normal computed using *suv*-space projection. The proposed spectral differencing method is much simpler and faster to compute, which is why we adopt it in our pipeline.



(a) $\mathbf{N}_M$ (b) $\mathbf{N}_D^{suv}$ (c) $\mathbf{N}_D$

Figure 3.7: Comparison of mixed normals $\mathbf{N}_M$ (a) obtained directly using binary spherical gradient illumination vs. pure diffuse normals obtained using *suv* projection $\mathbf{N}_D^{suv}$ (b), and our proposed spectral differencing (c) $\mathbf{N}_D$.



(a) Render, ours (b) Shape, ours (c) Render, [76] (d) Shape, [76]

Figure 3.8: Comparison of rendering and geometry between our method and the state-of-the-art method of Ghosh et al. [76]

The final separated diffuse and specular albedo and normals are shown and compared with the state-of-the-art [76] in Figure 3.8. Our separation is comparable to the state-of-the-art, achieves a view-independent specular albedo and sharper specular normals, while introducing a few artifacts in the grazing angles of the diffuse albedo. Moreover, it uses the significantly simplified capturing setup of [147]. Finally, Figure 3.8 shows the comparison of rendered reconstructions between our pipeline and the state-of-the-art.

3.2 Capturing and Reconstruction Automation

Here we introduce the Light Stage capturing and reconstruction automation and robustness optimization, which was carried at the Imperial College Light Stage [147], in order to enable the capture of a vast dataset of facial reflectance and geometry using the aforementioned diffuse-specular separation method.

3.2.1 The Capturing System

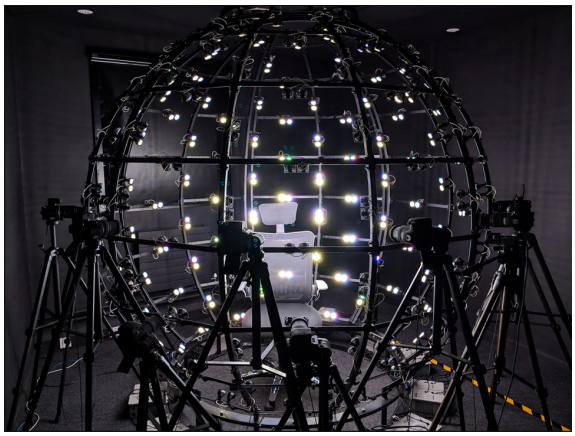


Figure 3.9: The Imperial College Multispectral LightStage [146]

Our capture setup consists of the pre-existing Imperial Multi-spectral LED [146] sphere with 168 LED lighting nodes and a multiview facial capture setup consisting of 9 DSLR cameras (Canon 800D) (Figure 3.9). For acquisition, we employ only the white LEDs on the LED sphere (Philips Color Kinetics iW MR gen3) to illuminate a subject with six binary spherical gradient illumination conditions (along principal X, Y and Z directions and their complements) as proposed by [147] (Figure 3.3), and amended by our method (Sec. 3.1).

3.2.2 Capturing

We rapidly acquire a face under these six binary illumination conditions from our nine cameras operating in burst mode to minimize any subject motion during capture. The data acquired are then employed for both multi-view facial geometry reconstruction and for high-quality reflectance separation which leads to the estimation of separated diffuse and specular photometric normals and albedo for rendering. We have empirically found that the off-the-shelf DSLR cameras used may tremble when they start capturing in burst mode. To avoid images with motion blur due to camera shaking, we capture extra two photos in the beginning, which are then discarded.

3.2.3 Transfer and Pre-processing

Transferring and pre-processing the captures is an important step; it introduces a significant bottleneck when capturing multiple subjects. Transferring the captures with the internal SD cameras requires manual work and the WiFi transferring requires over 10 minutes per capture set.

Therefore, our pipeline is devised as follows. The DSLR cameras are connected via a USB hub to the data computer using the MTP file protocol. Each camera's storage is automatically queried for the captured data, and the corresponding images are stored in folders per camera and capture, in an anonymized directory. Each transferred photograph is converted from its RAW format linearly to 16bit TIFF. Finally, the last color chart capture is used to automatically white balance the photographs, concluding the pre-processing.

3.2.4 Geometry Acquisition

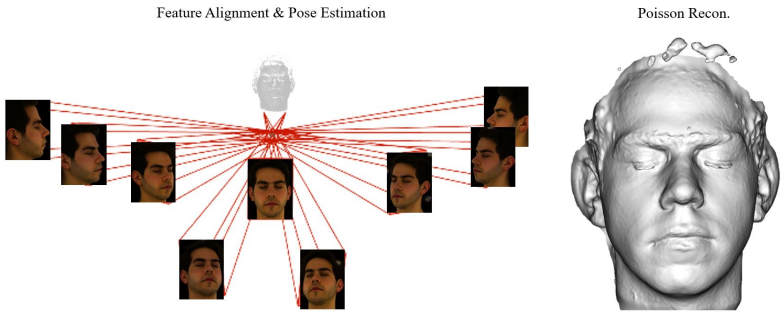


Figure 3.10: Stereo reconstruction on mixed albedo images, with Poisson filter applied for smoothing on the reconstructed geometry, using COLMAP [151].

The next step in the pipeline is the acquisition of the base reconstruction, directly from the captured images, using a photogrammetry method based on structure-from-motion [152, 153]. Several solutions exist and our personal choice is COLMAP [151], which is open-source, fast, and robust with Light Stage data. Alternatively, Reality capture[1] is the state-of-the-art but expensive commercial software at the moment, while Agisoft MetaShape (previously called Photoscan)[2] also works effectively with Light Stage captures.

In the case of the polarized spherical gradient capture [76], the cross-polarized full-on X_F^{cp} (see Eq. 3.1) captures from each camera can be used for photometric stereo, as they exhibit

[1] www.capturingreality.com
[2] www.agisoft.com

the same and only diffuse reflectance for all viewpoints, while having no shadows except ambient occlusion. For our capture using the unpolarised binary spherical gradient capture [147], the pure-diffuse albedo images that can be calculated after the separation, exhibit high-frequency view-dependent noise, because of the nature of the algorithm. We find that the unpolarised full-on captures X_F (see Eq. 3.2) can be used, even if they contain some view-dependent specular highlights. These produce noisy areas, which can be alleviated with a Laplacian filter.

3.2.5 Template Registration and UV Texturing

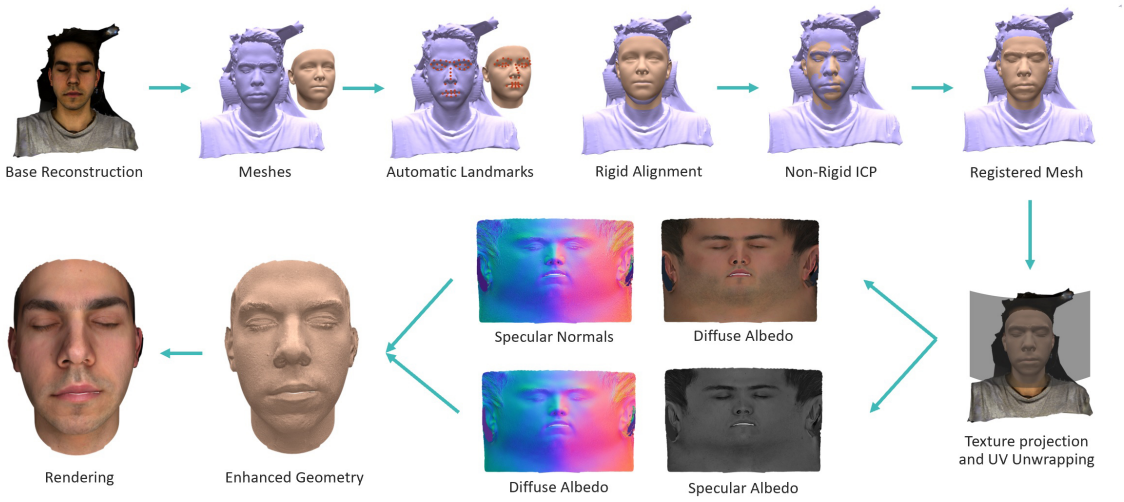


Figure 3.11: Post-processing of the stereo-reconstruction, as described in section 3.2.5. Clockwise, from the top left and following the arrows: (a) we start with the mesh acquired using stereo-reconstruction, (b) the mesh is automatically fitted with 43 landmarks using a rendered image of the reconstructed mesh, (c) the template mesh is rigidly aligned with the reconstruction, using an affine transformation computed using their landmarks, (d) using Non-rigid ICP, the template is registered on the reconstructed mesh, (e) the per-camera diffuse and specular components are projected on the registered mesh and unwrapped in UV space using the template's parameterization, (f) the enhanced geometry is acquired by displacing a subdivided registered template, (g) the mesh can be rendered using the reflectance UV maps.

Following the base reconstruction, we create a post-processing fully automated pipeline, which provides us with high-resolution and rendering-ready data, that are in dense correspondence (pixel-to-pixel and vertex-to-vertex) and ready for deep-learning applications. The high-level visualization is shown in Figure 3.11 and each step is described below:

- **Base Reconstruction Post-Processing**: Following the base reconstruction using one of the methods described in section 3.2.4, the resulting triangular mesh is colored

per-vertex using the corresponding values in the source images. Then, any holes are closed by connecting their vertices with new faces and finally, the mesh topology is simplified by reducing the triangle numbers to 300.000 with quadratic edge collapse.

- **Template Mesh Preparation**: We acquire a version of the LSFM [154] trained on 10.000 identities. This model has a standardized topology of 53.215 vertices and blendshapes for identity and expression. We extract the mean shape of the LSFM as a static triangular mesh. Using the landmarks or the base reconstruction, the identity or expression of the starting mesh can be pre-optimized.

- **Automatic Landmark Localization**: We automatically landmark both the base reconstruction and the template, using the iBug Facial Point Annotations [125]. The template mesh has a standardized topology and therefore known landmark vertices. We landmark the base reconstruction using the Python library for deformable models, Menpo [155] and render it facing the camera using orthographic projection, after it has been rotated to face the renderer's camera, using the coordinates of the cameras. The landmarks are then localized in 2D using an AAM [8] and projected back to the base reconstruction mesh, given the known parameters of the renderer's virtual camera.

- **Rigid Alignment**: Following the *Generalized Procrustes Analysis* [156], we compute an affine transformation between the landmarks of the template mesh and the base reconstruction mesh, and rigidly align the template mesh to the scale, pose, and position of the base reconstruction mesh.

- **Non-Rigid Deformation (Registration)**: Having closely aligned both meshes, we iteratively deform the template mesh so that its geometry matches the base reconstruction's geometry and bring them into correspondence. Similarly to [10], we are using a Non-Rigid Iterative Closest Point (NICP) algorithm [157]. We find that eight iterations of deformation with gradually reduced weights successfully deform the template. To reduce errors on the noisy sides of the face, we increase the stiffness weights of each vertex, based on its distance from the nose tip. Finally, to correctly deform the more complex lips region, we run another four iterations where the stiffness weights are only relaxed in an ellipsoid around the center of the lips' landmarks.

- **Texture Projection**: We project the corresponding texture from each captured photograph to the registered mesh, creating texture maps for each captured modality. This process is repeated for the diffuse and specular albedo and normals which have been separated in camera space (Sec. 3.1).

- **UV Parameterization**: The stitched UV parameterization may be discontinuous and repetitive and, therefore, a sub-optimal candidate for learning algorithms. We have manually constructed an "unwrapped" UV space, as shown in Figure 3.11 which keeps the central part of the face undistorted while stretching the outer parts to form a rectangle, a standard practice in graphics. For each texture type, using the texture coordinates of the unwrapped UV parameterization, we sample from the existing textures to synthesize the ones shown in Figure 3.11.

- **Optional Post-Processing for Rendering**: The registered reconstruction, along with the reflectance maps, can be used for photorealistic rendering in shaders that support physically based rendering. Moreover, the specular, normal map can be used to calculate a displacement map, as in [158], which can then be used to enhance a subdivided registered mesh of the registered mesh, by displacing its vertices.

In the end, the reconstructed geometry is represented in a known topology, that is rigged with expression and identity blendshapes, has a standardized and deep-learning-friendly UV parameterization, all reflectance maps needed in that UV parameterization and in dense correspondence with all other reconstructions.

3.3 RealFaceDB Dataset

The facial reflectance capturing methods and improvements introduced in Sec. 3.1 and the automation of the capturing setup introduced in Sec. 3.2, enabled the capture of a large facial reflectance dataset, whose parts were made publicly available upon the publication of [23] (Sec. 5.1).

The dataset, termed *RealFaceDB*, consists of facial reflectance data parameterized on a UV texture map, which can be shaded photorealistically and used to research facial reflectance or train machine learning algorithms for facial reflectance acquisition. More specifically,

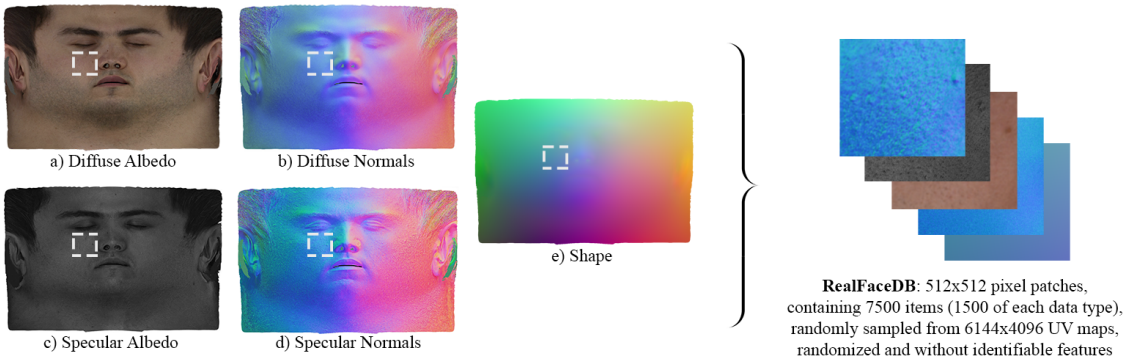


Figure 3.12: An example subject from the RealFaceDB dataset [23]. The reconstructed facial reflectance maps for each subject's reconstruction (Diffuse Albedo, Specular Albedo, Diffuse Normals, Specular Normals, and Shape) are split into patches that, after randomization, form the dataset.

200 subjects visited the Imperial College Light Stage [146] over the course of 4 months, and were captured using the methods of Sec. 3.1 and Sec. 3.2, by the author of this Thesis. The captures consist of a neutral closed-eyes scan and a number of facial expressions, including anger, smile, disgust, sadness, surprise, and fear. The FACS [159] expressions set was not used, as the subjects were randomly pooled from students and staff of Imperial College, and FACS requires time-consuming training to be followed correctly. The selected facial reflectance scans were finally processed and transferred to a standard UV parameterization, carefully constructed to be suitable for convolutional neural network training. Given the high resolution (6144×4096 pixels) of the reflectance maps, which is unwieldy for any current machine learning algorithm, the reflectance maps were cut into overlapping square patches of 512×512 pixels. As a result, the final reflectance dataset is still the largest public of its kind at the time of writing this report.

Given the sensitive nature of the facial data, only a randomized sample of the acquired reflectance patches was used to form the dataset. More specifically, 20% of the patches comprising each subject was added to the dataset, and the dataset was anonymized and randomized, with only the correspondence between different measurements of the same patch kept. In this manner, a subject cannot be reconstructed as a recognizable whole. Moreover, in order to enable the shading of these patches, we encode the geometry corresponding to each patch into the same UV topology ($x, y, z \rightarrow R, G, B$) and provide it along with the patches. Various shading models can be then used, as shown in AvatarMe^{++} [24] (Sec. 5.4.6, 6.3.3).

Despite its size, the dataset is limited in certain characteristics of the subjects. As the capturing device is not mobile, most subjects were UK residents and students or staff of Imperial College London. Tab. 3.1 shows their demographic information:

Male	**Female**	**Other**	**0-25 y.o.**	**25-40 y.o.**	**over 40 y.o.**
63.2%	36.7%	0%	55.8%	37.3%	6.9%

White	**Asian**	**Black**	**Other**
61.1%	26.3%	5.5%	7.1%

Table 3.1: RealFaceDB captured subject demographics, on gender, race, and age.

The dataset is available at the project page of AvatarMe [3]. At the time of writing, over 30 senior research staff from various universities and countries have requested and received access to the dataset.

3.4 Conclusion

In this chapter, we have shown the improvements made on an existing facial capturing setup and method, in order to capture multi-view facial reflectance and geometry, which were published as a short paper at SIGGRAPH [20]. Moreover, we have described the engineering work required to make the capturing setup able to capture and process tens of subjects per day and finally, we have introduced the dataset RealFaceDB [23]. Both the capturing improvements and the dataset have been crucial for the work described in the following three chapters.

[3] RealFaceDB available at `https://github.com/lattas/avatarme` .

Chapter 4

Practical and Scalable Desktop-based High-Quality Facial Capture

Contents

4.1 Introduction 57

4.2 Related Work 58

4.2.1 Active Illumination 58

4.2.2 Passive Capture 60

4.3 Desktop-based Capture System 61

4.3.1 Tablet-based Setup 61

4.3.2 Monitor-based Setup 63

4.4 Illumination 63

4.4.1 Modulated Binary Illumination 63

4.4.2 Spectral Multiplexing and Cross-Talk 64

4.4.3 Form-Factor Modulation 66

4.4.4 Multi-view Capture 66

4.5 Method 67

4.5.1 Acquisition using White Illumination 67

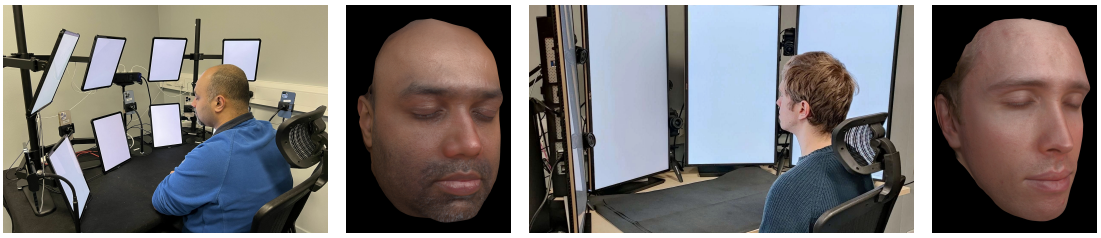
4.5.2 Color-Multiplexed Illumination 69

4.5.3 Specular Roughness Estimation 70

4.5.4 Dynamic Capture 70

4.5.5 Base Geometry Acquisition 71

4.6 Experiments 71
4.6.1 Evaluation 71
4.6.2 Static Capture 72
4.6.3 Dynamic Capture 72
4.7 Limitations 73
4.8 Conclusions 74



(a) Tablet-based Setup (b) Render (c) Monitor-based Setup (d) Render

Figure 4.1: Two proposed novel desktop-based setups (a, c) for high-quality facial capture (b, d). Left: setup consisting of a set of portable mobile devices – tablets and smartphones, for static facial capture. Right: setup consisting of a set desktop LCD displays for static and dynamic facial capture.

In this chapter, we present a novel desktop-based system for high-quality facial capture including geometry and facial appearance. The proposed acquisition system is highly practical and scalable, consisting purely of commodity components. The setup consists of a set of displays for controlled illumination for reflectance capture, in conjunction with multiview acquisition of facial geometry. We additionally present a novel set of binary illumination patterns for efficient acquisition of reflectance and photometric normals using our setup, with diffuse-specular separation. We demonstrate high-quality results with two different variants of the capture setup – one entirely consisting of portable mobile devices targeting static facial capture, and the other consisting of desktop LCD displays targeting both static and dynamic facial capture.

This chapter is supported by a published SIGGRAPH 2022 talk paper (*A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera, G. Chawla, and A. Ghosh, "Desktop-based high-quality facial capture for everyone," in* SIGGRAPH Talks*, 2022*), and a published ECCV 2022 paper (*A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera,*

G. Chawla, and A. Ghosh, "Practical and scalable desktop-based high-quality facial capture," in Proceedings of the European Conference on Computer Vision*, 2022*), which was presented as an oral presentation.

4.1 Introduction

Realistic reconstruction and rendering of human faces has been a long standing goal in computer vision and graphics. It has wide ranging applications in well known sectors such as entertainment, advertising, AR/VR, training systems, etc., as well novel applications of digital avatars in social media, and virtual presence in the envisioned *metaverse*. Great strides have been made towards achieving this goal over the last two decades, both in terms of development of photo-realistic rendering algorithms, as well as realistic modeling of facial shape and appearance. The latter has been revolutionized with the development of acquisition techniques for high-quality 3D facial capture. However, creation of realistic digital characters from acquired data requires very high-quality data capture typically acquired using custom designed and expensive capture systems such as the Lightstage [72][73][74][76]. This usually restricts the application of such acquisition techniques in practice to big-budget film and games VFX, and/or to organizations that can either afford to access such setups at specific locations or to construct such capture systems in-house. This severely limits the accessibility of such 3D acquisition technology for wider applications.

Instead, this work aims to make acquisition of high quality facial geometry and appearance of a subject much more practical and accessible, thereby democratizing the creation of realistic virtual humans for various applications. Specifically, we propose a practical and scalable desktop-based setup for high-quality facial capture. The setup consists of a set of displays for active control of illumination for facial appearance capture, in conjunction with multiview acquisition of facial geometry. We present two different variants of the capture setup constructed entirely using commodity components – a highly portable setup consisting of a set of mobile devices (tablets and smartphones) targeting static facial capture, and a setup consisting of a set of desktop LCD displays targeting both static and dynamic facial capture (see Fig. 4.1). We additionally present a novel set of binary

illumination patterns emitted by the display panels in our setup for efficient acquisition of diffuse-specular separated reflectance and photometric normals. We present high-quality results of facial capture using both the proposed setups, and demonstrate the results to be qualitatively competitive with those obtained using dedicated, expensive capture setups.

To summarize, this work introduces:

1. Two novel, scalable desktop setups for reflectance and shape capture, comprising only of commodity components.
2. A novel photometric acquisition pipeline with diffuse-specular separation employing binary illumination that achieves high-quality results with the proposed desktop setups.
3. Novel design of spectral multiplexing for binary illumination that enables high-quality two-shot capture and estimation of spatially-varying specular roughness.
4. Demonstration of applicability of proposed setup and spectral multiplexing for dynamic facial appearance capture.

4.2 Related Work

Here we limit the discussion to the most related works on facial capture. For a broader discussion, we refer the reader to an excellent surveys on the topic [160][161].

4.2.1 Active Illumination

Photometric stereo systems have traditionally been popular in computer vision for estimating surface normals, in conjunction with color-multiplexed illumination for dynamic capture [70] [71]. Debevec et al. [72] first proposed a specialized light stage setup to acquire a dense reflectance field of a human face for photo-realistic image-based relighting applications. They also fit microfacet-BRDF based reflectance using the acquired data. Weyrich et al. [73] instead employed an LED sphere and multiple cameras to densely record facial reflectance and computed view-independent estimates of spatially varying facial reflectance from the acquired data for driving a microfacet BRDF model-based rendering.

Subsequently, Ma et al. [74] introduced polarized spherical gradient illumination (using an LED sphere) for efficient acquisition of the separated diffuse and specular albedos and photometric normals of a face, and demonstrated high quality facial geometry including skin mesostructure as well as realistic rendering with the acquired data.

Ghosh et al. [75] further extended the acquisition method to acquire layered facial reflectance using a combination of polarization and structured lighting, and further extended the view-dependent solution of Ma et al. for multi-view facial acquisition with polarized spherical gradient illumination [76]. These techniques have had significant impact in film VFX. Fyffe et al. [77] employed the method of [78] for temporal alignment of spherical gradient illumination for dynamic performance capture. Their method involves a heuristic based separation of reflectance which is not accurate and the capture process requires high speed cameras. Fyffe & Debevec [79] further proposed a single-shot method using a complex setup of a polarized RGB LED sphere for color-multiplexed spherical gradients that are further polarized using the method of [76] for diffuse-specular separation.

Closer to our work, Kampouris et al. [80] have employed binary spherical gradient illumination using an LED sphere for facial appearance capture including diffuse-specular separation of albedo and photometric normals. We employ a similar analysis of binary illumination in our work for diffuse-specular separation. However, we demonstrate our lighting patterns in conjunction with our illumination setup to result in fewer measurements for facial capture, even enabling high-quality results with spectral multiplexing which is not achieved by [80]. More recently, Guo et al. [162] have employed complementary pair of color-multiplexed spherical gradient illumination patterns using a spectral LED sphere for estimating diffuse and specular reflectance and surface normals for full-body volumetric capture.

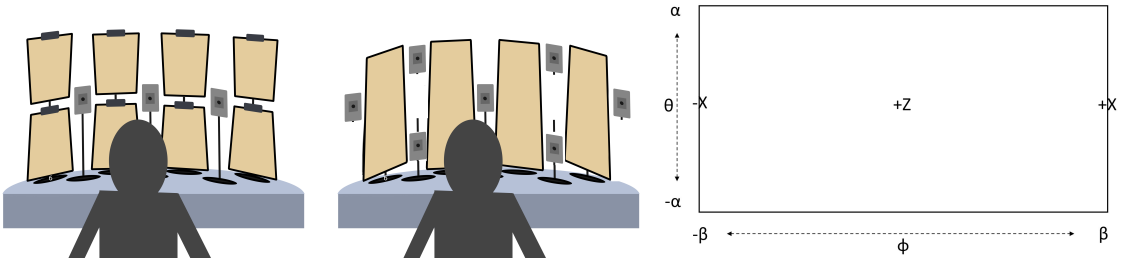
Unlike their approach, we employ a much simpler acquisition setup and estimate a broader set of spatially-varying reflectance parameters from each camera viewpoint using our spectral multiplexing approach. An alternate approach that does not require an LED sphere is the work of Fyffe et al. [81] on static facial appearance capture employing off-the-shelf photography hardware. However, the approach does not extend to dynamic facial appearance capture and still requires quite a complex hardware setup consisting of 24 cameras and 6 flashes that are triggered in sequence within a few milliseconds. Our approach is

inspired by these previous works. However, we aim to significantly simplify the capture process using commodity components and propose highly scalable desktop-based active illumination systems for high-quality facial capture. Very related to our approach is the recent work of Sengupta et al. [163] who employ a single desktop monitor to illuminate a face using a regular video sequence in order to learn a facial reflectance field with limited angular coverage. However, the goal of this work is to support facial relighting for video conferencing applications. Our desktop setup with brighter and wider illumination coverage is much more suitable for high-quality multiview capture, and we propose lighting patterns for direct acquisition of reflectance and photometric normals instead of learning based inference.

4.2.2 Passive Capture

Researchers have also investigated approaches for passive acquisition of faces and full-body. Such acquisition is particularly well suited for performance capture since active approaches usually require time-multiplexed illumination, imposing requirements of high frame rate acquisition, lighting control and synchronization. A popular approach has been to employ uniform constant illumination for multi-view facial capture [82][83]. Such an approach enables estimation of an albedo texture under flat lit illumination for rendering purposes besides facial geometry reconstruction based on multi-view stereo. High-frequency mesostructure is further embossed on the base-geometry using a high-pass filter on the diffuse texture [82]. The approach has been extended for reconstructing facial performances with drift-free tracking over long sequences using anchor frames [84]. While producing very good qualitative results for facial geometry, the estimated albedo is not completely diffuse and contains a small amount of specular reflectance baked into the texture and the embossed mesostructure is plausible rather than accurate. More recent works have extended passive capture systems for achieving facial reflectance estimation using inverse rendering in conjunction with view-multiplexing [85][86]. While achieving impressive results for passive capture, the quality of acquired reflectance maps is not quite at par with active illumination systems. Importantly, the capture process still requires a dedicated studio like setup with lighting control and mounting structure for cameras, and hence is not as portable or scalable as our proposed desktop-based capture system.

4.3 Desktop-based Capture System



(a) Tablet-based setup (b) Monitor-based setup (c) Illumination Coverage

Figure 4.2: Schematics of the two proposed desktop-based facial capture systems. (a): Tablet-based setup consisting purely of mobile devices. (b): Monitor-based setup consisting of four 27" LCD monitors mounted in portrait mode. (c): Frontal hemispherical zone (latitude-longitude parameterization) of directions covered by the display illumination. The zone spans $\beta < \pm 90°$ along the longitudinal direction, while spanning $\alpha < \pm 45°$ along the latitudinal direction.

We propose two desktop-based setups for facial capture in this work. Both setups employ specific types of display panels for controlled illumination, while employing a set of cameras for multiview acquisition. Both setups have been designed to provide controlled illumination over a frontal hemispherical zone of directions spanning $\beta < \pm 90°$ along the longitudinal directions ϕ, while spanning $\alpha < \pm 45°$ along the latitudinal directions θ (see supplemental material). Such a zone has previously been shown to include majority of facial surface normals [81].

In Fig. 4.2, we present schematics of the two proposed desktop capture systems – the first one employing portable mobile devices (tablets and smartphones), and the second one employing larger LCD monitors for piece-wise continuous illumination on a subject's face. The displays are arranged to illuminate a subject's face from a frontal hemispherical zone of directions (Fig. 4.2), spanning $\beta < \pm 90°$ along the longitudinal direction, while spanning $\alpha < \pm 45°$ along the latitudinal direction.

4.3.1 Tablet-based Setup

Our first desktop setup consists purely of a set of mobile devices – eight tablets and five smartphones, that are mounted on a desk as shown in Fig. 4.1 (a). The tablets are arranged in two rows (latitudes) of four devices and oriented longitudinally so as to cover a significant zone of the frontal hemisphere around a subject's face. The screens of

the tablets are oriented towards the subject and provide controlled piece-wise continuous illumination for acquiring facial reflectance. The selfie cameras on the tablets are employed to capture facial photometric response due to illumination during a capture sequence. These cameras are typically lower in resolution so they mostly serve to provide data for multiview 3D reconstruction (e.g., using structure-from-motion (SFM) [153]). We also place five smartphones in the setup along the equatorial plane, one between each column of the tablets, and employ their high-resolution back cameras (zoom lens) for acquiring facial reflectance and photometric normals. The tablets and the smartphones are mounted on a desk using appropriate table mounts. In this work, we employed iPad Air devices (4th generation) for the tablets and iPhone 12 Pro devices as the smartphones in our setup. Other types of tablets and smartphones (e.g., Android based) could also be used instead for the setup.

The devices are all controlled in synchronization during a capture process where one device acts as the master and the other devices act as slaves and they wirelessly communicate with each other over WiFi. Once a capture command is initiated by an operator on the master device, the master broadcasts the start time for the capture process to all slave devices. All devices in the setup pool the global GPS clock-time from the GPS signal, and having received the capture instruction and start-time from the master device, then execute their own individual capture processes. Each device has a pre-built capture process that for the tablets includes illuminating a sequence of patterns from the device screen and simultaneously recording images using the selfie-camera, while for the smartphones involves capturing high-resolution images using the back zoom camera. All devices start and end their capture processes together in synchronized fashion in a few seconds due to pre-set capture/illumination timings. Afterwards, the acquired data from all devices is transferred to a single machine for off-line processing. The data transfer is also done wirelessly once initiated by an operator. The entire setup is highly portable since it consists of only the mobile devices and their table mounts. Hence, it can be easily moved from one location to another and quickly re-assembled for use without having to deal with any custom electronics/wiring.

4.3.2 Monitor-based Setup

Our second setup consists of four desktop LCD monitors that we mount together on a desk in portrait mode as shown in Fig. 4.1 (c). We employ four 27" 4K monitors with 16:9 aspect ratio (Asus ProArt PA279CV) in our setup. The monitors are arranged longitudinally facing the subject to cover a similar frontal hemispherical zone of directions with screen illumination. The four monitors are all controlled by single workstation (running Windows 10) via HDMI and display ports. The workstation includes a high-end graphics card (Nvidia RTX 3070 Ti) which is employed to drive the four monitors. We additionally mount stereo pairs of digital cameras within the vertical gaps between the monitors for multiview capture. Eight cameras are mounted using small desk-tripods and we chose mirrorless cameras with small form factor (Canon EOS M200) for ease of mounting in the setup. During acquisition, the workstation controls the monitors and the set of cameras in synchronization to rapidly capture a sequence of images under a set of controlled illumination patterns for 3D shape and reflectance capture. The monitor-based setup has the advantage over the tablet-based setup of much more fine-grained synchronization between multiple displays supporting much faster capture, even supporting video rate capture required for acquiring dynamic facial performance.

4.4 Illumination

4.4.1 Modulated Binary Illumination

We employ horizontally and vertically aligned binary illumination patterns in this work, over the hemispherical zone of illumination of the proposed capture setups, for acquiring albedo and photometric normals with diffuse-specular separation. While this is similar to the approach of Kampouris et al. [80], our patterns require additional form-factor modulation due to being near-field with limited angular extent. Furthermore, we make the crucial observation that over the zone of hemispherical illumination covered by the displays in our setup, we do *not* need all three axis aligned (X, Y, Z) binary illumination conditions and their complements employed by [80]. Specifically, when we illuminate a subject with the horizontally aligned binary pattern and its complement (H and H' respectively), the lit

portion of the zone does not have its centroid aligned with X-axis but is actually centered around $+45°$ and $-45°$ directions respectively in the XZ plane (with $0°$ corresponding to the +Z axis). This results in the complementary pair of horizontal binary patterns H and H' to have their centroids orthogonal to each other in the XZ plane (albeit with a $45°$ in-plane rotation), and hence sufficient for determining the x and z components of a photometric normal. We additionally only need measurement of the vertically aligned binary pattern V and its complement V' to determine the y component of the photometric normal. This reduces the number of measurements to only four photographs under the horizontal and vertical binary illumination patterns (see Fig. 4.3, a).

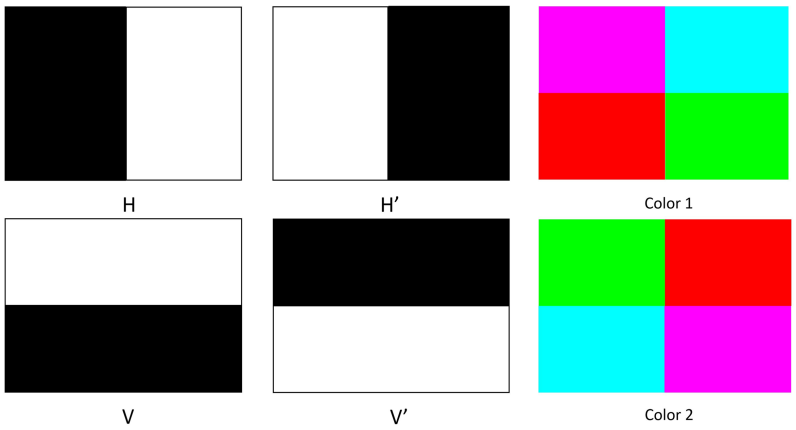
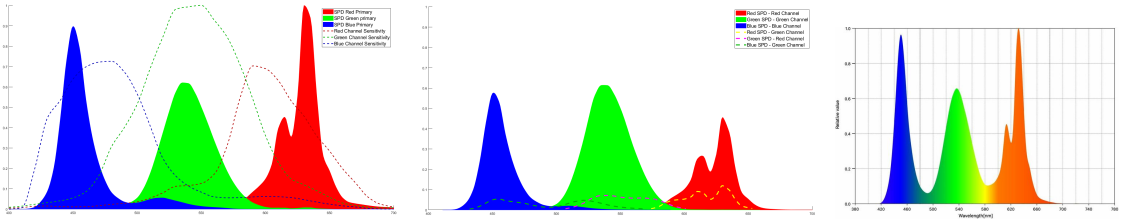


Figure 4.3: Four binary illumination patterns (H, H', V, V'), and two color-multiplexed patterns used for 2-shot acquisition.

4.4.2 Spectral Multiplexing and Cross-Talk

For faster acquisition, we further reduce the number of measurements to only two photographs by employing spectral multiplexing of the binary patterns using the R, G and B color channels of the display panels. While this is similar to the spectral multiplexing approach of [80], we demonstrate much higher quality results for faces with our spectral multiplexing scheme which are almost at par with results obtained using four measurements under binary white illumination. Unlike previous work employing spectral multiplexing for photometric measurements, we observe that typical RGB illumination observed by standard RGB cameras can suffer from a bit of spectral cross-talk between neighboring wavelengths. Specifically, depending on the spectral power distribution (SPD) of the RGB source and the spectral sensitivity of the camera, this can result in some cross-talk between the blue and green channels, or the green and red channels sensed by the camera. Both

our setups employ LCD displays with DCI P3 color gamut. We measured the SPD of our displays using a spectrometer (Sekonic SpectoMaster C700) and camera spectral sensitivity (using the procedure of [164]), and found the green and red illumination emitted by the displays to be spectrally closer to each other due to an amber peak in the red illumination spectrum. This causes the green channel of the cameras we employ to be sensitive to red illumination emitted by the displays, and hence causes some spectral cross-talk between these two color channels (Fig. 4.3(b)).



(a) Display SPD and camera sensitivity (b) Display SPD and spectral cross-talk (c) Display SPD of DCI P3

Figure 4.4: (a) SPD of display illumination overlayed with the spectral sensitivity of one of the color cameras (iPhone 12 Pro) employed for acquisition. (b) SPD of display illumination overlayed with spectral cross-talk in camera due to neighboring wavelength of spectral illuminations showing significant cross-talk between the green and red color channels. (c) Spectral power distribution (SPD) of DCI P3 gamut color LCD displays employed for both desktop capture setups.

The spectral power distribution (SPD) of the DCI P3 gamut color LCD panel displays that we employ in both the proposed setups is shown in Fig. 4.4. The color display SPD has three clear peaks corresponding to blue, green and red, although the red has a smaller amber peak as well. Fig. 4.4 plots the spectral sensitivity of one of the color cameras (iPhone 12 Pro) we employ for measurements overlaid on the SPD of the display illumination. We recover the camera sensitivity using measurements of a color chart under known display illumination spectrum using the procedure described in [164]. Fig. 4.4 also plots the magnitude of cross-talk in terms of a camera color channel being sensitive to illumination from a neighboring color channel. As can be seen, for our given pairs of display illumination SPD and camera sensitivity, there is significant spectral cross-talk between the green and red color channels.

In order to deal with this spectral cross-talk, we exploit our horizontal and vertical binary patterns to design two novel complementary spectral multiplexing patterns for high-quality photometric measurements. Specifically, we ensure that the patterns do not have any

overlap of green and red illumination and we spectrally encode the horizontal and vertical binary patterns and their complements in the two patterns shown in Fig. 4.3(a). Each pattern consists of both the horizontal pattern H and its complement H' encoded in the red and green channels respectively (or vice versa), while the blue channel encodes either the vertical pattern V or its complement V'. The two patterns are complementary and sum to uniform white illumination.

4.4.3 Form-Factor Modulation

As previously mentioned, we additionally need to modulate the intensity of the binary illumination patterns to compensate for the form-factor of the measurement setup. This is required since the illumination is restricted to a hemispherical zone and hence is not uniformly incident from all directions (unlike an LED sphere). Specifically, due to the limited extent of the zone in the vertical (latitudinal) direction compared to the horizontal (longitudinal) direction, we need to modulate the intensity along the vertical direction to create an even distribution of illumination. We do this by reducing the intensity nearer to the equator compared to latitudes near the top and bottom of the zone. This form-factor modulation also needs to account for any local effects of illumination due to the setup geometry. We pre-compute this vertical modulation function using inverse rendering a diffuse sphere lit by display panel arrangements of our two setups. We then apply this vertical modulation function to the binary illumination patterns during acquisition.

4.4.4 Multi-view Capture

The acquired patterns are axis-aligned with the global coordinates of the capture setup. We observe that the patterns form a steerable basis and rotate the corresponding pairs into the local coordinate frame of each camera before employing the photometric estimation process described in Sec. 4.5.

4.5 Method

4.5.1 Acquisition using White Illumination

Given the acquired set of photographs of a subject under the four binary patterns H, H', V, V' (Fig. 4.3), we first rotate the H and H' patterns around the Y axis by $\pm 45°$ to additionally obtain the axis aligned patterns X, X', and Z. Each binary-complement pair can be added to acquire a full-on observation F, e.g., $F = X + X'$. These observations can then be used to estimate photometric normals and separated diffuse and specular albedo.

Photometric Normals

The set of acquired binary patterns can be used to compute mixed photometric normals following [80]. However, due to incomplete hemispherical illumination (limited to $< \pm 45°$ latitudinal span) employed in our setups, this would result in skewed normals. Instead, we employ Singular Value Decomposition (SVD) to extract photometric normals with more correct global orientation. We directly stack the flattened H, H', V, V' measurements into a matrix $\mathbf{A} \in \mathbb{R}^{WH \times 4}$, where W, H are the width and height of each measurement. As shown by [165], the eigenvectors of the three largest eigenvalues of $\mathbf{A}\mathbf{A}^\top$ correspond to the normals components $\mathbf{N^X}, \mathbf{N^Y}, \mathbf{N^Z}$.

The orientation and order of the components of normals is arbitrary with SVD. Therefore we order them in the following way: We create a facial mask $\mathbf{M}$ by using a face detector [166] on F, and design a function $\mathcal{A}(\mathbf{N}, \mathbf{M})$, that aligns the components with the +X,+Y,+Z axes. Specifically, we fit for each component a linear regression on the X and Y axes, and assign each component to the axis with the highest absolute trend, while using the sign of the trend to orient the component. For N^Z, the regression is fit on half of the X axis. For an $\mathbf{USV}^\top$ SVD decomposition of $\mathbf{A}$ and $\mathbf{U}'$ including the first three vectors of $\mathbf{U}$:

$$\mathbf{N} = \mathcal{A}(\mathbf{U}', \mathbf{M}), \quad \mathbf{A} = \mathbf{USV}^\top \tag{4.1}$$

Diffuse Normals

Shorter-wavelength channels exhibit sharper normal details, because of the wavelength-dependent diffuse scattering of light [74]. Under white illumination, we observe that the different color channels share the same amount of specular reflectance, which is white, but different amounts of diffuse reflectance. We can therefore acquire pure diffuse signal using the spectral-differencing between the brightest (red) and darkest channels (blue) of each pattern, i.e., $X_D = X^R - X^B$. The pure diffuse patterns can be used with Eq. 4.1 to compute the diffuse normals N_D.

Specular Normals

We follow the observation of [80] that the mixed normals are a mixture of diffuse and specular normals. However, we employ a simpler empirical signal processing step for estimating specular normals from the mixture. Formulating mixed photometric normals $\mathbf{N}$ as a combination of the diffuse normals $\mathbf{N_D}$ and a portion of the specular signal $\alpha\mathbf{S}$, we separate the specular signal using a Gaussian high-pass filter $g()$. We empirically set $\alpha = 0.5$ for our results. To obtain the final specular normals $\mathbf{N_S}$, we add the separated specular signal to the diffuse normals and re-normalize the normals:

$$\mathbf{N_S} = \frac{\mathbf{N_D} + \frac{1}{\alpha}(\mathbf{N} - g(\mathbf{N}))}{|| \, \mathbf{N_D} + \frac{1}{\alpha}(\mathbf{N} - g(\mathbf{N})) \, ||} \tag{4.2}$$

Diffuse-Specular Albedo Separation

For separating the reflectance albedo into diffuse $\mathbf{A_D}$ and specular $\mathbf{A_S}$ components respectively, we employ the linear system in Eq. 4.3 originally proposed by [80] for color-multiplexed data. We find the linear system based separation to also be well suited for binary patterns with white illumination. Assuming the binary observation along X is brighter than its complement X', the linear system is expressed as:

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{X'} \end{bmatrix} = \begin{bmatrix} \mathbf{N_D}.\mathbf{x} & 1 \\ (1 - \mathbf{N_D}.\mathbf{x}) & 0 \end{bmatrix} \begin{bmatrix} A_D \\ A_S \end{bmatrix}, \tag{4.3}$$

, where $\mathbf{N_D}.\mathbf{x}$ is the x component of the diffuse normal scaled to $[0, 1]$. Eq. 4.3 can be solved separately for each binary-complement pair to acquire an estimate of $\mathbf{A_S}$. Instead of the median of three axis-aligned solutions proposed by Kampouris et al., we find the best estimate as the average $\mathbf{A_S}_{avg}$ of solutions for only the X and Y binary pairs. Finally, for computing the diffuse albedo $\mathbf{A_D}$, we solve the linear system independently for each color channel and then subtract the channel wise $\mathbf{A_S}_{avg}$ from full-on $\mathbf{F}$ to obtain $\mathbf{A_D}$.

4.5.2 Color-Multiplexed Illumination

We further reduce measurements to a two-shot capture process using color multiplexed binary patterns shown in Fig. 4.3. These patterns yield two spectral pairs of horizontally aligned patterns H_1, H_1', H_2, H_2' and one pair of vertically aligned patterns V, V'. Similarly to the white pattern captures, the orthogonal horizontal patterns can be rotated to create two sets of X and Y axis-aligned patterns. We then follow a similar approach as with white illumination, but with the following key differences:

Diffuse and Specular Normals

We acquire two horizontal pattern sets, using the green channel H_1, H_1' and the red channel H_2, H_2'. Photometric normals from shorter-wavelength channels exhibit sharper normal details [74]. Hence, we use the SVD method from Eq. 4.1 to estimate the mixed photometric normals $\mathbf{N}$ using the green-channel set H_1, H_1' which has substantial specular signal. Additionally, we use SVD to estimate the normals using the red-channel set H_2, H_2', which exhibits weaker specular signal, and treat these as the diffuse normals $\mathbf{N_D}$. Finally, we use Eq. 4.2 to calculate the specular normals given N and refined N_D.

Diffuse-Specular Separation

Similar to white illumination, we employ the linear system of Eq. 4.3 to estimate the specular albedo $A_{S_{avg}}$ by averaging the solutions for the X and Y axes. However, unlike the case of white illumination, to estimate the diffuse albedo $\mathbf{A_D}$ we first scale the average $\mathbf{A_S}_{avg}$ by the ratio r_c of the respective channel wise solutions of $\mathbf{A_S}$ relative to red channel

solution, before subtracting it from full-on image $\mathbf{F}$: $\mathbf{A_D} = \mathbf{F} - r_c * \mathbf{A_S}_{avg}$. Here, $\mathbf{F}$ is obtained by adding the two color-multiplexed patterns.

4.5.3 Specular Roughness Estimation

We further exploit the observations under the color-multiplexed patterns to additionally estimate spatially varying specular roughness. We make the observation that the two color patterns illuminate a subject with four saturated colors and increasing specular roughness blurs these colors to make them less saturated. Hence, we employ normalized color as a novel metric to estimate spatially varying specular roughness using an inverse rendering process. We employ the estimated diffuse albedo $\mathbf{A_D}$ and diffuse normal $\mathbf{N_D}$ to relight the subject under the two color multiplexed lighting conditions and subtract these rendered diffuse components from the photographs to isolate their specular components. We then employ the estimated specular albedo $\mathbf{A_S}$ and specular normal $\mathbf{N_S}$ to render the specular response to these lighting patterns and compare the normalized color of the renderings with various roughness parameters (Cook-Torrance BRDF) to the isolated specular component in the photographs. The estimated specular roughness corresponds to the best matching normalized color (L1 norm). This way, we exploit color as an additional cue to disambiguate specular albedo and roughness.

4.5.4 Dynamic Capture

The reflectance estimation approach using color-multiplexed patterns (Sec.4.5.2) is well-suited for dynamic capture. As proof-of-concept, we employ our monitor-based setup for dynamic capture and synchronize a machine vision camera (FLIR Grasshopper 3) with the refresh rate of the monitors. Using this setup, we capture alternating color-multiplexed patterns at 60 fps. The monitors take 16ms to update the pattern on the screens. In order to handle this, we double up the alternating patterns for two frames at a time and use every other frame in the sequence while ignoring the in-between frames corresponding to the screen updates. This provides an effective capture rate of 30 fps for the alternating patterns. Each pattern can be used in conjunction with the next pattern in the sequence for reflectance estimation at the effective capture rate.

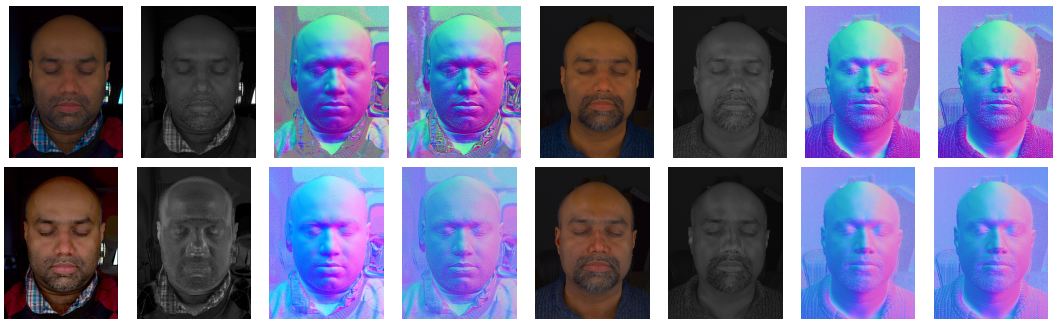
4.5.5 Base Geometry Acquisition

We reconstruct base geometry using multiview capture for our static capture examples. We employ widely used structure-from-motion software COLMAP [152, 153] for this purpose and provide it a full-on image $\mathbf{F}$ (computed as sum of a binary pair) from each camera viewpoint. Each camera's processed reflectance and normals are projected on to the reconstructed mesh.

4.6 Experiments

4.6.1 Evaluation

Fig. 4.5 presents comparison of reflectance and photometric normal maps estimated using our 4-shot method employing white binary illumination (top-left), and maps estimated using our 2-shot method using spectral multiplexing (top-right), vs those estimated using the 6-shot method (bottom-left) and the 2-shot method (bottom-right) proposed by Kampouris et al. [80]. Here, all methods have been implemented on our tablet-based capture setup. Both our 4-shot and 2-shot method achieve high quality results that are quite comparable, while achieving results that are superior to those obtained with the 6-shot and the 2-shot method respectively of [80] for both albedo and photometric normals. We note that the method of [80] was designed for an LED sphere and hence it does not perform as optimally in our setup. Finally, Fig. 4.13 shows a comparison of our reconstruction with a single-image learning-based method [24].



(i) $\mathbf{A}_D$ (j) $\mathbf{A}_S$ (k) $\mathbf{N}_D$ (l) $\mathbf{N}_S$ (m) $\mathbf{A}_D$ (n) $\mathbf{A}_S$ (o) $\mathbf{N}_D$ (p) $\mathbf{N}_S$

Figure 4.5: Comparison of our results (top) with 4-shot white illumination (left), and 2-shot spectral multiplexing (right), against that of [147] (bottom) using 6-shot white illumination (left) and 2-shot spectral multiplexing (right).

4.6.2 Static Capture

Fig. 4.6 presents qualitative comparisons of acquired reflectance and photometric normal maps for a subject using both our proposed desktop-based capture setups. As can be seen, both setups acquire high-quality data. Fig. 4.7 presents maps of spatially varying specular roughness of a few subjects that were estimated using novel color cues given observations of the subjects under the two color multiplexed illumination conditions.

Fig. 4.10 presents results of several subjects acquired using both of our proposed desktop-based capture setups. Here, we present results of our 2-shot capture process using spectral multiplexing. As can be seen, our estimated diffuse-specular separated reflectance and normals maps (c, d) enable camera-space renderings (b) that are a good qualitative match to the comparison photographs (a). The acquired data can be combined from multiple viewpoints for high quality renderings of 3D geometry (e), and appearance (f). In Fig. 4.11, we present additional results in high-resolution. Specifically, we show side-views of the captured photo, diffuse albedo, specular albedo, diffuse normals, specular normals and specular roughness, from two subjects from the main paper. These are acquired with our 2-shot method using the tablet-based setup. We show multiview data projected to UV maps in Fig. 4.8. We compare the quantitative error between the validation photos (a) and our camera-space renderings (b) and report an average MSE of 0.0019 and PSNR of 27.205 across 7 subjects.

Moreover, in Fig. 4.9 we present a comparison of our method and Ghosh et al. [76]. We acquire a diffuse albedo by fitting a cross-polarized filter on our cameras, and present it side-to-side with the mixed albedo and our diffuse albedo. We calculate a mean squared error of 0.0028 and a Peak Signal-to-Noise Ratio (PSNR) of 25.41. Our method is able to successfully to separate the diffuse signal, without having to rely on extra hardware, such as a polarized filter.

4.6.3 Dynamic Capture

Fig. 4.14 presents a few frames from a dynamic capture sequence of a female subject acquired at effective 30FPS (every other frame of a 60FPS capture process) using a machine vision camera. As can be seen, the monitor-based setup, in conjunction with our proposed



(a) $\mathbf{A}_D$ (b) $\mathbf{A}_S$ (c) $\mathbf{N}_D$ (d) $\mathbf{N}_S$

Figure 4.6: Comparison of reflectance separation between the Tablet-based setup (top), and the monitor-based setup (down).

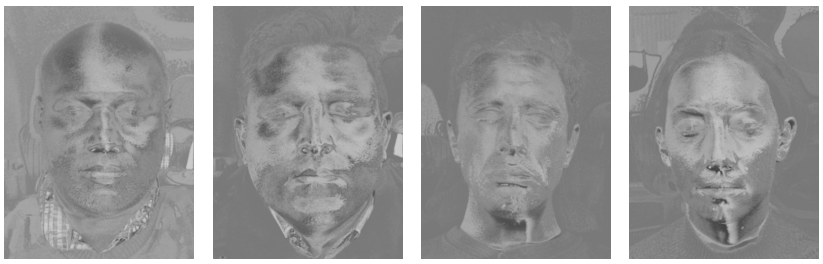


Figure 4.7: Spatially-varying specular roughness maps of subjects estimated using novel color cues from the color multiplexed patterns.

color multiplexed patterns, is well suited for obtaining video-rate diffuse-specular separated albedo maps and photometric normals of a dynamic capture sequence. We provide an additional example in the supplemental material and video.

4.7 Limitations

Our proposed capture technique, while highly practical, does have some imitations. The illumination is restricted to a frontal zone which, while well suited for faces, may not be suitable for all kinds of object shapes. Even for faces, there is limited illumination from above and below the face which can be a source of bias in the estimated photometric maps. The diffuse-specular separation method, like all color-space methods, does not work well for white/gray surfaces. We do not model any single scattering in skin which has been



(a) Diff. Alb (b) Spec. Alb (c) Sp. Norm

Figure 4.8: Completed facial UV maps of captured reflectance, which showcases the multi-view capabilities of our approach.



(a) Unpolarized (b) $\mathbf{D}_A$, [76] (c) $\mathbf{D}_A$, Ours

Figure 4.9: Diffuse albedo comparison using cross polarization [76] versus our method.

shown to have subtle effect on appearance [75]. The dynamic capture rate is limited by the refresh rate of the monitors, as well as the dynamic range of the sensor and the display panels.

4.8 Conclusions

We presented two novel desktop-based setups for high quality facial capture including reflectance and photometric normals. The setups are highly practical and scalable, consisting entirely of commodity components. We also present novel analysis of binary illumination in conjunction with our setup for efficient acquisition with reduced measurements while achieving high-quality diffuse-specular separation, particularly for the case of spectral multiplexing. We further exploit novel color cues in conjunction with color multiplexing for estimating spatially-varying specular roughness, and also demonstrate possibility of video rate dynamic appearance capture. Our proposed systems have the potential to make high-quality facial capture widely accessible for many applications.



(a) Photo (b) Render (c) $\mathbf{A}_D$ (d) $\mathbf{A}_S$ (e) $\mathbf{N}_S$ (f) Geom. (g) Render

Figure 4.10: Results of various subjects acquired with our proposed tablet-based setup (top two rows), and monitor-based setup (Bottom two rows). All results were acquired using our 2-shot spectral multiplexing method.



(a) Photo (b) Diff. Alb. (c) Sp. Albedo (d) Sp. Norm. (e) Roughness

Figure 4.11: Side-view high-resolution results of two subjects. All results were acquired using our 2-shot spectral multiplexing method and the tablet-based setup.



(a) $\mathbf{A}_D$ (b) $\mathbf{A}_S$ (c) $\mathbf{A}_D$ (d) $\mathbf{A}_S$

Figure 4.12: Comparison of albedo measurement and separation with our method using our setup (left), vs using a single screen (right).



(a) Input (b) Ours (c) AM

Figure 4.13: Our results compared with AvatarMe^{++}, a deep-learning method [24].

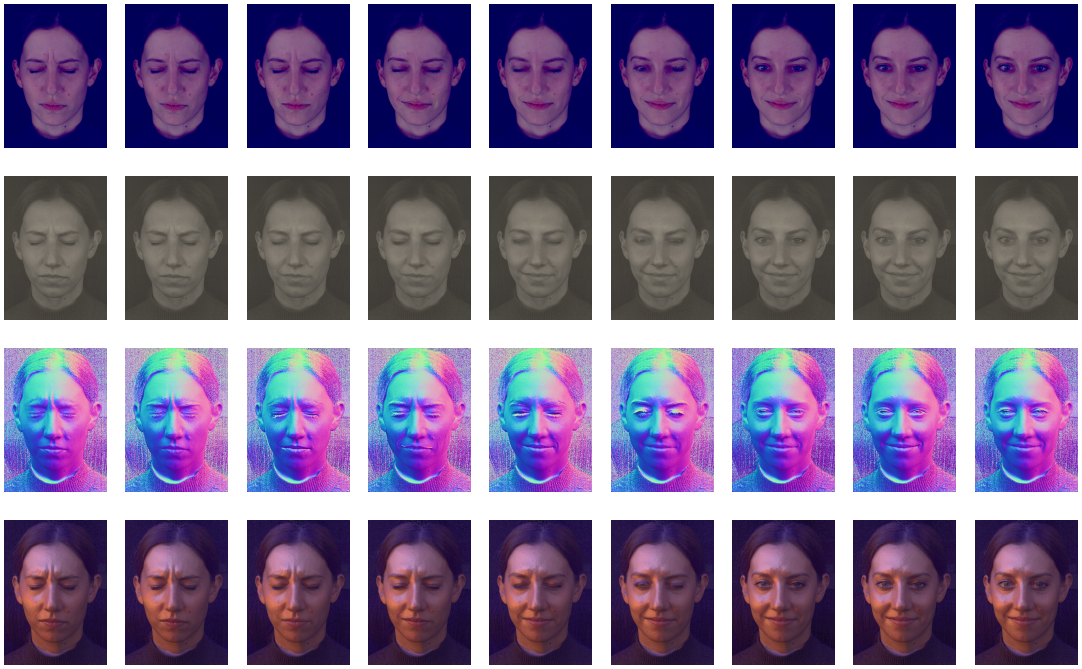


Figure 4.14: Example of dynamic capture using our monitor-based setup, acquired at 60 FPS for results at 30 FPS. From top-to-bottom: diffuse albedo, specular albedo, specular normals and rendering in the Grace Cathedral environment.

Chapter 5

Facial Appearance Reconstruction from a Single-Image

Contents

5.1 Introduction 79

5.2 Related Work 82

5.2.1 Facial Geometry and Reflectance Capture 82

5.2.2 Facial Geometry and Texture Estimation 83

5.2.3 Image-to-Image Translation 84

5.2.4 Differentiable Rendering 84

5.2.5 Facial BRDF Estimation 85

5.3 Data Acquisition 86

5.3.1 Training Data Capturing Setup 86

5.3.2 Data Augmentation 87

5.4 Method 88

5.4.1 Overview 88

5.4.2 Initial Geometry and Texture Estimation 89

5.4.3 3DMM Capturing Environment Estimation 90

5.4.4 Super Resolution 91

5.4.5 Reflectance Inference with AvatarMe 91

5.4.6 Photorealistic Differentiable Facial Rendering 95

5.4.7 Reflectance Inference with AvatarMe^{++} 99

5.5 Experiments 102
5.5.1 Implementation Details 102
5.5.2 Evaluation 104
5.5.3 Ablation 106
5.5.4 Limitations 108
5.6 Conclusion 109

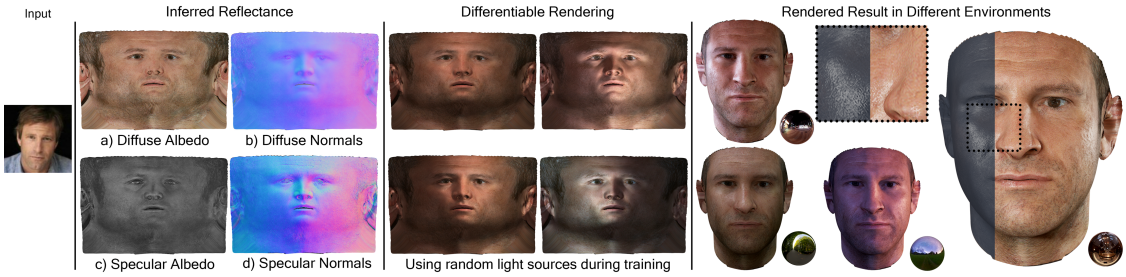


Figure 5.1: From left to right: Input image from LFW [167]; AvatarMe^{++} predicted reflectance (diffuse albedo, diffuse normals, specular albedo and specular normals); Rendered predictions with our photorealistic differentiable rendering; Rendered reconstruction in different environments.

Here we introduce AvatarMe, a deep-learning method for facial appearance reconstruction from arbitrary facial images, which was published in *A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "AvatarMe: Realistically renderable 3d facial reconstruction" in-the-wild"," in* Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 760–769, 2020.* Moreover, in this chapter we introduce the extension of the above method, AvatarMe^{++}. In contrast with AvatarMe, it uses photorealistic differentiable rendering to improve the training process and generalization of AvatarMe, requiring a smaller network and achieving successful reconstruction from arbitrary illuminations. AvatarMe^{++} was published in *A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, A. Ghosh, and S. Zafeiriou, "AvatarMe++: Facial shape and BRDF inference with photorealistic rendering-aware GANs,"* IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2021.*

Over the last years, with the advent of Generative Adversarial Networks (GANs), many face analysis tasks have accomplished astounding performance, with applications including,

but not limited to, face generation and 3D face reconstruction from a single "in-the-wild" image. Nevertheless, to the best of our knowledge, there is no method which can produce render-ready high-resolution 3D faces from "in-the-wild" images and this can be attributed to the: (a) scarcity of available data for training, and (b) lack of robust methodologies that can successfully be applied on very high-resolution data. In this paper, we introduce the first method that is able to reconstruct photorealistic render-ready 3D facial geometry and BRDF from a single "in-the-wild" image. To achieve this, we capture a large dataset of facial shape and reflectance, which we have made public. Moreover, we define a fast and photorealistic differentiable rendering methodology with accurate facial skin diffuse and specular reflection, self-occlusion and subsurface scattering approximation. With this, we train a network that disentangles the facial diffuse and specular reflectance components from a mesh and texture with baked illumination, scanned or reconstructed with a 3DMM fitting method. As we demonstrate in a series of qualitative and quantitative experiments, our method outperforms the existing arts by a significant margin and reconstructs authentic, 4K by 6K-resolution 3D faces from a single low-resolution image, that are ready to be rendered in various applications and bridge the uncanny valley.

5.1 Introduction

3D face reconstruction from a single image is one of the most popular and well-studied problems in the intersection of computer vision, graphics and machine learning. Apart from its countless applications, it demonstrates the power of recent developments in scanning, learning, and synthesizing 3D objects [6, 168]. Recently, mainly due to the advent of deep learning, tremendous progress has been made in 3D face reconstruction from images captured even in arbitrary recording conditions (also referred to as "in-the-wild") [19, 129, 132, 169]. Nevertheless, even though the geometry can be inferred somewhat accurately, in order to render a reconstructed face in arbitrary virtual environments, much more information than a 3D smooth geometry is required, i.e., skin reflectance as well as high-frequency surface normals. In this paper, we propose a meticulously designed pipeline for the reconstruction of high-resolution render-ready faces from "in-the-wild" images captured in arbitrary poses, lighting conditions, and occlusions. A result from our pipeline is showcased in Fig. 5.1.

The seminal work in the field is the 3D Morphable Model (3DMM) fitting algorithm [6]. The facial texture and shape that is reconstructed by the 3DMM algorithm always lies in a space that is spanned by a linear basis, which is learned by Principal Component Analysis (PCA). The linear basis, even though remarkable in representing the basic characteristics of the reconstructed face, fails in reconstructing high-frequency details in texture and geometry. Furthermore, the PCA model fails in representing the complex structure of facial texture captured in "in-the-wild" conditions. Therefore, 3DMM fitting usually fails in "in-the-wild" images. In the years that followed, 3DMM fitting was extended so that it could use a PCA model on robust features, i.e., Histogram of Oriented Gradients (HOGs) [170], for representing facial texture [9], with improved results in "in-the-wild" images. The recently proposed, Morphable Face Albedo model [11] additionally reconstructs diffuse and specular albedo with PCA. Nevertheless, these methods cannot reconstruct high-resolution facial textures. Finally, the non-linear facial and head albedo and normals models [171, 172], generate high-resolution facial textures but have not been shown in "in-the-wild" fitting.

With the advent of deep learning, many regression methods using an encoder-decoder structure have been proposed to infer 3D geometry, reflectance and illumination [19, 49, 129, 132, 138, 168, 169]. Some of the methods demonstrate that it is possible to reconstruct shape and texture, even in real-time on a CPU [168]. However, the methods [49, 129, 132, 168, 169] fail to reconstruct highly-detailed texture and shape, due to various factors such as the use of basic reflectance models (e.g., the Lambertian reflectance model), the use of synthetic data, or mesh-convolutions on colored meshes. Their results are not render-ready, and cannot be used directly in industrial rendering applications for photorealistic results. Furthermore, in many of the above methods, the reconstructed texture and shape lose many of the identity characteristics of the original image.

Arguably, the first generic method which demonstrated that it is possible to reconstruct high-quality texture and shape from single "in-the-wild" images is the recently proposed GANFIT method [19]. GANFIT can be described as an extension of the original 3DMM fitting strategy but with the following differences: (a) instead of a PCA texture model, it uses a GAN [14] trained on high-resolution UV-maps, and (b) in order to preserve the identity in the reconstructed texture and shape, it uses features from a state-of-the-art face

recognition network [135]. However, the reconstructed texture and shape is not render-ready due to (a) the texture containing baked illumination, and (b) not being able to reconstruct high-frequency normals or specular reflectance.

Inverse rendering of an "in-the-wild" image or 3DMM-reconstructed texture to acquire its shape and its Bidirectional Reflectance Distribution Function (BRDF) parameters is an ill-posed problem and hence statistical priors are needed. Numerous works have proposed the use differentiable rendering loss while reconstructing 3DMMs [19, 43, 49, 134, 173]. However none of them photorealistically render the reconstructed face and acquire its reflectance properties, but project the reconstructed shape and use a simplistic shading model. This is mostly because of the low availability in facial reflectance data, and the inherent challenges in differentiable rendering. Recent works in differentiable rendering [43, 174] enables us to capitalize on our large reflectance dataset to implement a fast photorealistic facial differentiable rendering framework and use it in reconstructing high-resolution facial shapes and BRDF.

Early attempts to infer photorealistic render-ready information from single "in-the-wild" images have been made in some works [138, 140, 141, 175]. Arguably, some of the results showcased in the above papers are of decent quality. Nevertheless, the methods do not generalize since they directly manipulate and augment the low-quality and potentially occluded input facial texture. As a result, the quality of the final reconstruction always depends on the input image. Even more, the employed 3D model may not be very representative, and a very small number of subjects (e.g., 25 [141], 122 [138]) were available for training for the high-frequency details of the face. While closest to our work, these approaches focus on easily creating a digital avatar rather than high-quality render-ready face reconstruction from "in-the-wild" images, which is the goal of our work.

We present an elaborate methodology for high-quality 3D facial geometry and reflectance reconstruction from a single "in-the-wild" image. In particular, we collect a big dataset of facial reflectance, and use an end-to-end reflectance inference network with a photorealistic differentiable rendering loss. Our method builds upon recent reconstruction methods (e.g., GANFIT [19]) and applies super-resolution and domain-adaption algorithms to GAN-generated generated high-quality facial textures. We show that this methodology is superior to the previous state-of-the-art (e.g. [138, 141]), who apply algorithms for high-

frequency estimation of the original input, which could be of low quality and are affected by environment illumination. We demonstrate that it is possible to produce render-ready faces from arbitrary faces (pose, occlusion, etc.) including paintings, which can be realistically re-rendered in any environment. Specifically, our contributions are:

- A dataset of facial reflectance and geometry collected using state-of-the-art methods from over 200 subjects, which is now available to the community.
- A differentiable rendering framework that fully utilizes both diffuse and specular reflectance data, and enables the fast approximation of subsurface scattering and occlusion shadows.
- An image-translation network that transforms a facial geometry with baked-in illumination to diffuse and specular albedo and normals using the above differentiable rendering framework.
- An end-to-end algorithm for reconstructing high-resolution 3D faces including their shape and BRDF, from a single "in-the-wild" image.

This work is an extension of AvatarMe[23]. Compared to the conference paper, AvatarMe^{++} adds: a) a photorealistic differentiable rendering method; b) online data augmentation of the training data in randomized illumination environments; c) a single image-translation network for BRDF inference, that utilizes a stochastic rendering loss, geometrical and global information; d) an extensive ablation study and experiments for the aforementioned additions.

5.2 Related Work

5.2.1 Facial Geometry and Reflectance Capture

Debevec et al. [2] first proposed to employ a specialized light stage setup to acquire a reflectance field of a human face for photorealistic image-based relighting. Weyrich et al. [176] used an LED sphere and 16 cameras to densely record facial reflectance and

computed view-independent estimates of facial reflectance from the acquired data, including per-pixel diffuse and specular albedos, and per-region specular roughness parameters. These initial works required cumbersome and impractical dense capturing.

Ma et al. [4] introduced polarized spherical gradient illumination (using an LED sphere) for efficient acquisition of separated diffuse and specular albedos and photometric normals of a face using just eight photographs. They demonstrated high quality facial geometry, including skin mesostructure as well as realistic rendering with the acquired data. However, the method was restricted to frontal viewpoint acquisition, as the polarization pattern used on the LED sphere was view-dependent. Subsequently, Ghosh et al. [5] extended polarized spherical gradient illumination for multi-view facial acquisition by employing two orthogonal spherical polarization patterns. This allows capturing separated diffuse and specular reflectance and photometric normals from any viewpoint around the equator of the LED sphere. Until today, it can be considered the state-of-the art in terms of high quality facial capture. In the recent years, significant progress has also been made in passive facial capture, from high quality facial geometry capture [91] to even detailed facial appearance estimation [177]. However, the quality of the acquired data with such methods is lower compared to active illumination techniques.

Recently, Kampouris et al. [62] demonstrated how to utilize unpolarized binary spherical gradient illumination for estimating separated diffuse and specular albedo and photometric normals using color-space analysis. The method does not require polarization and hence needs half the number of photographs compared to polarized spherical gradients. Moreover, it enables completely view-independent reflectance separation, making it faster and more robust for high quality facial capture [178]. For our work, we use the unpolarised active illumination-based multi-view facial capture method [62, 178] for acquiring high quality facial reflectance data in order to build our training data.

5.2.2 Facial Geometry and Texture Estimation

Over the years, numerous methods have been introduced in the literature that tackle the problem of 3D facial reconstruction from a single input image [6, 9, 109, 130, 179, 180, 181, 182]. Early methods required a statistical 3DMM both for shape and appearance,

usually encoded in a low dimensional space constructed by PCA [6, 9, 179]. A 3DMM is typically fitted on a 2D image using an energy based cost optimization with respect to the model's identity and expression parameters as well as the parameters of the camera and scene illumination, as thoroughly explained in the 3DMM review of [109]. Moreover, many approaches have tried to leverage the power of Convolutional Neural Networks (CNNs) to either regress the latent parameters of a PCA model [130, 180] or utilize a 3DMM to synthesize images and formulate an image-to-image translation problem using CNNs [181, 182]. Similar works have also modeled complete head topologies [121, 183, 184, 185]. Finally, the recent Morphable Face Albedo Model [11] separately models diffuse and specular albedo with a PCA model.

5.2.3 Image-to-Image Translation

Image-to-image translation refers to the task of translating an input image to a designated target domain (e.g., turning sketches into images, or day into night scenes). With the introduction of GANs [13], image-to-image translation improved dramatically [106, 108]. Recently, with the increasing capabilities in the hardware, image-to-image translation has also been successfully attempted in high-resolution data [18]. In this work, we improve on pix2pixHD [18], by building an image-translation network that incorporates the photo-realistic differentiable rendering of its results, so that a rendering loss can be introduced. Our model successfully learns to disentangle reflectance components from textures with rendered illumination and occlusion shadows.

5.2.4 Differentiable Rendering

Multiple works in the past have attempted to differentiate rendering models to solve inverse rendering problems with limited success [39]. OpenDR [41] was the first complete framework in Python for differentiable rendering, built using an auto-differentiation framework. Neural 3D Mesh Renderer [42] introduced a rasterization approximation which enables differentiation for non-occluded gradients. TF Mesh Renderer [43] introduced a differentiable rasterizer which interpolates per-vertex attributes in view-space, using positive and negative barycentric coordinates to overcome discontinuities. SoftRasterizer [44] introduced

a fully differentiable rendering framework, by using a probabilistic rasterization function with an aggregation function for z-buffering. It significantly improves the gradient flow over [41, 42] and can be used with differentiable local-illumination models. Similarly, [45] separate the fore- and background rasterization and use barycentric coordinates to propagate the gradient only for the foreground pixels. For global illumination models, Li et al. [46] introduced a differentiable ray tracer, which uses an edge-sampling-based method to provide a continuous rendering function and importance sampling to improve on performance. Moreover, Loubet et al. [47] introduced spherical rotations that remove the discontinuities, with respect to visibility, cameras, lights, and geometry.

Several complete frameworks exist that combine deep learning with differentiable rendering. TF Mesh Renderer [43] is integrated with Tensorflow, which also includes a library for graphics and differentiable rendering. Kaolin [48] is a library of Pytorch implementations including [44, 49]. Finally, Pytorch3D [174] is a complete modular differentiable rendering framework, based on SoftRasterizer, with additional modules for shading, performance, and compatibility improvements. To the best of our knowledge, our method is the first to show fast photorealistic differentiable rendering for human skin. We extend Pytorch3D framework [174] for accurate facial skin diffuse and specular reflection, self-occlusion and subsurface scattering approximation, and integrate it with a high-resolution image-translation GAN.

5.2.5 Facial BRDF Estimation

Many approaches have been successful in acquiring the reflectance of materials from a single image, using deep networks with an encoder-decoder architecture [143, 144, 145, 186]. However, they only explore planar surfaces in a constrained environment, typically assuming a single point-light source. Similar principles have also been successfully applied to "in-the-wild" outdoor images [187]. Early applications on human faces [49, 137] used image translation networks to infer facial reflection from an "in-the-wild" image, producing low-resolution results. Recent approaches attempt to incorporate additional facial normal and displacement mappings resulting in representations with high frequency details [138]. Although this method demonstrates impressive results in geometry inference, it tends to fail in conditions with challenging illumination and extreme head poses, and does not produce

re-lightable results. Saito et al. [175] proposed a deep learning approach for data-driven inference of high resolution facial texture map of an entire face for realistic rendering, using an input of a single low-resolution face image with partial facial coverage. This has been extended to inference of facial mesostructure, given a diffuse albedo [140], and even complete facial reflectance and displacement maps besides albedo texture, given a partial facial image as input [141]. The above methods are the most related to our work and achieve the creation of digital avatars from "in-the-wild" images. In this work, we show high quality facial reflectance reconstruction, from such images and existing models and datasets.

Various alternative paradigms that produce renderable human faces have been recently proposed. [188] introduce a coarse-to-fine optimization utilizing differentiable ray-tracing for facial geometry and albedo reconstruction. [189] reconstruct neural face reflectance fields that enable rendering with complex physical effects. Finally, [190] generate dynamic head textures and shapes with an encoder-decoder, that are relightable and animatable from VR-headset views.

The state-of-the-art facial 3DMM fitting method GANFIT [19] uses a GAN-generated texture with an iterative optimization method of the 3DMM's weights. Each iteration utilizes lambertian differentiable rendering and a deep face recognition network, achieving high quality texture with fine identity characteristics. In this work, we use an image-translation network that learns to disentangle the reflectance components reconstructed from a 3DMM fitting method, guided by our high-resolution photorealistic differentiable rendering loss. Our facial geometry and spatially-varying BRDF textures are high-resolution and ready to be rendered with high-quality photorealistic results.

5.3 Data Acquisition

5.3.1 Training Data Capturing Setup

We use our facial capturing system [191], comprising of an LED sphere with 168 lights (partitioned into two polarization banks) and 9 DSLR cameras. Half of the LEDs on the sphere are vertically polarized (for parallel polarization), and the other half are horizontally

polarized (for cross-polarization) in an interleaved pattern. On this setup, we can employ the state-of-the-art method of [5] for capturing high resolution pore-level reflectance maps of faces. On the same apparatus, we remove the polarizers and use the color-space analysis for diffuse-specular separation and multi-view facial capture [62, 178], to acquire reflectance of similar quality (Fig. 5.2). This is our preferred method, since it requires less than half of the data captured (hence reduced capture time) and provides a simpler setup without polarizers, enabling the acquisition of larger datasets.

Following the capture and diffuse-specular separation with [62], we develop a pipeline to prepare the training data as follows: A base geometry is reconstructed from the full-on images using structure-from-motion [153] and multi-view stereo [152]. The geometry is fitted with landmarks [192] using its rendering and registered to a template mesh [127]. Finally, the camera-space reflectance is projected to a uniform UV map, manually constructed for minimal distortion, at a resolution of $\hat{H}, \hat{W} = 6144, 4096$ pixels, and then cropped in overlapping patches of $512, 512$ pixels.

The dataset, RealFaceDB, is described in detail in Sec. 3.3, including the detailed processing pipeline, capturing process and demographics. Fig. 5.2 shows an example of a captured subject.



(a) $\mathbf{A}_D$ (b) $\mathbf{A}_S$ (c) $\mathbf{N}_D$ (d) $\mathbf{N}_S$ (e) $\mathbf{S}$

Figure 5.2: Example of a captured subject data using our pipeline, registered and projected to a standard UV topology.

5.3.2 Data Augmentation

In our initial method AvatarMe[23], we only rendered our dataset in the environment of the target 3DMM. In AvatarMe^{++} (Sec. 5.4.7), we can augment the training data by rendering them in random environments, centered around the target environment. This does not only improve the model's accuracy on the target environment, but enables it to successfully generalize to other domains (Figs. 5.11,5.13,5.14). Moreover, our captured dataset is imbalanced on race, due to the demographic limitations in our area and the

immobility of our capturing device. In an attempt to balance the dataset, we use the albedo measurements of [52] to augment our acquired albedos. Specifically, we use a patch from the forehead of the captured diffuse albedo, match it to the closest albedo from [52] and then apply a transformation to another albedo from their chart. Since all our albedos are in the same UV space, a manually constructed "skin" mask ensures that common non-skin areas remain unchanged.

5.4 Method

5.4.1 Overview

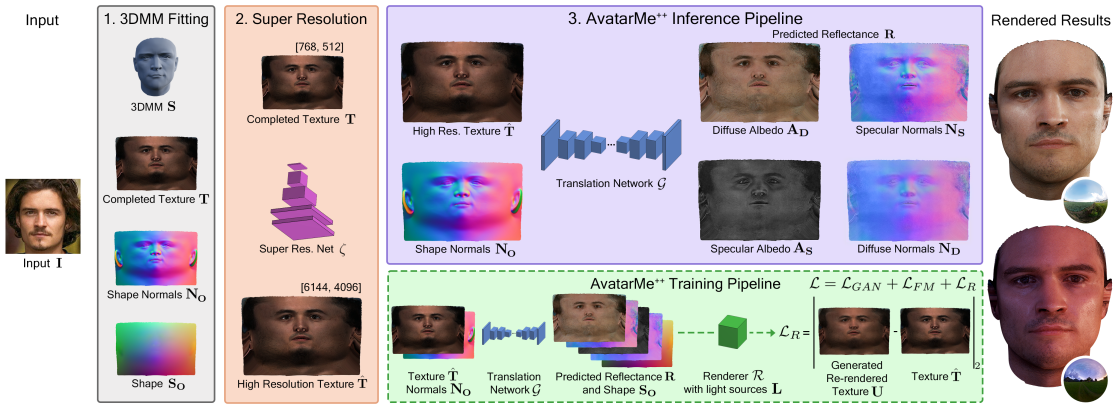


Figure 5.3: Summary of the AvatarMe^{++} method: Given an "in-the-wild" image $\mathbf{I}$, we first fit a 3D Morphable Model (3DMM) to acquire the shape $\mathbf{S}_O$, texture $\mathbf{T}$ and shape normals $\mathbf{N}_O$ in UV space. Then, we upscale the texture $\mathbf{T}$ using a state-of-the-art super resolution network ζ, trained on synthetic data rendered in the texture's $\mathbf{T}$ domain. A deep network $\mathcal{G}$ is then used to transform the upscaled texture $\hat{\mathbf{T}}$ and normals $\mathbf{N}_O$ to reflectance maps, namely the diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$, diffuse normals $\mathbf{N}_D$ and specular normals $\mathbf{N}_S$. The deep image-translation network is trained on high-resolution captured facial BRDF, which we have made public as RealFaceDB. To train AvatarMe^{++}, we define a photorealistic differentiable rendering module $\mathcal{R}$, with subsurface-scattering and self-occlusion approximation. During training, $\mathcal{R}$ is used to create synthetic data pairs, by rendering the captured data in the target's environment $\mathbf{L}$ and random ones. The loss $\mathcal{L}$ used during training, is comprised of an adversarial loss $\mathcal{L}_{GAN}$, a feature-matching loss $\mathcal{L}_{FM}$ and our photorealistic differentiable loss $\mathcal{L}_R$. The complete high resolution (up to $6\text{k} \times 4\text{k}$) BRDF maps can be used for photorealistic rendering, while the specular normals $\mathbf{N_S}$ can be used to enhance the 3DMM's geometry.

We aim to reconstruct the shape and reflectance properties of a subject from a single "in-the-wild" image, that can be used for photorealistic rendering. These are the shape $\mathbf{S}$, diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$, diffuse normals $\mathbf{N}_D$ and specular normals $\mathbf{N}_S$, facial reflectance components that can be used for photorealistic rendering (i.e. [4, 5]).

As shown in Fig. 5.3, we fit a 3DMM to an "in-the-wild" image (Sec. 5.4.2), obtaining a 3D facial geometry $\mathbf{S}$ with a texture $\mathbf{T}$, which is typically of low-resolution and contains baked-in illumination and shadows. We upsample $\mathbf{T}$ using a deep super-resolution network trained on textures of the same domain as the ones from the 3DMM (Sec. 5.4.4). Then, the AvatarMe models (Sec. 5.4.5) or the AvatarMe^{++} model (Sec. 5.4.7) transform the upsampled texture $\hat{\mathbf{T}}$ into the reflectance components $\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_D, \mathbf{N}_S$. AvatarMe utilizes four image-translation networks for the above transformation. AvatarMe^{++}, an extension to AvatarMe, uses a combined image-translation network and incorporates a photorealistic differentiable renderer (Sec. 5.4.6), achieving improved results, generalization and computational speed.

5.4.2 Initial Geometry and Texture Estimation

The first step of our method is the acquisition of a facial shape $\mathbf{S}$ and texture $\mathbf{T}$ from a single image. In our approach, we adopt the 3DMM-based fitting method GANFIT [19]. Apart from the usage of deep identity features, GANFIT synthesizes consistent realistic UV texture maps, using a GAN as a statistical representation of the facial texture. Alternatively, our method and training can easily be modified to use other methods (i.e. [45, 60]) as long as they produce a consistent shape and texture (results in Fig. 5.11). We reconstruct the initial base shape $\mathbf{S} \in \mathbb{R}^{n\times 3}$ of n vertices and texture $\mathbf{T} \in \mathbb{R}^{W\times H\times 3}$ from the input image $\mathbf{I}$ as follows:

$$\mathbf{T}, \mathbf{S} = \mathcal{F}(\mathbf{I}) \tag{5.1}$$

where $\mathcal{F} : \mathbb{R}^{k\times m\times 3} \mapsto \mathbb{R}^{W\times H\times 3}, \mathbb{R}^{n\times 3}$ denotes the GANFIT reconstruction method for an $\mathbf{I} \in \mathbb{R}^{k\times m\times 3}$ arbitrary sized image, and n number of vertices on a fixed topology.

The acquired shape is of adequate quality for rendering, however the texture $\mathbf{T}$ is of limited resolution and most importantly, contains significant baked-in illumination and self-shadows. We proceed by upsampling $\mathbf{T}$ in the next section, and then discuss the ways to learn the disentanglement of the baked-in illumination in $\mathbf{T}$, into high-resolution spatially-varying reflectance parameter UV maps.

5.4.3 3DMM Capturing Environment Estimation

A drawback of the texture modeled by typical 3DMMs is that they reproduce the environment conditions of their training data (i.e. reflection and shadows), which inhibits rendering. In our case, the textures generated by [19] contain sharp highlights and shadows, made by point-light sources, as well as environment illumination. In order to alleviate this problem, we model the illumination conditions of the dataset used in [19] and synthesize UV maps with the same illumination, in order to train a transformation between texture with baked-illumination $\mathbf{T}$ and reflectance maps $\mathbf{R}$.

Initially, we acquire random texture and mesh outputs from GANFIT, by fitting random facial images. Using a cornea model [193], we estimate the average view direction $\mathbf{v_d}$, the direction for the apparent 3 point light sources $\mathbf{l_d}$ and their intensity $\mathbf{c_d}$ used in the 3DMM texture data, including an environment color defined as $\mathbf{e_d}$. Then, we render our acquired subjects (Section 5.3), as if they were samples from the dataset used in the training of the 3DMM used, in our case [19]. In this way, we also have accurate ground truth of their reflectance. We compute a rendering ρ for each subject with reflectance $\mathbf{R} = \{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_D, \mathbf{N}_S\}$, directly in a UV map $\mathbf{T_d}$, using the predicted environment parameters. We denote this rendering process by $\rho : \mathbf{R}, \mathbf{c_d}, \mathbf{l_d}, \mathbf{v_d}, \mathbf{e_d} \mapsto \mathbf{T_d} \in \mathbb{R}^{\hat{W} \times \hat{W} \times 3}$ which renders the captured reflectance to the domain of the 3DMM textures with baked illumination.

The above estimation of the target 3DMM environment can be further improved, after an initial training of the AvatarMe^{++} network $\mathcal{G}$. We acquire a number of random 3DMM-generated textures in the target environment and use $\mathcal{G}$ to acquire their reflectance. Then, we initialize our differentiable renderer (Sec. 5.4.6) with the calculated parameters $(\mathbf{v_d}, \mathbf{l_d}, \mathbf{c_d})$. In an iterative process, we render the acquired reflectance maps in the current best environment parameters and compare the rendering with the initial 3DMM-generated textures, using an L1 rendering loss. Since the renderer is differentiable, we use gradient descent to further optimize the estimated parameters $\mathbf{v_d}, \mathbf{l_d}, \mathbf{c_d}$. Then, we can re-train $\mathcal{G}$ in the optimized environment estimation.

5.4.4 Super Resolution

Although the texture $\mathbf{T} \in \mathbb{R}^{W \times H \times 3}$ from GANFIT [19] has reasonably good quality and resolution ($H, W = 768, 512$) it is below par compared to artist-made render-ready 3D faces. On the contrary, the facial reflectance textures we capture in (Sec. 5.3.1) are in a resolution of $\hat{W}, \hat{H} = 6144, 4096$. Therefore, we train a state-of-the-art super resolution network, RCAN [194], that increases the resolution of $\mathbf{T} \in \mathbb{R}^{W \times H \times 3}$ to $\mathbf{T} \in \mathbb{R}^{\hat{W} \times \hat{H} \times 3}$, using $\times 4$ upsampling twice. We define the super-resolution network ($\zeta : \mathbb{R}^{H_p \times H_p \times 3} \mapsto \mathbb{R}^{\hat{H}_p \times \hat{H}_p \times 3}$), which is trained on square patches of $H_p = 64 \mapsto \hat{H}_p = 512$, given the large size of the results and the low number of available training data. At testing time, the whole texture from GANFIT $\mathbf{T}$ is up-scaled by the following:

$$\hat{\mathbf{T}} = \zeta(\mathbf{T}) \tag{5.2}$$

To train the super-resolution ζ to upsample the textures generated from Eq. 5.1, we use our estimation of GANFIT's illumination environment (Sec.5.4.3) to render our captured data with the same environment.

5.4.5 Reflectance Inference with AvatarMe

The significant issue of the texture $\mathbf{T}$ produced by typical 3DMMs is that they are trained on data with ambient illumination (i.e. reflection, shadows), which they reproduce. GANFIT-produced textures contain sharp highlights and shadows, made by strong point-light sources, as well as baked environment illumination, which prohibits photorealistic rendering. In order to alleviate this problem, we first model the illumination conditions of the dataset used in [19] and then synthesize UV maps with the same illumination $\mathbf{T_d}$ (Sec. 5.4.3). We can then use the pairs of $\mathbf{T_d}$ with the ground truth reflectance data ($\mathbf{A_D}, \mathbf{A_S}, \mathbf{N_D}, \mathbf{N_S}$) to train image-translation networks in a supervised manner. Finally, following [18], we formulate the networks' objective as:

$$\min_{G} \left(\max_{D} \mathcal{L}_{GAN}(G, D_k) + \lambda_{FM} \mathcal{L}_{FM}(G, D_k) \right) \tag{5.3}$$

where $\mathcal{L}_{GAN}(G, D_k)$ is the sum of adversarial loss and $\mathcal{L}_{FM}(G, D_k)$ is the feature matching loss, for all 3 discriminators of pix2pixHD [18]. The feature matching term is controlled by λ_{FM}.

We find that we can improve learning by giving the network an insight into the geometry of the reconstructed shape. In this manner, for each training data pair, we extract the shape $\mathbf{S_O}$ and its normals $\mathbf{N_O}$ in the same UV parameterization as the textures, and complete the 2D RGB texture, by using bilinear interpolation in the 2D UV space. Below we describe the baseline pipeline, *AvatarMe*, in which 4 image-translation networks are used, to first acquire the diffuse albedo $\mathbf{A_D}$ from the upsampled reconstructed texture $\hat{\mathbf{T}}$ and then the specular albedo $\mathbf{A_S}$, diffuse normals $\mathbf{N_D}$ and specular normals $\mathbf{N_S}$ from the diffuse albedo $\mathbf{A_D}$. The better performing pipeline *AvatarMe*$^{++}$, with a single rendering-aware network is described in Sec. 5.4.7.

Diffuse Albedo Extraction

We formulate de-lighting as a domain adaptation problem and train an image-to-image translation network. To do this, we follow two strategies different from the standard image translation approaches. Firstly, the shading and occlusion on the skin surface is geometry dependent and thus use both the texture and geometry of the 3DMM as input to the network. We find that this improves not only the network's accuracy, but also the consistency between patches. Instead of using the 3-channel shape texture $\mathbf{S_O}$, we define the 1-channel texture $\mathbf{D_O}$, that contains only the Z axis of $\mathbf{S_O}$. To do so, we concatenate the texture $\mathbf{T_d}$ with the UV map of the depth of the mesh in object space $\mathbf{D_O}$. We feed the network with a 4D tensor of $[\mathbf{T_{d_R}}, \mathbf{T_{d_G}}, \mathbf{T_{d_B}}, \mathbf{D_O}]$ and predict the resulting 3-channel albedo $\mathbf{A_D}$. Instead of $\mathbf{D_O}$, the shape normals ($\mathbf{N_O}$) can also be used. Secondly, we split the original high-resolution data into overlapping patches of $\hat{H}_p \times \hat{H}_p$ pixels, in order to augment the number of data samples and fit the data into the available GPU memory.

Therefore, in order to de-light $\hat{\mathbf{T}}$, and acquire the diffuse albedo $\mathbf{A_D}$, we train an image-to-image translation network $\mathcal{G}_{A_D} : \mathbf{T_d}, \mathbf{D_O} \mapsto \mathbf{A_D} \in \mathbb{R}^{\hat{H}_p \times \hat{H}_p \times 3}$ and then extract the diffuse

albedo $\mathbf{A_D}$ by the following:

$$\mathbf{A_D} = \mathcal{G}_{A_D}(\hat{\mathbf{T}}, \mathbf{D_O}) \tag{5.4}$$

Specular Albedo Extraction

Predicting the entire specular BRDF and the per-pixel specular roughness from the illuminated texture $\hat{\mathbf{T}}$ or the inferred diffuse albedo $\mathbf{A_D}$, poses an unnecessary challenge. As shown in [5, 62] a subject can be realistically rendered using only the intensity of the specular reflection (specular albedo) $\mathbf{A_S}$, which is consistent on a face due to the skin's refractive index. The spatial variation is correlated to facial skin structures such as skin pores, wrinkles, or hair, which are apparent in both the baked texture $\mathbf{T}$ and the diffuse albedo $\mathbf{A_D}$. Both can be used as input to the network, and we empirically found that our predicted high quality diffuse albedo $\mathbf{A_D}$ produces more accurate and consistent results. Therefore, having inferred $\mathbf{A_D}$ with $\mathcal{G}_{\mathcal{A}_\mathcal{D}}$, we infer the specular albedo $\mathbf{A_S}$ by a similar patch-based image-to-image translation network from the diffuse albedo ($\mathcal{G}_{A_S} : \mathbf{A_D} \mapsto \mathbf{A_S} \in \mathbb{R}^{\hat{H}_p \times \hat{H}_p \times 1}$):

$$\mathbf{A_S} = \mathcal{G}_{A_S}(\mathbf{A_D}) \tag{5.5}$$

Specular and Diffuse Normals Extraction

The specular normals exhibit sharp surface details, such as fine wrinkles and skin pores, and are challenging to estimate, as the appearance of some high-frequency details is dependent on the lighting conditions and viewpoint of the texture. Therefore, much detail may not be apparent in the input image or reconstructed texture. Previous works fail to predict high-frequency details [138], or rely on separating the mid- and high-frequency information in two separate maps, as a generator network may discard the high-frequency as noise [141]. Instead, we show that it is possible to employ an image-to-image translation network with feature matching loss [18] on a large high-resolution training dataset, which produces more detailed and accurate results.

Similarly to the specular albedo inference with $\mathcal{G}_{A_S}$, we feed the network with the predicted diffuse albedo $\mathbf{A_D}$. Using $\mathbf{A_D}$ instead of the 3DMM texture $\mathbf{T}$ produces more consistent

results. Even though $\mathbf{T}$ contains some specular highlights, these are always concentrated on a small subset of the image, since they're reconstructed using [19]. We can also luma-transform (in sRGB) the diffuse albedo to grayscale $\mathbf{A_D^{(gray)}}$, in order to reduce the number of channels. Moreover, the consistency of the results is greatly improved when also feeding the network with the 3DMM geometry, in this case, the shape normals. Finally, we also transform the shape normals $\mathbf{N_O}$ in tangent space $\mathbf{N_T}$, where the basis is a vector pointing to $[0, 0, 1]$. We find that in this multiple-network approach, the inferred specular normals details are better accentuated, when using both the input shape normals $\mathbf{N_T}$ and the predicted specular normals $\mathbf{N_S}$ in the tangent space.

Therefore, we train another image-translation network $\mathcal{G}_{N_S} : \mathbf{A_D^{gray}}, \mathbf{N_T} \mapsto \mathbf{N_S}, \in \mathbb{R}^{\hat{H}_p \times \hat{H}_p \times 3}$ to transform the concatenation of the grayscale diffuse albedo $\mathbf{A_D^{gray}}$ and the shape normals in tangent space $\mathbf{N_T}$ to the specular normals $\mathbf{N_S}$. The specular normals are extracted by the following:

$$\mathbf{N_S} = \mathcal{G}_{N_S}(\mathbf{A_D^{gray}}, \mathbf{N_T}) \tag{5.6}$$

The diffuse normals $\mathbf{N_D}$ are highly correlated with the 3DMM-reconstructed shape normals $\mathbf{N_O}$, as the evenly scattered light blurs most skin details. Similarly fpr $\mathcal{G}_{\mathcal{N}_\mathcal{S}}$, we train a network $\mathcal{G}_{N_D} : \mathbf{A_D^{gray}}, \mathbf{N_O} \mapsto \mathbf{N_D} \in \mathbb{R}^{\hat{H}_p \times \hat{H}_p \times 3}$ to map the concatenation of the grayscale diffuse albedo $\mathbf{A_D^{gray}}$ and the shape normals in object space $\mathbf{N_O}$ to the diffuse normals $\mathbf{N_D}$. The diffuse normals are extracted as:

$$\mathbf{N_D} = \mathcal{G}_{N_D}(\mathbf{A_D^{gray}}, \mathbf{N_O}) \tag{5.7}$$

Finally, the inferred specular normals can enhance the mesoscopic structure of the reconstructed geometry $\mathbf{S}$, by refining its features and adding plausible details. Based on [69], we integrate the specular normals in the tangent space $\mathbf{N_S}$ to produce a height UV map, which describes high-resolution per-pixel surface elevation. The height map can be then be embossed on a subdivided 3DMM-reconstructed geometry $\mathbf{S}$, to produce a higher-resolution shape.

5.4.6 Photorealistic Differentiable Facial Rendering

Here, we formulate a photorealistic differentiable rendering methodology, that can be incorporated in our image-translation networks during training and render the training data and results in different illumination environments.

Shading can be modeled as *local illumination*, which only models the surface reflection of light sources on objects and *global illumination*, which models light propagation in a scene, including indirect illumination, and produces more realistic results at a higher computational cost. We choose to rely on local illumination shading, since most such models are differentiable and much faster to compute than global illumination. Despite producing more realistic results, rendering high-resolution human skin with global illumination takes several minutes, and would be impractical to use while training a deep neural network like ours. There exist various local illumination models appropriate for rendering human skin, on which we capitalize on to compose the following methodology for photorealistic facial rendering. We achieve fast and differentiable photorealistic rendering, by using a local illumination model and approximations for self-occlusion and subsurface scattering. Additionally, ambient occlusion is inherently baked in the captured diffuse albedo and is thus reproduced during rendering.

Shading Model

We use Lambertian shading for the diffuse component $\mathbf{U}_D$ and Blinn-Phong [55] shading for the specular component $\mathbf{U}_S$, given their photorealistic results for human skin and their cheap computation. For the specular exponent s, we use a common spatially varying UV map. For a reflectance and shape set $(\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_D, \mathbf{N}_S, \mathbf{S})$, a camera with view direction $\mathbf{v}$ and a set of n_l light sources with $\mathbf{l_j}$ direction and $\mathbf{c_j}$ intensity and an ambient illumination intensity $\mathbf{c_a}$, we evaluate the shading for each pixel i as follows:

$$\mathbf{U_{D_i}} = \mathbf{c_a}\mathbf{A_{D_i}} \sum_{j=1}^{n_l} (\mathbf{N_{D_i}} \cdot \mathbf{l_j})\mathbf{c_j} \tag{5.8}$$

$$\mathbf{U_{S_i}} = \mathbf{A_{S_i}} \sum_{j=1}^{n_l} (\chi^+(\mathbf{N_{S_i}} \cdot \mathbf{h_j}))^s \mathbf{c_j}, \qquad \mathbf{h_j} = \frac{\mathbf{l_j} + \mathbf{v_j}}{||\mathbf{l_j} + \mathbf{v_j}||} \tag{5.9}$$

where $\chi^+(x)$ a piece-wise function that returns $\max\{0, x\}$, since negative angles between the normals and light source direction do not contribute to specular reflection.

Rendering Directly in UV Space

Rasterization, the process of transforming the geometrical shape and texture to pixels visible by a camera, is traditionally non-differentiable, and also computationally expensive. The various methods that have been proposed for differentiable rasterization are based on sub-optimal approximations [42, 44, 45] or are very expensive [46, 47]. This motivated our pipeline to completely avoid rasterization, by rendering directly on the UV space.

The geometry shape vertices $\mathbf{S}$ are projected and interpolated in the same UV space as the reflectance textures $\mathbf{S}_O$, creating texels with a one-to-one correspondence between normals, shape and reflectance pixels. Hence, each texel's $S_{O_{u,v}}$ are used to evaluate the view direction $\mathbf{v}$ used in Eq. 5.9. This way (a) is faster than using rasterization, (b) is differentiable, (c) can be used with small patches of a larger texture and shape and (d) creates a pixel-to-pixel correspondence between reflectance and rendering.

Fast Differentiable Facial Subsurface Scattering

Subsurface scattering (SSS) describes the light that exits a translucent medium at a different point from where it had entered. Human skin, a dielectric material, exhibits such properties and the travel distance can be further than that covered by the lambertian model (Eq.5.8). Subsurface scattering in the skin has a smoothing effect, with predominantly red color bleed and is required for the photorealistic rendering of skin [195]. These effects also vary across the skin and are stronger in more translucent areas such as the nose. Accurate SSS requires the expensive measurements of light transport, however we find that the following modifications to our renderer produce a photorealistic approximation that improves the results of our method.

A local illumination BRDF as described in Eq. 5.8 cannot model light scattered over large areas. However, the scattering occurring in human skin travels only a few millimeters and can be modeled by separately modeling the normals for diffuse reflection $\mathbf{N_D}$ [4]. We separately capture (Sec. 5.3.1) and infer both $\mathbf{N_D}$ and $\mathbf{N_S}$, which are then separately used

to evaluate the diffuse (Eq. 5.8) and specular (Eq. 5.9) components. Both normals are wavelength dependent [62] and we acquire $\mathbf{N_D}$ from the red channel and $\mathbf{N_S}$ from the blue channel of our captures. This method accurately models the SSS angular blur (Fig. 5.4) and does not impose a computational overhead during rendering or training.

Nevertheless, the above does not reproduce the spectrally-dependent spatial blur produced by SSS, which results in red-dominated color bleed and shadow smoothing. These can be accurately approximated by texture-space SSS [195], which blurs the diffuse component $\mathbf{U_D}$ under multiple kernels, based on the wavelength associated with the R, G, B channels. [195] uses the weighted combinations per channel for 6 different kernels, which is too expensive for our training requirements. In line with [196], we find that a single kernel is adequate and much faster. Empirically, for a gaussian filter $g()$, we calculate the mean kernel at $k = 1.4mm$ or 21 pixels in our standard topology and define a weighted combination of $\mathbf{w_D} = diag(0.5, 0.85, 0.95)$ for the diffuse component and $\mathbf{w_{SSS}} = diag(0.5, 0.15, 0.05)$ for the subsurface scattering. We use a manually created standard translucency map $\mathbf{C}$, which describes the amount of light absorbed and scattered on different facial areas. Finally, since we are using N_D, we localize this effect only on the darker areas, by multiplying the translucency map with an inverse brightness mask. Therefore, the diffuse component $\mathbf{U_D}$ with subsurface scattering $\mathcal{S}$ is defined as:

$$\mathcal{S}(\mathbf{U_{D_i}}) = (\mathbf{1} - \mathbf{C'_i}) \circ \mathbf{w_D}\mathbf{U_{D_i}} + \mathbf{C'_i} \circ \mathbf{w_{SSS}}\ g(\mathbf{U_{D_i}}) \tag{5.10}$$

$$\mathbf{C'} = \mathbf{C} \circ \left(\mathbf{1} - \sum_{j=1}^{n_L} (\mathbf{N_D} \cdot \mathbf{l}_j)\mathbf{c_j}\right)$$

where $\circ$ is the Hadamard product. As shown in Fig. 5.4, the usage of both SSS methods provides realistic results, with the minimum computational overhead.

Differentiable Shadows Simulation

The rendering framework so far does not include self-occlusion shadows, whose computation entails several challenges. Pytorch3D [174] does not support self-occlusion tracing, and efficient local illumination models, such as Blinn-Phong, inherently do not model it. On the other hand, differentiable global illumination algorithms [46, 47], that compensate for self-occlusion, are too expensive while training. Finally, the patches being rendered are

often unaware of the geometry that causes self-occlusion, as it may appear on other patches (i.e. patches of nose and cheek).



(a) Pytorch3D, with $\mathbf{S}, \mathbf{A_D}, \mathbf{N}$ (b) Blinn-Phong, using $\mathbf{S}, \mathbf{A_D}, \mathbf{A_S}, \mathbf{N_S}$ (c) b) with Subsurface Scattering (d) c) with self-occlusion AE

Figure 5.4: The impact of our rendering modifications, to the Pytorch3D Mesh Renderer [174]. Top row: rasterized mesh with rendered texture, bottom: detail. Our improvements in realisticity, also improve the network's ability to recover the albedo and specular highlights.

As an efficient and simple solution, we propose a simple autoencoder network, that generates self-occlusion UV maps based on light source direction and intensity. We train the autoencoder on UV maps baked only with self-occlusion. We also train a linear regressor that maps light sources features $\mathbf{L}$ to the autoencoder's learned latent features $\mathbf{H}_{\mathcal{O}}$. For each training example, we create a set of n_l random light sources, with direction $\mathbf{l_j}$ and luminosity $\mathbf{c_{l_j}}$, which we stack in a matrix $\mathbf{L} = [\mathbf{l_j} \ \dots \ \mathbf{l_{N_l}} \ \mathbf{c_{l_j}} \ \dots \ \mathbf{c_{l_{n_l}}}]^\top$. We acquire self-occlusion UV-maps using a global illumination method, at the same topology of our main dataset, for the mean geometry of our dataset. We then pre-train the autoencoder and regressor on these data. On each rendering step of the main network training, the rendering environment's light source features $\mathbf{L}$ are regressed to the latent space of the autoencoder, using the learned weights $\mathbf{W}_{\mathcal{O}}$ from which the decoder $\mathcal{O}(\mathbf{L}\mathbf{W}_{\mathcal{O}})$ generates the self-occlusion map. The result is multiplied with the rendered diffuse and specular components, to produce the final rendering:

$$\mathcal{R}(\mathbf{R}, \mathbf{L}) = \mathcal{O}(\mathbf{L}\mathbf{W}_{\mathcal{O}}) \circ (\mathcal{S}(\mathbf{U_D}) + \mathbf{U_S}) \tag{5.11}$$



(a) Generated (b) Truth (c) Generated (d) Truth

Figure 5.5: Prediction and ground truth for our self-occlusion autoencoder $\mathcal{O}$ (Sec. 5.4.6) for randomly sampled sets of 3 light sources, as input.

Self-occlusion on human faces (i.e. Fig. 5.5) does not exhibit sharp edges and is similar between different facial geometries. Therefore, we make the following simplifications: a) We use self-occlusion UV maps rendered from our dataset's mean shape, which enables the decoder $\mathcal{O}$ to correctly learn a meaningful latent representation, b) We train the autoencoder on low-resolution inputs and upsample the needed cropped patch. Thus, we have fast and differentiable self-occlusion generation during training, with a minimal footprint, so that our main network can learn to remove it.

5.4.7 Reflectance Inference with AvatarMe^{++}

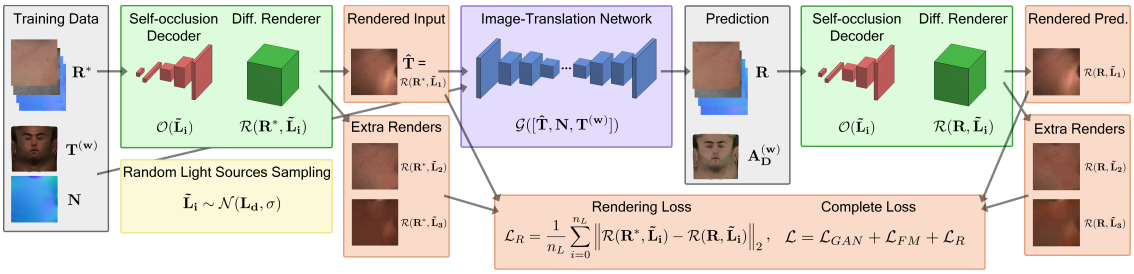


Figure 5.6: AvatarMe^{++} training methodology: For each iteration we render reflectance patches $\mathbf{R}^* = [\mathbf{A_D}, \mathbf{A_S}, \mathbf{N_D}, \mathbf{N_S}]$ from the captured data (RealFaceDB), using our differentiable renderer $\mathcal{R}$ (Sec. 5.4.6) and occlusion autoencoder (Sec. 5.4.6). The rendering parameters $\tilde{\mathbf{L}}_\mathbf{i} \sim \mathcal{N}(\mathbf{L_d}, \sigma), \quad i = 1 \dots n_L$ are sampled from a distribution with mean the target 3DMM environment $\mathbf{L_d}$. We pass one rendered patch $\hat{\mathbf{T}} = \mathcal{R}(\mathbf{R}, \tilde{\mathbf{L}}_\mathbf{i})$ to our main network $\mathcal{G}$ (Sec. 5.4.7) and produce the reflectance patches $\mathbf{R}$. The training $\mathbf{R}^*$ and generated $\mathbf{R}$ reflectance patches are used for the adversarial loss $\mathcal{L}_{GAN}$ and feature-matching loss $\mathcal{L}_{FM}$. Moreover, the consistency of the predicted patches is improved by including the down-scaled input texture $\mathbf{T}^{(\mathbf{w})}$ in $\mathcal{G}$'s input, and the down-scaled diffuse albedo $\mathbf{A}_\mathbf{D}^{(\mathbf{w})}$ in the $\mathcal{G}$'s target. Additionally, we render the training and predicted data, with each $\tilde{\mathbf{L}}_\mathbf{i}$ and define the rendering loss $\mathcal{L}_R$, as the average loss for each random environment.

Here, we introduce a single rendering-aware image-translation network which jointly generates all reflectance components $(\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_D, \mathbf{N}_S)$ from the upsampled 3DMM texture $\hat{\mathbf{T}}$. The motivation behind a single network is three-fold: jointly generating the above reflectance components: a) enables the introduction of a stochastic photorealistic render-

ing loss, which we show that improves the accuracy and generalization of the network, b) enables the network to share learned parameters between components, decreasing the size and memory requirement of the network, c) reduces the required inference time, since only a single forward pass is required. Fig. 5.6 shows an overview of this approach. In this manner, we introduce AvatarMe^{++} with a novel fast photorealistic differentiable rendering methodology (Sec. 5.4.6), an updated model architecture (Sec. 5.4.7) and a stochastic rendering loss (Sec. 5.4.7) which greatly improves the the results of AvatarMe.

Combined Image-to-Image Translation Model

We formulate the reflectance acquisition problem, as a domain adaptation problem and train a single image-translation network, The network $\mathcal{G}$ learns an inverse rendering function, on a UV texture $\hat{\mathbf{T}}$ with baked illumination.

An important challenge of this patch-based inference is producing consistent patches (especially of diffuse albedo $\mathbf{A_D}$) that can be seamlessly stitched together. We find that we can alleviate this issue by including the whole texture $\mathbf{T^{(w)}} = \mathbf{T}_{\downarrow_{\hat{H}_p \times \hat{H}_p}}$ to $\mathcal{G}$'s input and the diffuse albedo $\mathbf{A_D^{(w)}} = \mathbf{A_D}_{\downarrow_{\hat{H}_p \times \hat{H}_p}}$ to $\mathcal{G}$'s output, both downsampled ($\downarrow$) to the same size as the training patches. By including $\mathbf{T^{(w)}}, \mathbf{A_D^{(w)}}$ in the adversarial loss, we show that the network can learn the albedo color from the $\mathbf{T^{(w)}}$ and apply it when generating the high-resolution albedo patches.

The input to the generator $\mathcal{G}$ is formulated as the 9D tensor $\hat{\mathbf{T}}^+ = [\hat{\mathbf{T}}, \mathbf{N_O}, \hat{\mathbf{T}}^{(\mathbf{w})}] \in \mathcal{R}^{\hat{W} \times \hat{H} \times 9}$. The output is a 13D tensor $\mathbf{R}^+ = [\mathbf{A_D}, \mathbf{A_S}, \mathbf{N_D}, \mathbf{N_S}, \mathbf{A_D^{(w)}}] \in \mathcal{R}^{\hat{W} \times \hat{H} \times 13}$. Due to the resolution of our textures, we train the network on randomized $\hat{H}_p \times \hat{H}_p$ patches of $\hat{\mathbf{T}}^+$ and $\mathbf{R}$. $\mathbf{A_D^{(w)}}$ is only used for the adversarial loss and ignored at testing. Therefore, the reflectance $\mathbf{R}$ is extracted by the following:

$$\mathbf{R} = [\mathbf{A_D}, \mathbf{A_S}, \mathbf{N_D}, \mathbf{N_S}] = \mathcal{G}([\hat{\mathbf{T}}, \mathbf{N}, \hat{\mathbf{T}}^{(\mathbf{w})}]) \tag{5.12}$$

Stochastic Rendering Loss

So far, we have described a single-network $\mathcal{G}$ that generates the facial reflectance (Sec. 5.4.7) and a fast photorealistic differentiable rendering method (Sec. 5.4.6). Therefore, we can

now introduce a rendering loss in the training of $\mathcal{G}$, where the input texture $\mathbf{T_d}$ is compared with the predicted reflectance, rendered in the domain $\mathbf{L_d}$ of the input texture. The photorealistic rendering loss is defined as $\mathcal{L}_R = \|\mathbf{T_d} - \mathcal{R}(\mathbf{R}, \mathbf{L_d})\|_2$, using Eq. 5.11. Please note that $\mathcal{R}$ also uses the shape $\mathbf{S_O}$ which remains static during training and we omit it to avoid cluttering. A similar approach has been proposed for planar surfaces under flash illumination in [143, 144]. To the best of our knowledge, this is the first attempt on facial BRDF and the first included in a GAN-based image-translation network. Other facial acquisition methods (i.e. [19, 43, 49, 131, 137]) that use a differentiable rendering loss, merely re-project the inferred texture or use the unrealistic Lambertian model.

The introduction of the above rendering loss leaves the network unaware of specular features across the whole face, while it motivates the network to include shading elements in the diffuse albedo, which could be accurately reproduced in the rendering. This is mainly because all the training data are rendered in the same domain of the 3DMM textures used (Sec. 5.4.3) and the network can always expect shadows and highlights at the same places and intensities. Therefore we take 2 steps to introduce stochasticity to our training data, similar to [144], which improves our network's accuracy and generalization outside the target 3DMM domain.

Firstly, we sample random variations of the estimated target environment parameters $\tilde{\mathbf{L}} \sim \mathcal{N}(\mathbf{L_d}, \sigma)$ in each training iteration. We use it to render both the input to the network $\tilde{\mathbf{T}} = \mathcal{R}(\mathbf{R}^*, \tilde{\mathbf{L}})$ and the predicted reflectance $\mathcal{R}(\mathbf{R}, \tilde{\mathbf{L}})$ for the rendering loss. Secondly, for each training iteration, we sample additional n_L environment parameters, $\tilde{\mathbf{L}}_\mathbf{i} \sim \mathcal{N}(\mathbf{L_d}, \sigma), i = 1 \dots n_L$ which are not fed to the network, but are only used to compute an average rendering loss for these environments. In this manner, we stochastically approximate all light source directions within the allowed variation, while also penalizing the network for over-fitting the target 3DMM environment. Then, for the ground truth $\mathbf{R}^*$ and the inferred reflectance $\mathbf{R}$, for each training iteration, the rendering loss $\mathcal{L}_R$ is defined as:

$$\mathcal{L}_R = \frac{1}{n_L} \sum_{i=0}^{n_L} \left\| \mathcal{R}(\mathbf{R}^*, \tilde{\mathbf{L}}_\mathbf{i}) - \mathcal{R}(\mathbf{R}, \tilde{\mathbf{L}}_\mathbf{i}) \right\|_2, \tilde{\mathbf{L}}_\mathbf{i} \sim \mathcal{N}(\mathbf{L_d}, \sigma) \tag{5.13}$$

Overall, the objective of our image-translation network, which is based on pix2pixHD [18]

is defined as:

$$\min_{G}\left(\max_{D}\mathcal{L}_{GAN}(G,D_k)+\lambda_{FM}\mathcal{L}_{FM}(G,D_k)+\lambda_{R}\mathcal{L}_{R}\right) \tag{5.14}$$

where $\mathcal{L}_{GAN}(G,D_k)$ is the sum of adversarial loss and $\mathcal{L}_{FM}(G,D_k)$ is the feature matching loss, for all 3 discriminators of pix2pixHD [18]. The feature matching and rendering loss terms are controlled by λ_{FM} and λ_{R}.

5.5 Experiments

5.5.1 Implementation Details

The task of disentangling the diffuse and specular components, from a given input image with baked illumination can be formulated as an image-to-image translation problem. Nevertheless, as discussed previously: (a) our captured data are of very high-resolution (more than 4K) and thus cannot be used for training "as-is', due to hardware limitations (note not even on a 32GB GPU we can fit such high-resolution data in their original format), (b) pix2pixHD [18] takes into account only the texture and optionally labels, and thus geometric details, in the form of the shape and shape normals cannot be exploited to improve the quality of the generated diffuse and specular components.

Patch-Based Image-to-Image Translation

To alleviate the aforementioned shortcomings, we split the original high-resolution data into smaller patches of $\hat{H}_p \times \hat{H}_p$ size. More specifically, using a stride of size 256, we derive the partially overlapping patches by passing through each original UV horizontally as well as vertically. For each translation task we utilize the shape or shape normals, projected and interpolated in the same UV parameterisation as the textures. This increases the accuracy and level of detail in the derived outputs as the geometry act as a "guide" to the network. Finally, we downsample the whole input texture to the patch size and include it as well, as we find it greatly improves the consistency of the predicted patches. For the AvatarMe^{++} pipeline, we concatenate them channel-wise with the texture input and thus feed to the network a 9D tensor comprising of $\hat{\mathbf{T}}, \mathbf{N_O}, \hat{\mathbf{T}}^{(w)}$ and generate a 13D tensor comprising of

$\mathbf{A_D}, \mathbf{A_S}, \mathbf{N_D}, \mathbf{N_S}$ (Eq. 5.12) and $\hat{\mathbf{T}}^{(w)}$, which is discarded. During inference, that patch size can be larger (e.g. $1k \times 1k$), since the network is fully-convolutional.

Training Setup

To train RCAN [194], we use the default hyper-parameters. For the rest of the translation of models, we use a custom translation network as described earlier, which is based on pix2pixHD [18]. More specifically, we use 9 and 3 residual blocks in the global and local generators, respectively. The learning rate we used is 0.0002, whereas the Adam betas are 0.5 for β_1 and 0.999 for β_2. In our best model, we use a feature matching loss controller of $\lambda_{FM} = 10.0$ and rendering loss controller of $\lambda_R = 0.3$, for which perform an ablation study in the following section. Finally, we use a variable number of input and outputs as $N-$dimensional tensors, based on the implementation (Sec. 5.4.5, 5.4.5, or 5.4.7). As mentioned earlier, this substantially improves the results by accentuating the details and enforcing patch consistency.

Rendering Setup

To implement the facial photorealistic differentiable rendering (Sec. 5.4.6), we extend the recently published PyTorch3D [174] (version 0.3.0), for its speed, easily modifiable modular design and its compatibility with our image-translation network. Specifically, we fully integrate it with the generator and discriminator networks of our image-translation network, implement objects for multiple reflectance textures and implement a texture-space shader, that uses our framework from Sec. 5.4.6. For the shader's parameters, i.e. shininess exponent and translucency masks, we use a single manually created UV map with spatially varying values.

For the self-occlusion prediction, we train an autoencoder and a linear regressor that maps light source features to the autoencoder's latent values. The encoder and decoder consist of 5 convolutional blocks, each block having 2 convolutional layers with ELU [197], batch normalization [101] and a down- or up-sampling layer. The hidden layer has 64 features and connects to the encoder and decoder with a fully connected layer of 256 features.

5.5.2 Evaluation

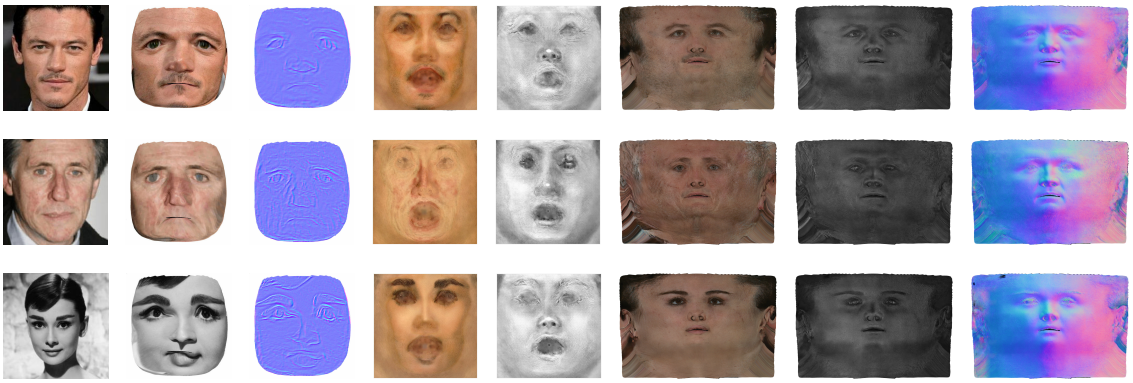


Figure 5.7: Reflectance maps produced by our method AvatarMe^{++}, against state-of-the-art methods. Reconstructions of [141] are provided by the authors and [138] are acquired using their open-sourced models.

To evaluate our reconstruction pipeline, we compare reconstructed reflectance maps and renderings acquired with AvatarMe^{++}, with ground truth data captured in a similar manner as our dataset RealFaceDB, the digital Emily Project [198] and current state-of-the-art. We use the Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) [199]. In Table 5.1, we conduct quantitative comparisons against the state-of-the-art [45, 141]. As can be seen, our method outperforms [138] and [141] by a significant margin. All meshes were manually registered to the same topology and UV parameterization. Moreover using a state-of-the-art face recognition algorithm [135], we also find the highest match of facial identity compared to the input images when using our method. The input images were compared against renderings of the faces with reconstructed geometry and reflectance, including eyes added manually to [141]. We also present qualitative comparisons in Fig. 5.7 and Fig. 5.8.

Table 5.1: Quantitative comparisons with state-of-the-art, between 6 reconstructions of the same subject, from different "in-the-wild" images, and ground truth using [178]. We transform [138, 141] results to our UV topology and compute only for a $2K \times 2K$ centered crop, as they only produce the frontal part of the face and manually add eyes to [141].

Algorithm	[141]	[138]	AvatarMe	AvatarMe^{++}
PSNR (Albedo)	11.225	14.374	24.05	**26.18**
PSNR (Normals)	21.889	17.321	26.97	**27.12**
MSE (Albedo)	0.0225	0.0140	0.0049	**0.0038**
MSE (Normals)	0.0047	0.0049	0.0031	**0.0025**
Rendered ID Score [135]	0.629	0.632	0.873	**0.881**

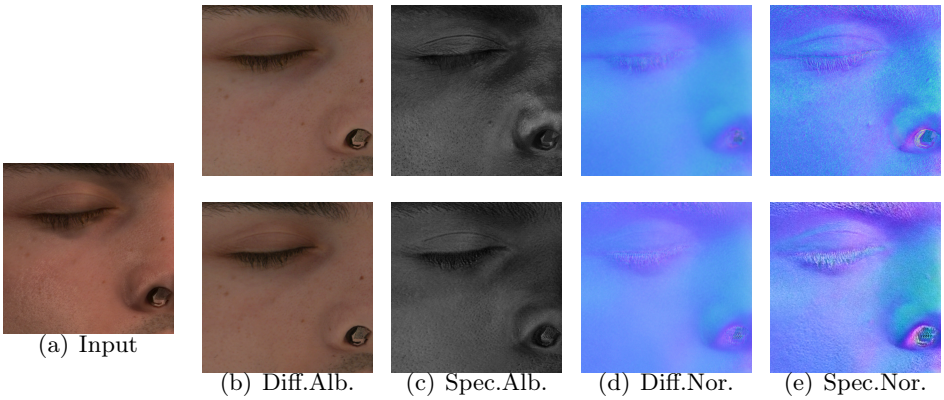


(a) Input (b) Chen et al. [138] (c) Yamag. et al. [141] (d) Ours, AvatarMe^{++}

Figure 5.8: Qualitative comparison of rendered reconstructions from a frontal and challenging side image. [141] results are provided by the authors and [138] results are acquired from their open-sourced models.

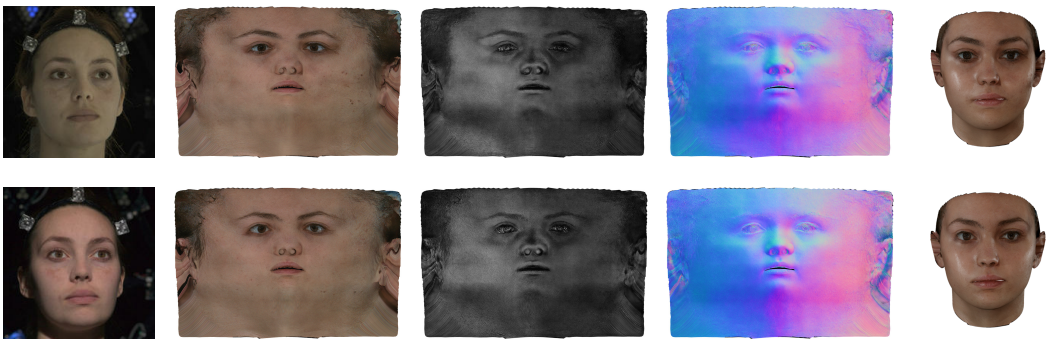
For the qualitative comparisons, we perform 3D reconstructions of arbitrary images. As shown in Figs. 5.7 and 5.8, our method does not produce any artifacts in the final renderings and successfully handles extreme poses and occlusions such as sunglasses. We infer the texture maps in a patch-based manner from high-resolution input, which produces higher-quality details than [138] and [141], who train on high-quality scans but infer the maps for the whole face, in lower resolution. This is also apparent in Fig. 5.9 and Fig. 5.16, which shows our reconstruction after each step of our process. Moreover, we can successfully acquire each component from black-and-white images (Fig. 5.7) and even paintings (Fig. 5.12).

Furthermore, we experiment with different environment conditions, in the input images and while rendering. As presented in Fig. 5.10, the extracted normals, diffuse and specular albedos are consistent, regardless of the illumination on the original input images. Moreover, Fig. 5.12 shows different subjects rendered under different environments. We can realistically illuminate each subject in each scene and accurately reconstruct the environment reflectance, including detailed specular reflections and subsurface scattering.



(a) Input (b) Diff.Alb. (c) Spec.Alb. (d) Diff.Nor. (e) Spec.Nor.

Figure 5.9: Comparison of AvatarMe^{++} predicted reflectance with ground truth. Left: Patch of rendered test subject in target domain, Top row: predicted reflectance with our method, Bottom row: ground truth.



(a) Input (b) Diff.Alb. (c) Sp.Alb. (d) Sp.Norm (e) Render

Figure 5.10: Consistency of AvatarMe^{++} on varying conditions, from the Digital Emily Project [198]. We calculate on average 30.94 PSNR and 0.0007 MSE between our results. Compared to the ground truth from [198], we achieve on average 0.0083 MSE and 20.13 PSNR on albedo and and 0.011 MSE and 24.02 PSNR on normals.

5.5.3 Ablation

Method Components and Variants

We investigate the importance of the various components of our method, when added on the base network. We create a test set $\mathcal{T} = \{\mathcal{R}(\mathbf{R}_i, \mathbf{L}_j)\}$, by rendering 5 test subjects from our captured dataset, RealFaceDB, with reflectance $\mathbf{R}_j$, in 10 total illumination environments $\mathbf{L}_j$ including: a) the 3DMM-target environment (Sec. 5.4.3), b) $-/+30\%$ light source variation in position, c) $-/+30\%$ in intensity from a), and d) directional front and side illumination. We compare the mean squared error, between the ground truth and the model's predicted reflectance and the rendering of the prediction in the source testing environment, as well as in the target environment. Testing in various illumina-

Table 5.2: Method components ablation. Training time is given per training iteration, testing time is given for the inference of whole $\hat{W} \times \hat{H}$ textures. AvatarMe time includes the training time for all 4 of its networks. We measure the mean squared error (MSE) between ground truth and inferred reflectance maps, for the test set $\mathcal{T}$ (Sec 5.5.3).

Method	**Train time**	**Test time**	**MSE**
AvatarMe (Sec. 5.4.5)	22.8	15.98	0.0079
Single Network (Sec. 5.4.7)	8.0	6.35	0.0080
+ Rendering loss (Sec. 5.4.7)	14.4	6.48	0.0075
+ Whole low-res texture (Sec. 5.4.7)	14.8	9.31	0.0066
+ Random environment (Sec. 5.4.7)	14.8	9.35	0.0055
+ Multiple random env. (Sec. 5.4.7)	24.0	9.35	0.0048
+ Occl. AE (Sec. 5.4.6) (AvatarMe^{++})	28.8	9.35	0.0043

tion environments evaluates the generalization abilities of our networks, outside the target environment. We plot both inference and re-rendering error in Fig. 5.13 and show the reconstructed diffuse albedo $\mathbf{A_D}$ in Fig. 5.14. Moreover, although using a single network reduces the training and inference time by about 75%, the additional rendering adds a significant overhead in training time, but does not significantly increase testing time. We show this trade-off between reconstruction quality and training and testing time Table 5.2.

The introduction of the stochastic rendering loss (Sec. 5.4.7) enables AvatarMe^{++} to generalize to facial textures with different environments, while AvatarMe is trained only on the target 3DMM environment. Fig. 5.13 shows the quantitative improvement of AvatarMe^{++} on the test set $\mathcal{T}$ with 10 different environments, two of which are shown in Fig. 5.14. Finally, Fig. 5.11 shows the network's ability to generalize to textures obtained from different datasets (i.e. [200, 201]) and acquisitions methods (i.e. [18, 60]).

Network and Rendering Hyper-parameters

Training the network on 4 Tesla V100 GPUs takes about one day for a base single network (Sec. 5.4 or Sec. 5.4.7) and up to three days for the complete method, with multiple rendering loss computations and self-occlusion in rendering (Sec. 5.4.7). Given these restrictions, we perform a study on the effect of the important hyper parameters. We train the complete method with different seeds and record an average 3.14% standard deviation in the error of reflectance reconstruction and 5.37% standard deviation in the error of re-rendered textures. All the other results are reported using the same seed for all stochastic

operations. The impact of the rendering loss controller λ_R (Eq. 5.14) is shown in Fig. 5.17. We find $\lambda_R = 0.3$ to yield the best results.

Table 5.3 shows the effect of the scene variation hyper-parameters when training with randomized light source $\mathbf{l}$, camera direction $\mathbf{v}$ and light source intensity $\mathbf{c_l}$. We find the best results at up to 50% variation in light source position, and up to 25% in camera position and light source intensity. Finally, Table 5.3 shows the effect of the number of random scenes n_L used to evaluate the rendering loss (Eq. 5.13). We find a significant improvement in using 2 scenes instead of 1, and a smaller improvement when adding additional scenes. However each additional rendering loss evaluation introduces a trade-off with computational time.

Table 5.3: Ablation study for rendering loss hyper-parameters. Comparison of reflectance prediction and re-rendering error for variation in scene parameters (light source position $\mathbf{l}$, camera view direction $\mathbf{v}$ light source intensity $\mathbf{c_l}$) around the target environment (top table), and number of random scenes evaluated n_L in the rendering loss (bottom table) (Sec. 5.4.7). Average error reported for test set $\mathcal{T}$ (Sec. 5.5.3).

Scene Variation			**Results**	
l	**v**	**$\mathbf{c_l}$**	**Recon. MSE**	**Render MSE**
0%	0%	0%	0.0055	0.0059
50%	25%	25%	**0.0043**	**0.0030**
75%	50%	50%	0.0051	0.0036

Evaluations n_L	**Train Time**	**Recon. MSE**	**Render MSE**
1	**15.6**	0.0054	0.0037
2	22.4	0.0045	0.0032
3	28.8	**0.0043**	**0.0030**

5.5.4 Limitations

While our dataset contains a relatively large number of subjects, it does not contain sufficient examples of subjects from certain ethnicities (Sec. 5.3). Despite our effective training data augmentation, we find that the predicted diffuse albedo for subjects of the lightest or darkest skin types, produces patches of inconsistent intensity and the specular normals of older subjects are reconstructed sub-optimally. Moreover, the accuracy of facial reconstruction is not completely independent of the quality of the input photograph, and well-lit, higher resolution photographs produce more accurate results, depending on the

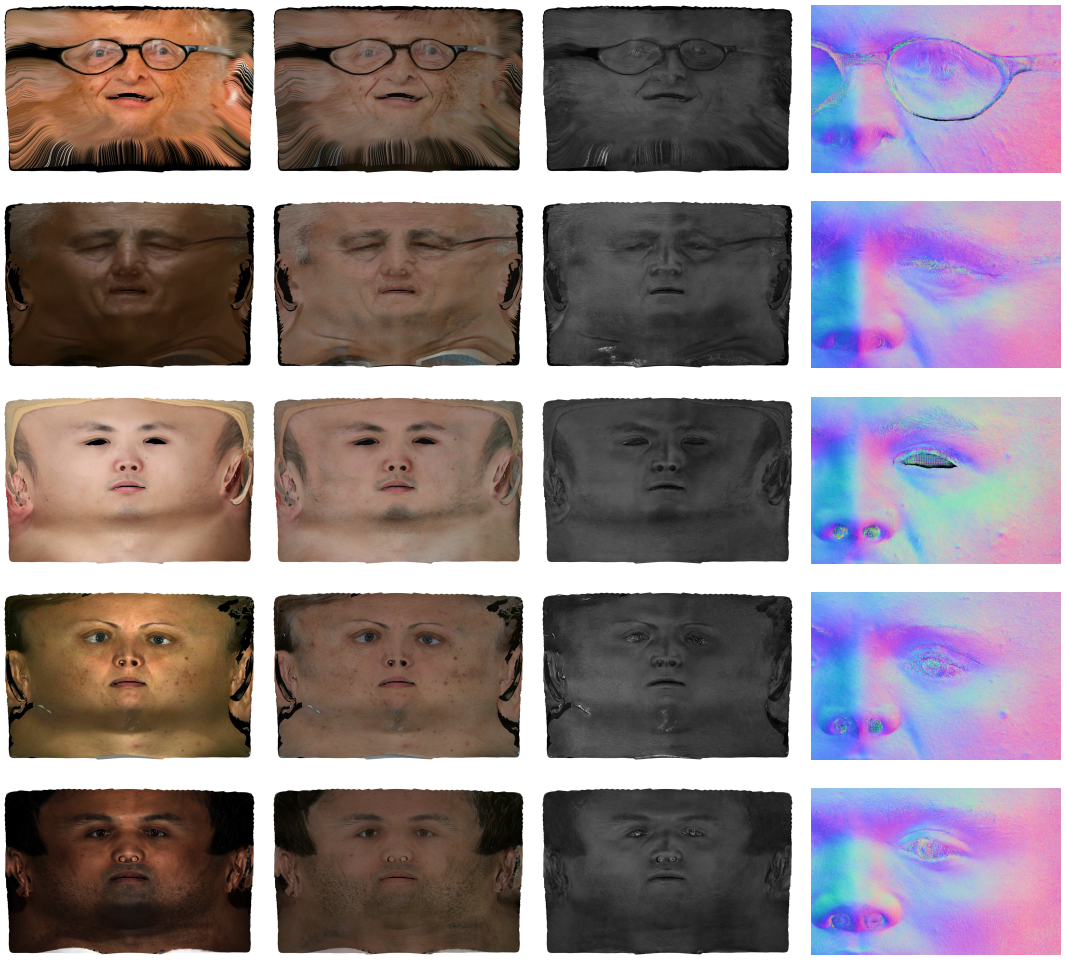
3DMM method used. Additionally, we show that our method can generalize to various reconstruction and capturing methods, however, the model expects their light sources to be in front of the subject. Finally, our renderer models self-occlusion but not occlusion from foreign objects. This could be modeled by augmenting our dataset with randomized occlusions.

5.6 Conclusion

In this paper, we propose the first methodology that produces high-quality rendering-ready face reconstructions from arbitrary "in-the-wild" images. We build upon recently proposed 3D face reconstruction techniques and train an image translation network that can perform estimation of high quality (a) diffuse and specular albedo, and (b) diffuse and specular normals. This is made possible with a large training dataset of 200 faces acquired with high quality facial capture techniques and a fast photorealistic differentiable rendering framework. We demonstrate that it is possible to produce rendering-ready faces from arbitrary face images varying in pose, occlusions, etc., including black-and-white and drawn portraits. Our results exhibit unprecedented level of detail and realism in the reconstructions, while preserving the identity of subjects in the input photographs.

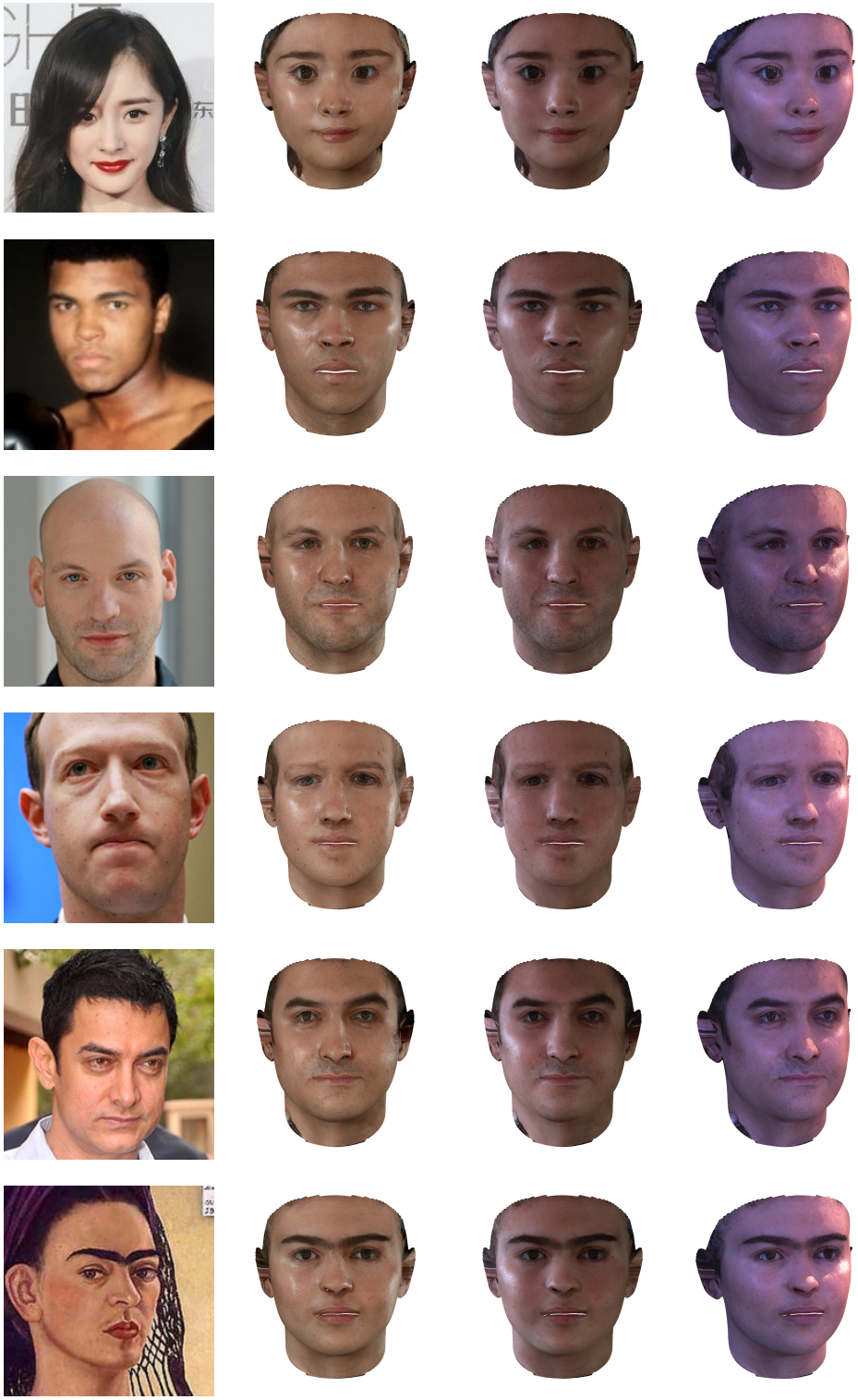
Acknowledgments

AL was supported by EPSRC Project DEFORM (EP/S010203/1) and SM by an Imperial College DTA. AG acknowledges funding by the EPSRC Early Career Fellowship (EP/N006259/1) and SZ from a Google Faculty Fellowship and the EPSRC Fellowship DEFORM (EP/S010203/1).



(q) Input (r) Diffuse Albedo (s) Specular Albedo (t) Spec. Normals

Figure 5.11: Generalization of AvatarMe^{++}: Training AvatarMe^{++} with stochastically varied rendering scene parameters and our rendering loss (Sec. 5.4.7), makes the network domain-agnostic to an extent. Here we show results of a single AvatarMe^{++} network, on reconstructions and captured data, registered to our topology. From top to bottom: a) Reconstructed subject, with Facial Details Synthesis proxy and texture [138], b) Reconstructed subject with OSTeC fitting and texture completion [60], c) Captured subject from FaceScape [200], d) Captured subject from Superface [201], e) Captured subject with 3dMDface (3dmd.com).



(a) Input (b) Cathedral (c) Night Street (d) Sunset

Figure 5.12: AvatarMe^{++} results rendered in different environments.

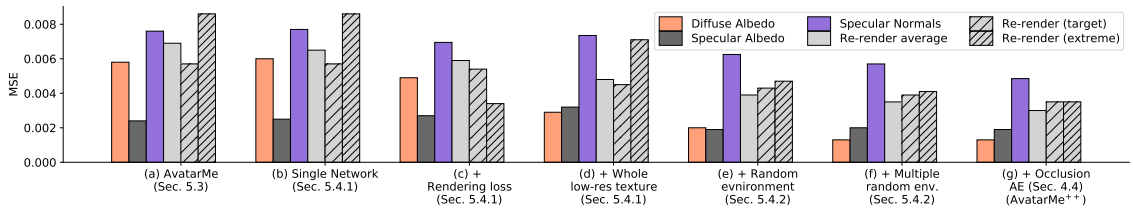


Figure 5.13: Quantitative ablation results for the different variations of our method. From left to right, we start with the base AvatarMe and introduce each component of AvatarMe^{++}. Each variation is applied on test set $\mathcal{T}$ (Sec. 5.5.3). We measure the mean squared error (MSE) between ground truth and prediction's reflectance, and re-rendering in the target environment and an extreme side illumination environment used in $\mathcal{T}$.



Figure 5.14: From left to right, rendered test image, predicted diffuse albedo $\mathbf{A_D}$ from each of the main method variations and the ground truth diffuse albedo $\mathbf{A_D^*}$. Top row input is rendered in our target environment, bottom row in an extreme side illumination environment from $\mathcal{T}$.

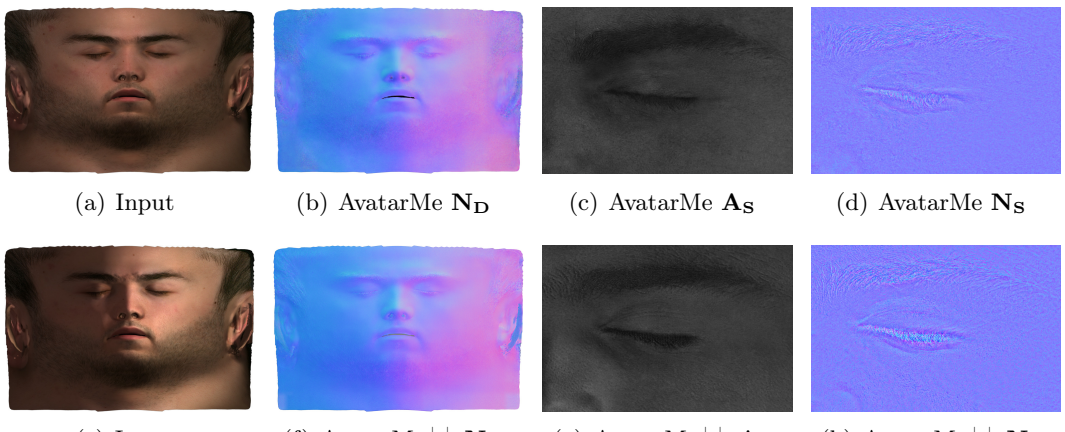


Figure 5.15: Predicted diffuse normals $\mathbf{N_D}$, specular albedo $\mathbf{A_D}$, and specular normals details $\mathbf{N_S}$ in tangent space, between AvatarMe (Sec.5.4.5) and AvatarMe^{++}. Diffuse albedo $\mathbf{A_D}$ comparison included in Fig. 5.14.



(a) Input (b) 3DMM Recon [19] (c) Ours, 3DMM Super Res. (d) Ours, AvatarMe (e) Ours, AvatarMe^{++}

Figure 5.16: Comparison of our shape reconstruction (left) and rendering (right) between the 3DMM fitting with GANFIT [19], upsampled texture (Sec. 5.4.4), AvatarMe (Sec. 5.4.5) and AvatarMe^{++} (Sec. 5.4.7).

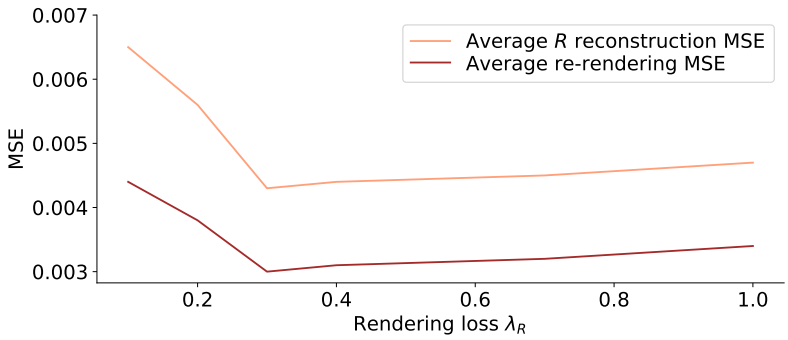


Figure 5.17: The effect of different values of the rendering loss controller λ_R (Eq. 5.14) in the average reflectance $\mathbf{R}$ reconstruction for 5 subjects rendered in 10 different environments and the average re-rendering mean squared truth with the ground-truth.

Chapter 6

Facial Reflectance Generation and Fitting to Arbitrary Images

Contents

6.1 Introduction . . . 115

6.2 Related Work . . . 120

6.2.1 3D Morphable Models . . . 120

6.2.2 Deep Generative Networks . . . 121

6.2.3 Facial Reflectance Acquisition . . . 121

6.3 Method . . . 122

6.3.1 The FitMe Deep 3D Morphable Model . . . 123

6.3.2 Dataset Acquisition and Augmentation . . . 124

6.3.3 Fast Differentiable Photorealistic Rendering . . . 126

6.3.4 Fitting by Inversion through Rendering . . . 127

6.4 Experiments . . . 130

6.4.1 Implementation Details . . . 130

6.4.2 Single-Image Reconstruction . . . 131

6.4.3 Multi-Image Reconstruction . . . 132

6.4.4 Dataset Augmentation . . . 133

6.5 Model Manipulation . . . 134

6.6 Ablation Study . . . 135

6.7 Additional Results . . . 137

6.8 Implementation Details 139

6.9 Limitations 140

6.10 Conclusion 140

Here we introduce FitMe, a multi-modal deep 3DMM for facial and head avatars, which builds on and improves the previous work presented in the previous section. This work has been published in *A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, J. Deng, and S. Zafeiriou, "Fitme: Deep photorealistic 3D morphable model avatars," in* Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), *pp. 8629–8640, June 2023.*

In this paper, we introduce *FitMe*, a facial reflectance model and a differentiable rendering optimization pipeline, that can be used to acquire high-fidelity renderable human avatars from single or multiple images. The model consists of a multi-modal style-based generator, that captures facial appearance in terms of diffuse and specular reflectance, and a PCA-based shape model. We employ a fast differentiable rendering process that can be used in an optimization pipeline, while also achieving photorealistic facial shading. Our optimization process accurately captures both the facial reflectance and shape in high detail, by exploiting the expressivity of the style-based latent representation and of our shape model. FitMe achieves state-of-the-art reflectance acquisition and identity preservation on single "in-the-wild" facial images, while it produces impressive scan-like results, when given multiple unconstrained facial images pertaining to the same identity. In contrast with recent implicit avatar reconstructions, FitMe requires only one minute and produces relightable mesh and texture-based avatars, that can be used by end-user applications.

6.1 Introduction

Despite the tremendous steps forward witnessed in the last decade, 3D facial reconstruction from a single unconstrained image remains an important research problem with an active presence in the computer vision community. Its applications are now wide-ranging, including but not limited to human digitization for virtual and augmented reality applications, social media and gaming, synthetic dataset creation, and health applications. However, recent works come short of accurately reconstructing the identity of different subjects and

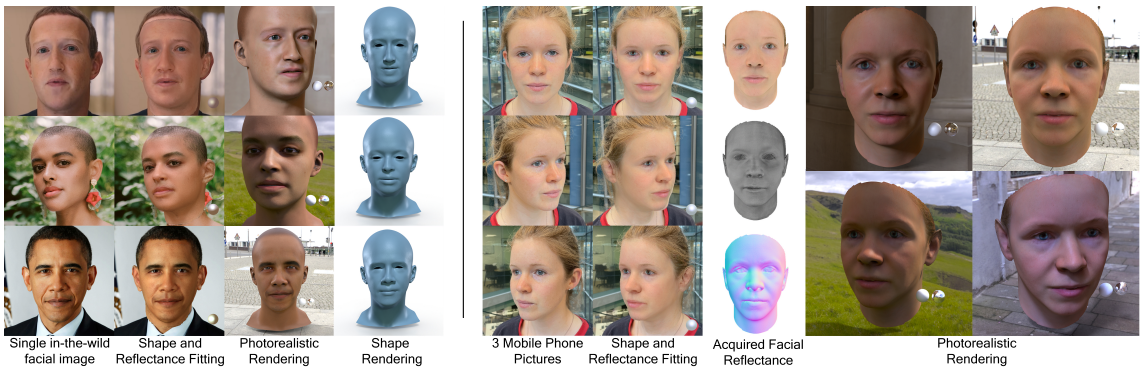


Figure 6.1: FitMe reconstructs relightable shape and reflectance maps for facial avatars, from a single (left) or multiple (right) unconstrained facial images, using a reflectance model and differentiable rendering. The results can be photorealistically rendered in common engines.

usually fail to produce assets that can be used for photorealistic rendering. This can be attributed to the lack of diverse and big datasets of scanned human geometry and reflectance, the limited and ambiguous information available on a single facial image, and the limitations of the current statistical and machine learning methods.

3D Morphable Models (3DMM) [202] have been a standard method of facial shape and appearance acquisition from a single "in-the-wild" image. The seminal 3DMM work in [6] used Principal Component Analysis (PCA), to model facial shape and appearance with variable identity and expression, learned from about 200 subjects. Since then, larger models have been introduced, .i.e. the LSFM [154], Basel Face Model [203] and Facescape [200], with thousands of subjects. Moreover, recent works have introduced 3DMMs of complete human heads [204, 205, 206] or other facial parts such as ears [206] and tongue [207]. Finally, recent works have introduced extensions ranging from non-linear models [208, 209, 210] to directly regressing 3DMM parameters [211, 212]. However, such models do not have the capacity to capture photorealistic textures.

During the last decade, we have seen considerable improvements in deep generative models. Generative Adversarial Networks (GANs) [13], and specifically progressive GAN architectures [213] have achieved tremendous results in learning distributions of high-resolution 2D images of human faces. Recently, style-based progressive generative networks [16, 17, 214, 215] are able to learn meaningful latent spaces that can be traversed in order

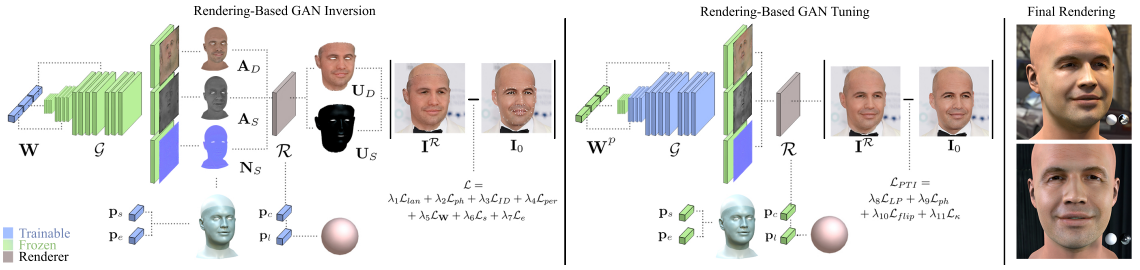


Figure 6.2: FitMe method overview. For a target image $\mathbf{I}_0$, we optimize the latent vector $\mathbf{W}$ of the generator $\mathcal{G}$, and the shape identity $\mathbf{p}_s$, expression $\mathbf{p}_e$, camera $\mathbf{p}_c$ and illumination $\mathbf{p}_l$ parameters, by combining 3DMM fitting and GAN inversion methods, through accurate differentiable diffuse $\mathbf{U}_D$ and specular $\mathbf{U}_S$ rendering $\mathcal{R}$. Then, from the optimized $\mathbf{W}^p$, we tune the generator $\mathcal{G}$ weights, through the same rendering process. The reconstructed shape $\mathbf{S}$ and facial reflectance (diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and normals $\mathbf{N}_S$), achieve great identity similarity and can be directly used in typical renderers, to achieve photorealistic rendering, as shown on the right-hand side.

to reconstruct and manipulate different attributes of the generated samples. Some methods have also been shown effective in learning a 2D representation of 3D facial attributes, such as UV maps [19, 185, 216, 217].

3D facial meshes generated by 3DMMs can be utilized in rendering functions, in order to create 2D facial images. Differentiating the rendering process is also required in order to perform iterative optimization. Recent advances in differentiable rasterization [218], photorealistic facial shading [24], and rendering libraries [50, 219, 220] enable the photorealistic differentiable rendering of such assets. Unfortunately, 3DMM [19, 154, 185] works rely on the Lambertian shading model which comes short of capturing the complexity of facial reflectance. The issue is photorealistic facial rendering requires various facial reflectance parameters instead of a single RGB texture [24]. Such datasets are scarce, small, and challenging to capture [74, 76, 221], despite recent attempts to simplify such setups [22]. Despite the tremendous steps forward witnessed in the last decade, 3D facial reconstruction from a single unconstrained image remains an important research problem with an active presence in the computer vision community. Its applications are now wide-ranging, including but not limited to human digitization for virtual and augmented reality applications, social media and gaming, synthetic dataset creation, and health applications. However, recent works come short of accurately reconstructing the identity of different subjects and usually fail to produce assets that can be used for photorealistic rendering. This can be attributed to the lack of diverse and big datasets of scanned human geometry and reflectance, the limited and ambiguous information available on a single facial image,

and the limitations of the current statistical and machine learning methods.

3D Morphable Models (3DMM) [202] have been a standard method of facial shape and appearance acquisition from a single "in-the-wild" image. The seminal 3DMM work in [6] used Principal Component Analysis (PCA), to model facial shape and appearance with variable identity and expression, learned from about 200 subjects. Since then, larger models have been introduced, .i.e. the LSFM [154], Basel Face Model [203] and Facescape [200], with thousands of subjects. Moreover, recent works have introduced 3DMMs of complete human heads [204, 205, 206] or other facial parts such as ears [206] and tongue [207]. Finally, recent works have introduced extensions ranging from non-linear models [208, 209, 210] to directly regressing 3DMM parameters [211, 212]. However, such models cannot produce textures capable of photorealistic rendering.

During the last decade, we have seen considerable improvements in deep generative models. Generative Adversarial Networks (GANs) [13], and specifically progressive GAN architectures [213] have achieved tremendous results in learning distributions of high-resolution 2D images of human faces. Recently, style-based progressive generative networks [16, 17, 214, 215] are able to learn meaningful latent spaces that can be traversed in order to reconstruct and manipulate different attributes of the generated samples. Some methods have also been shown effective in learning a 2D representation of 3D facial attributes, such as UV maps [19, 185, 216, 217].

3D facial meshes generated by 3DMMs can be utilized in rendering functions, in order to create 2D facial images. Differentiating the rendering process is also required in order to perform iterative optimization. Recent advances in differentiable rasterization [218], photorealistic facial shading [24] and rendering libraries [50, 219, 220] enable the photorealistic differentiable rendering of such assets. Unfortunately, 3DMM [19, 154, 185] works rely on the Lambertian shading model which comes short of capturing the complexity of facial reflectance. The issue being photorealistic facial rendering requires various facial reflectance parameters instead of a single RGB texture [24]. Such datasets are scarce, small, and challenging to capture [74, 76, 221], despite recent attempts to simplify such setups [22].

Several recent approaches have achieved either high-fidelity facial reconstructions [19, 185, 222] or relightable facial reflectance reconstructions [23, 24, 188, 223, 224, 225], includ-

ing infra-red [32], however, high-fidelity and relightable reconstruction still remains elusive. Moreover, powerful models have been shown to capture facial appearance with deep models [27, 226], but they fail to show single or multi-image reconstructions. A recent alternative paradigm uses implicit representations to capture avatar appearance and geometry [227, 228], the rendering of which depends on a learned neural rendering. Despite their impressive results, such implicit representations cannot be used by common renderers and are not usually relightable. Finally, the recently introduced Albedo Morphable Model (AlbedoMM) [223] captures facial reflectance and shape with a linear PCA model, but per-vertex color and normal reconstruction are too low-resolution for photorealistic rendering. AvatarMe++ [23, 24] reconstructs high-resolution facial reflectance texture maps from a single "in-the-wild" image, however, its 3-step process (reconstruction, upsampling, reflectance), cannot be optimized directly with the input image.

In this work, we introduce *FitMe*, a fully renderable 3DMM with high-resolution facial reflectance texture maps, which can be fit on unconstrained facial images using accurate differentiable renderings. FitMe achieves identity similarity and high-detailed, fully renderable reconstructions, which are directly usable by off-the-shelf rendering applications. The texture model is designed as a multi-modal style-based progressive generator, which concurrently generates the facial *diffuse-albedo*, *specular-albedo* and *surface-normals*. A meticulously designed branched discriminator enables smooth training with modalities of different statistics. To train the model, we create a capture-quality facial reflectance dataset of 5k subjects by fine-tuning AvatarMe++ on the MimicMe [229] public dataset, which we also augment in order to balance skin-tone representation. For the shape, we use interchangeably a face and head PCA model [206], both trained on large-scale geometry datasets. We design a single or multi-image fitting method based on style-based generator projection [16] and 3DMM fitting. The rendering function must be differentiable and fast to perform efficient iterative fitting (under 1 minute) the rendering function must be differentiable and fast, making models such as path tracing unusable. Prior works in the field [19, 154, 230] use simpler shading models (e.g., Lambertian) or much slower optimization [188]. We add a more photorealistic shading than prior work, with plausible diffuse and specular rendering, which can acquire shape and reflectance capable of photorealistic rendering in standard rendering engines (Fig. 6.1). The flexibility of the generator's extended latent space and the photorealistic fitting, enables FitMe to reconstruct high-fidelity facial

reflectance and achieve impressive identity similarity, while accurately capturing details in diffuse, specular albedo and normals. Overall, in this work we present:

- the first 3DMM capable of generating high-resolution facial reflectance and shape, with an increasing level of detail that can be photorealistically rendered,
- the first branched multi-modal style-based progressive generator of high-resolution 3D facial assets (diffuse albedo, specular albedo, and normals), and a suitable multi-modal branched discriminator,
- a method to acquire and augment a vast facial reflectance dataset of using assets from a public dataset,
- a multi-modal generator projection optimized with diffuse and specular differentiable rendering.

6.2 Related Work

6.2.1 3D Morphable Models

Early facial modeling and fitting methods, starting from the seminal 3DMM work of Blanz and Vetter in [6], have always used a linear model for facial shape and appearance [154, 203]. They are thoroughly analyzed in the recent review of Egger et al. [202]. Moreover, 3DMMs have been extended to the entire head in [206]. Finally, the recent AlbedoMM [223] uses separate PCA models of the diffuse and specular albedo. However, the per-vertex albedo is low resolution and inadequate for photorealistic rendering.

Recent works have been replacing parts of the linear shape/texture models with neural networks in order to capture non-linearities. Bagautdinov et al.[231] introduced the first method for non-linear facial shape modeling based on variational autoencoders. 3DFace-GAN [208] introduced a GAN-based approach for facial geometry based on UV maps which captured important non-linearities of the facial shape. Closest to our work, [27, 226] designed models that combine facial albedo, shape, and normals.

6.2.2 Deep Generative Networks

GANs [13] first achieved high-resolution facial generation with the progressive growing of GANs [213], which introduced a generator-discriminator pair trained on progressively growing resolutions. StyleGAN [214] also introduced a noise-injection technique and a mapping network that learns meaningful latent representations. StyleGAN2 [17] further optimized the architecture and projection method, while StyleGAN2ADA [16] introduced online data augmentation methods that help when training with limited data, where the discriminator overfits. Finally, [215] improved the signal flow in the generator. In our work, we build upon these powerful models in order to achieve multi-modal reflectance generation.

6.2.3 Facial Reflectance Acquisition

The first device to accurately acquire 3D facial scans was the LightStage [232], a room-sized dome equipped with programmable illumination and high-end cameras. The diffuse and specular components of the reflectance can be separated by exploiting polarization in multiple captures of gradient illumination [76]. Simplified methods have also been proposed, using unpolarized binary patterns [147] or passive illumination [221]. Although highly accurate, the above methods require large and expensive equipment. A recent practical system made of commodity devices [22] enables significantly faster and cheaper facial reflectance acquisition.

Inverse Rendering approaches have also been successful in acquiring facial reflectance. Recent works employ differentiable ray tracing algorithms [188, 224] to solve an inverse rendering optimization that yields facial diffuse and specular components, as well as specular roughness. However, such methods are computationally expensive and the optimization is susceptible to ambiguity in the subject images.

To overcome such ambiguities, multiple approaches use linear or deep models as priors or directly regress the facial reflectance. For example, early approaches proposed deep image-translation models [230, 233, 234], while later methods [139, 235, 236] first acquired realistic facial albedos and displacement textures, using one or more deep neural networks. GANFIT [19] was the first method to use a linear shape model in combination with a

GAN-based texture model. Nevertheless, the inferred texture contained baked environment illumination, making photorealistic rendering impossible. AvatarMe [23] introduced a super-resolution and an image-translation network that transformed textures generated from GANFIT into high-resolution facial reflectance. Its extension AvatarMe++ [24] introduced a generalized facial model that works with arbitrary 3DMM fitting algorithms or scanning methods using a differentiable shader. However, both methods are separated from the initial fitting method and cannot pick up all the facial and illumination cues of the input image. An alternative approach [222] produces highly realistic facial shape and albedo but requires a video stream of RGB and depth data for optimization. Moreover, TRUST [225] proposes a method to overcome the ambiguity between skin tone and illumination optimization by using cues from the background illumination or multiple faces in the same image. Closest to our work, Luo et al. [185] acquire normalized avatars using a style-based generator together with an iterative optimization. Although similar, our approach has the following advantages: a) our generative model is extended to complete facial reflectance, and our renderer is extended with a more photorealistic shading model, b) our branched generator and discriminator approach generates additional modalities, c) our fitting pipeline introduces a latent space and shading regularization, as well as identity loss, and d) our fine-tuning results in an editable latent space, which can be further optimized through rendering.

6.3 Method

In this work, we present FitMe, a deep facial reflectance 3DMM, which is based on a branched multi-modal style-based generative network (Sec. 6.3.1). It is trained on a big capture-quality facial reflectance dataset, which is augmented to balance skin tones (Sec. 6.3.2). The nature of the data allows accurate diffuse and specular differentiable rendering using an appropriate shader (Sec. 6.3.3). Finally, we combine a style-based generator latent space projection with 3DMM fitting methods (Sec. 6.3.4) and achieve high-fidelity facial reconstruction from a single or multiple images. The acquired shape and reflectance avatars achieve high identity similarity (given the model's flexibility, diverse dataset, and rendering), have expression blendshapes and can be directly used by common rendering applications.

6.3.1 The FitMe Deep 3D Morphable Model

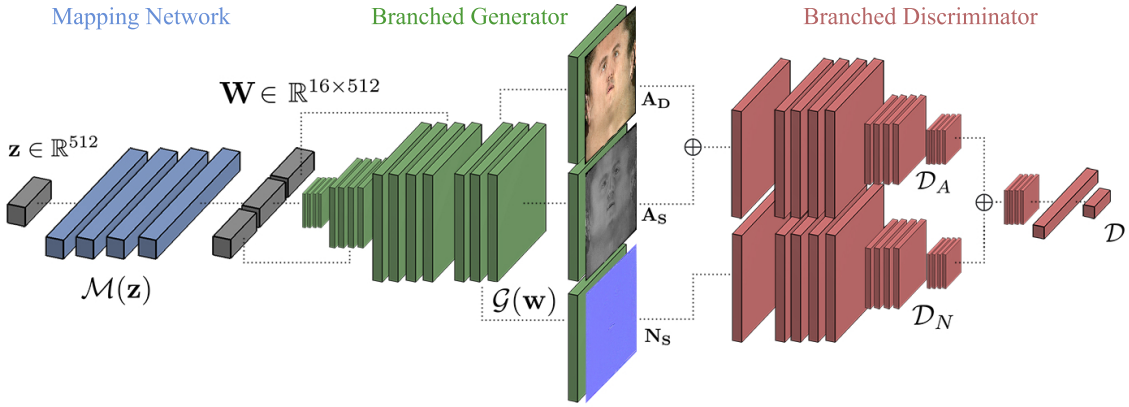


Figure 6.3: Overview of BRDF-GAN, a style-based generator [17], consisting of a mapping network $\mathcal{M}$ which translates the latent vector $\mathbf{z}$, to style $\mathbf{W}$. The branched, multi-modal synthesis network $\mathcal{G}$ generates a diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and normals $\mathbf{N}_S$ at 1024×1024 resolution. The model is trained in tandem with a branched multi-modal discriminator $\mathcal{D}$. Each branch enables $\mathcal{D}$ to model the distribution of the albedo and normals separately The output of each branch is concatenated before entering the last convolutional block and the fully connected layers of $\mathcal{D}$, in order to maintain feature consistency between modalities.

We introduce BRDF-GAN, a multi-modal style-based generative network, which concurrently generates facial diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and surface normals $\mathbf{N}_S$ in a UV parameterization at 1024×1024 resolution. The model is trained by a novel branched discriminator, that ensures the consistency between the modalities. The shape is modeled with a 3DMM, and we model both the facial and the head mesh topology similar to [206].

Generating the shape as another branch of BRDF-GAN requires many redundant parameters, is prone to quantization, and its fitting is problematic. A recent approach generates 3DMM offsets as a UV texture map but still requires 3DMM projection to generate the final shape [185]. We opt instead to use a 3DMM with a mesh representation. We use the Universal Head Model [206], with a facial UV topology, which is completed after fitting. For a set of identity $\mathbf{p_s}$ and expression $\mathbf{p_e}$ parameters, with identity and expression bases $\mathbf{U_s}$ and $\mathbf{U_e}$, respectively, and mean $\mathbf{m_s}$, the facial geometry $\mathbf{S}(\mathbf{p_s}, \mathbf{p_e}) \in \mathbb{R}^{106317\times3}$ can be reconstructed as:

$$\mathbf{S}(\mathbf{p_s}, \mathbf{p_e}) = \mathbf{m_s} + \mathbf{U_s}\mathbf{p_s} + \mathbf{U_e}\mathbf{p_e} \tag{6.1}$$

Based on StyleGAN2 [17], BRDF-GAN consists of a mapping network $\mathcal{M} : \mathbf{z} \rightarrow \mathbf{W}$, which translates a latent vector $\mathbf{z} \in \mathbb{R}^{512}$ into the latent space $\mathbf{W} \in \mathbb{R}^{16\times512}$, a multi-modal

branched synthesis network $\mathcal{G} : \mathbf{W} \rightarrow \mathbf{T}$, where $\mathbf{T}_R = \{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_S\} \in \mathbb{R}^{7\times1024\times1024}$. Both the diffuse albedo and normals have three channels, however, we use a monochrome specular albedo, which is typical practice in shading [24, 147]. The synthesis network $\mathcal{G}$ follows the skip-connection convolutional blocks architecture of StyleGAN2ADA[16], however, the last block of each up-sampled resolution is branched per reflectance mode, as shown in Fig. 6.3. For resolutions of 512×512 and up, we find this necessary to achieve satisfactory FID scores. Furthermore, using a single generator while branching the final block for each modality enables the details to be accurately captured between modalities. It ensures the final output corresponds to the same identity.

BRDF-GAN is also trained using a branched discriminator $\mathcal{D}$. We observed that the distributions of the albedos $\mathbf{A}_D$, $\mathbf{A}_S$ with mean 0.24 and std. 0.12, and the normals $\mathbf{N}_S$ with mean 0.61 and std 0.24, were significantly different, something which inhibited the GAN training using a vanilla $\mathcal{D}$ (i.e., a discriminator with no branches). Therefore, we designed $\mathcal{D}$ as a long-branched network, based on the residual-based discriminator of StyleGAN2-ADA[16]. One branch $\mathcal{D}_A$ receives as input the concatenation of the diffuse and specular albedos $\mathbf{A}_D \bigoplus \mathbf{A}_S$, which have similar statistics, while the other branch $\mathcal{D}_N$ receives the surface normals $\mathbf{N}_S$. After running all-but-the-last convolutional blocks, the output of both branches is concatenated and given to the last convolutional block and the fully connected layers of $\mathcal{D}$. This ensures that $\mathcal{D}$ can accurately capture the much different distribution of the albedo and normals, while ensuring consistent facial features per generated subject. A detailed visualization is shown in Fig. 6.2. Luo et al. [185] recently proposed a similar model, where the generator outputs an albedo and a shape and is trained by three complete and separate discriminators. Our approach not only requires a smaller network but is also able to generate facial reflectance modalities of highly different distributions and, as we demonstrate in the experiments, outperforms their generalization capabilities.

6.3.2 Dataset Acquisition and Augmentation

State-of-the-art 3DMMs and GANs require thousands of data samples [16, 154], however, no facial reflectance datasets exist at this scale. Instead, we utilize a recently published large dataset of facial captures [229], and a state-of-the-art facial appearance generation model [24], to create such a dataset at capture-like quality and high resolution. Finally,

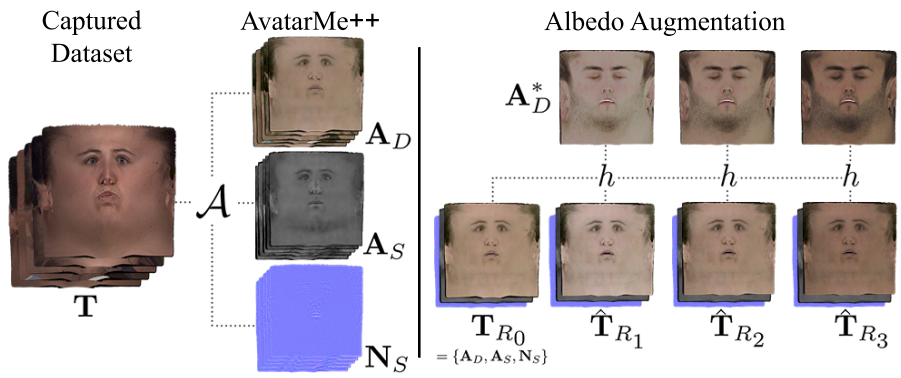


Figure 6.4: Dataset and augmentation. Left: creation of diffuse $\mathbf{A}_D$ and specular $\mathbf{A}_S$ albedo and normals $\mathbf{N}_S$, using finetuned a AvatarMe++ [24] network $\mathcal{A}$ on the MimicMe dataset [229] of textures $\mathbf{T}$. Right: albedo skin-tone augmentation using our masked histogram matching h on different target albedos $\mathbf{A}_D^*$ (Sec. 6.3.2).

we alleviate the skin-tone imbalance in the dataset, by augmenting the resulting data with histogram matching [237].

Firstly, we acquire the MimicMe dataset [229], consisting of $\mathcal{T} = \{\mathbf{T}_0, \ldots, \mathbf{T}_{n_T}\}, n_T = 4,700$ neutral facial textures, which are projected to a common UV map parameterization. These textures include diverse subjects and expressions [229], however, they have baked illumination from the capturing system and do not constitute an albedo. We train an image-to-image translation network $\mathcal{A} : T \rightarrow \{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_S\}$ that transforms textures from $\mathcal{T}$ to reflectance textures $\{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_S\}$, following AvatarMe++ [24], while making the following changes: (a) we approximate the capturing environment of [229] using AvatarMe++ [24], and train the AvatarMe++ network $\mathcal{A}$ for this environment only, (b) we generate only the surface (i.e., specular) normals $\mathbf{N}_S$, in tangent space, which aids the FitMe fitting, as discussed in 6.3.3 and (c) we do not use a super-resolution network since the textures in $\mathcal{T}$ are high-resolution captured textures. In this manner, we can use $\mathcal{A}$ to transform $\mathcal{T}$ into a dataset of reflectance textures $\mathcal{T}_R$.

Regarding the shape, we choose not to represent it as a UV map, and instead, we use an existing PCA model. Although we can extract the shape from the above datasets, our experiments show no improvement over using PCA, while also making the network training more challenging. Therefore, with our method being shape-model agnostic, we use the public UHM head shape model [206].

Biased skin tone prediction is a significant problem [225], and therefore balanced skin tone representation in the dataset is of paramount importance. Skin tone augmentation can be

done by manipulating the melanin and hemoglobin values of the albedo, which requires expensive data collection [28], pre-training [238] or potentially noisy inverse rendering [239]. Instead, we propose an alternative method based on histogram matching [237] (Overview in Fig. 6.4), which is trivial to use and noise-free. We acquire 10 scanned albedos [76], with each subject matched to the Monk Skin-Tone Scale (MST) [240] and transform them into our facial UV topology. Then, for each albedo $\mathbf{A}_D$ in the reflectance dataset, we perform histogram matching h [237] with a number of randomly selected target albedos $\mathbf{A}_D^*$ from the MST scale albedos. To avoid non-skin features that could affect the transformation, we calculate a mask $\mathbf{M}$ using the distance of each pixel from the average skin tone $\bar{\mathbf{A}}_D$ in the forehead region $\mathbf{M} = \left|\left|\mathbf{A}_D - \bar{\mathbf{A}}_D\right|\right|$. Then, we acquire an augmented albedo $\hat{\mathbf{A}}_D$ as:

$$\hat{\mathbf{A}}_D = \mathbf{M} \odot h(\mathbf{A}_D, \mathbf{A}_D^*) + (1 - \mathbf{M}) \odot \mathbf{A}_D \tag{6.2}$$

6.3.3 Fast Differentiable Photorealistic Rendering

Typically, 3DMM models rely on a single RGB per-vertex texture color, a pinhole camera projection model and a diffuse-only Lambertian shading model, based on spherical harmonics illumination [202]. Although such modeling provides enough cues to the fitting function for the shape and texture parameters to be optimized, the texture model cannot capture the facial reflectance parameters required for photorealistic rendering, and the Lambertian shading model cannot handle both the diffuse and specular components.

On the contrary, differentiable local-reflection models such as Blinn-Phong [58] can achieve more accurate facial shading [24]. We define a viewing direction $\mathbf{v}$, an ambient intensity $\mathbf{c_a}$, and a set of n_l light sources with direction $\mathbf{l_j}$ and intensity $\mathbf{c_j}$. For a rasterized pixel i with diffuse albedo $\mathbf{A}_D$, diffuse normals $\mathbf{N}_D$, specular albedo $\mathbf{S}_A$ and surface normals $\mathbf{N}_S$, the diffuse $\mathbf{U_{i_D}}$ and specular $\mathbf{U_{i_S}}$ shading components are defined as:

$$\mathbf{U_{D_i}} = \mathbf{c_a}\mathbf{A_{D_i}} \sum_{j=1}^{n_l} (\mathbf{N_{D_i}} \cdot \mathbf{l_j})\mathbf{c_j} \tag{6.3}$$

$$\mathbf{U_{S_i}} = \mathbf{A_{S_i}} \sum_{j=1}^{n_l} (\chi^+(\mathbf{N_{S_i}} \cdot \mathbf{h_j}))^s \mathbf{c_j}, \quad \mathbf{h_j} = \frac{\mathbf{l_j} + \mathbf{v_j}}{||\mathbf{l_j} + \mathbf{v_j}||} \tag{6.4}$$

where $\chi^+(x)$ is a piece-wise function that returns $\max\{0, x\}$, and s is the shininess co-

efficient. The final shading is acquired by addition $\mathbf{U_i} = \mathbf{U_{i_D}} + \mathbf{U_{i_S}}$. We also perform subsurface scattering using [24].

For photorealistic shading, separate diffuse and surface normals are required [24, 74, 76], so that the diffuse component approximates the smoother subsurface scattering. In order to correctly optimize the shape mesh during fitting, we use the smoother geometric normals as diffuse normals $\mathbf{N}_D$. The detailed specular normals are generated by BRDF-GAN in tangent space $\mathbf{N}_S$, and during rendering are added to the shape normals $\mathbf{N}$, to acquire the specular normals in object space.

Our differentiable rendering implementation, not only renders skin photorealistically compared to recent fitting methods [19, 185, 223], but it is also implemented directly in image-space, compared to [24], so that direct supervision can be achieved. Moreover, the 3DMM shape $\mathbf{S}$ and GAN-generated reflectance texture maps $\mathbf{T}_R = \{\mathbf{A}_D, \mathbf{A}_S, \mathbf{N}_S\}$ can be efficiently optimized by photometric, identity and perceptual losses. The rendering function $\mathcal{R}$ with optimizable camera $\mathbf{p}_c$ and illumination $\mathbf{p}_l$ can be formulated as:

$$\mathbf{I}_{\mathcal{R}} = \mathcal{R}\left(\mathbf{S}, \mathbf{T}_R, \mathbf{p_c}, \mathbf{p_l}\right) \tag{6.5}$$

6.3.4 Fitting by Inversion through Rendering

Following conventional 3DMM fitting approaches [6, 19, 154, 202, 241], we build an optimization pipeline with two major improvements: (a) we replace the statistical texture model and its PCA-based optimization, with BRDF-GAN textures and GAN inversion optimization [17, 242], b) we implement an accurate differentiable renderer in the image space, to enable BRDF-GAN inversion by rendering its results in high fidelity. As described below, our optimization pipeline combines the merits of both 3DMM fitting and GAN inversion and achieves photorealistic reflectance reconstruction.

BRDF-GAN Inversion

Generator inversion refers to the acquisition of latent code that recreates the target image and its application on style-based generators is a widely studied problem [17, 242]. However, all the above assume a 2D target image and can directly optimize the generator's

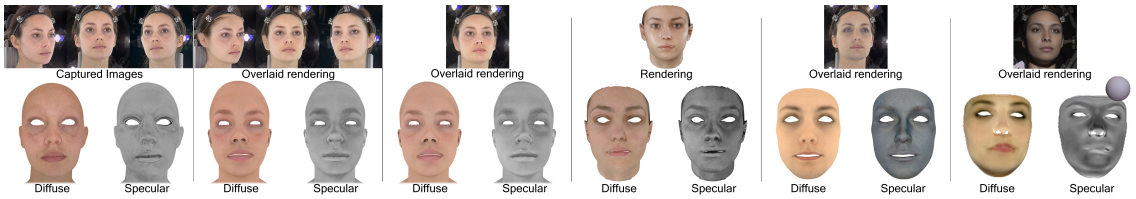
output. Moreover, FitMe requires to pass the generated reflectance through rasterization and rendering, while preserving the meaningful properties of the texture map, for which a standard inversion does not suffice. Inspired by both previous 3DMM fitting methods [19, 24, 154, 188], and inversion [16, 242], we design a rendering-based inversion method for our multi-modal generator.

Style-based generators [16, 17] feed a latent code $\mathbf{z}$ into a mapping network to acquire the native latent space $\mathbf{W}$, which is fed to the synthesis network. Inversion methods optimize $\mathbf{W}$ directly, whilst also optimizing the noise vector $\mathbf{n}$ [16, 242], by using a perceptual LPIPS loss [243]. However, the multi-modal generator faces three challenges: a) being multi-modal makes the concurrent optimization of the different modes more difficult, b) requires rasterization, and as such, not all pixels are visible to the loss function and c) requires shading, shape, camera, and light source optimization, which all remain sensitive to the optimization with LPIPS. We find that an additional set of losses make the inversion more accurate and stable:

The terms of our objective function include both primitive supervision, such as a landmark, photometric, and regularization loss functions, as well as identity and perceptual losses. For a target image $\mathbf{I_0}$ and a rendered reconstruction $\mathbf{I}_{\mathcal{R}}$, we used the following losses to optimize the shape 3DMM, the camera and illumination parameters, and also to guide the rasterized BRDF-GAN results:

Landmark Loss: We estimate 3D landmarks using a deep alignment network [126] $\mathcal{M}(\mathbf{I}) : \mathbb{R}^{H\times W\times C} \rightarrow \mathbb{R}^{68\times 2}$ and penalize the distance by $\mathcal{L}_{lan} = ||\mathcal{M}(\mathbf{I_0}) - \mathcal{M}(\mathbf{I}_{\mathcal{R}})||_2$.

Photometric Loss: It is defined as the distance between per-pixel color intensity values, to capture the skin color and illumination from the target image as $\mathcal{L}_{ph} = ||\mathbf{I_0} - \mathbf{I}_{\mathcal{R}}||_1$.



Scan [244] Ours, 3 images Ours, 1 image AvatarMe++[24] Alb. MM [223] Dib et al. [224]

Figure 6.5: Comparison of diffuse and specular albedo reconstruction and rendering of Digital Emily [244] with prior works. Our single and three-image reconstructions achieve similar results to the captured data, which need specialized hardware and hundreds of images.

Identity Loss: Following [216, 245], we supervise fitting by identity features extracted by a face recognition network [246] with n layers, $\mathcal{F}^n(\mathbf{I}) : \mathbb{R}^{H\times W\times C} \rightarrow \mathbb{R}^{512}$ to provide substantial identity similarity to the target image:

$$\mathcal{L}_{ID} = 1 - \frac{\mathcal{F}^n(\mathbf{I_0}) \cdot \mathcal{F}^n(\mathbf{I}_\mathcal{R})}{||\mathcal{F}^n(\mathbf{I_0})||_2 \cdot ||\mathcal{F}^n(\mathbf{I}_\mathcal{R})||_2} \tag{6.6}$$

Along with the abstract level supervision provided by identity features, we also optimize for intermediate activations of the face recognition network to ensure mid-level perceptual information can be reconstructed as well:

$$\mathcal{L}_{per} = \sum_{j}^{n} \frac{||\mathcal{F}^j(\mathbf{I_0}) - \mathcal{F}^j(\mathbf{I}_\mathcal{R})||_2}{H_{\mathcal{F}^j} \cdot W_{\mathcal{F}^j} \cdot C_{\mathcal{F}^j}} \tag{6.7}$$

BRDF-GAN Regularization: Rasterization results in various pixels and facial areas not being used during optimization, while shading may be imperfect in complex illumination environments, both of which introduce noise and environmental features to our albedo and normals during $\mathbf{W}$'s optimization. To avoid such artifacts, we follow the $\mathbf{W}$ initialization protocol introduced in [214], and on top, we apply an L_2 constraint to make sure $\mathbf{W}$'s values do not significantly deviate from their feasible space.

Shape Regularization: We follow the literature [10] in constraining the shape parameters weighted by their inverse eigenvalues: $\mathcal{L}_s = ||\mathbf{p}_s||^2_{\mathbf{\Sigma}_s^{-1}}$ and $\mathcal{L}_e = ||\mathbf{p}_e||^2_{\mathbf{\Sigma}_e^{-1}}$, where $\mathbf{\Sigma}$ denotes a diagonal matrix with the eigenvalues.

Overall loss: It is defined as follows, where all λ's are hyper-parameters suitably chosen to balance the different loss terms during optimization:

$$\begin{aligned}\mathcal{L} =& \lambda_1\mathcal{L}_{lan} + \lambda_2\mathcal{L}_{ph} + \lambda_3\mathcal{L}_{ID} + \lambda_4\mathcal{L}_{per} \\ &+ \lambda_5\mathcal{L}_{\mathbf{W}} + \lambda_6\mathcal{L}_s + \lambda_7\mathcal{L}_e,\end{aligned} \tag{6.8}$$

BRDF-GAN Tuning

Recent works show that $\mathbf{W}$-based inversion cannot fully recover the target image [242]. One approach is to optimize the extended latent space $\mathbf{W}+$ [185, 247], however, this inhibits latent code manipulations [242], while, in our experiments, ends up in an outright

copy of the target image into the albedo, also inhibiting the re-rendering of the avatar. On the other hand, Pivotal Tuning Inversion (PTI) [242] Fine-tune the target image's generator after finding and freezing $\mathbf{W}$. We find this approach much more accurate in our case, as only the visible texture parts produce gradients and are optimized. Again, given imperfect illumination optimization and shape fitting, PTI transfers high-frequency noise, illumination, and non-skin features on the albedo when tuning the whole generator.

Our experiments show that during fine-tuning, the first layers of BRDF-GAN change the color of the albedo but absorb illumination, the middle layers change the mesostructure of the skin, while the last layers change fine details but absorb noise from the target image. Therefore, we found that it is optimal only to tune the four middle layers, while keeping the rest frozen. We use the proposed LPIPS $\mathcal{L}_{LP}$ [243], using a pre-trained VGG network [248], and photometric $\mathcal{L}_{ph}$ losses. Moreover, we add two regularization losses often used in inverse rendering [224]: i) a horizontal flip loss $\mathcal{L}_{flip}$, and ii) a chromaticity loss $\mathcal{L}_{\kappa}$, in order to restrict the fine-tuning to add shadows and highlights, the rendering does not reproduce that, to the albedo:

$$\mathcal{L}_{PTI} = \lambda_8 \mathcal{L}_{LP} + \lambda_9 \mathcal{L}_{ph} + \lambda_{10} \mathcal{L}_{flip} + \lambda_{11} \mathcal{L}_{\kappa} \tag{6.9}$$

where λ's are hyperparams chosen before the fine-tuning.

6.4 Experiments

6.4.1 Implementation Details

Our generator code builds on the public repository of StyleGAN2-ADA [16]. The branches follow the same architecture and are connected through concatenation. For the differentiable rendering, we use a Blinn-Phong shader on Pytorch3D [50] following [24]. Our fitting code requires on average 50 seconds on a machine with one NVIDIA 2080 GPU. We run the fitting for 200 iterations for inversion (Sec. 6.3.4) and 20 iterations for tuning (Sec. 6.3.4). Detailed loss parameters are included in the supplemental.

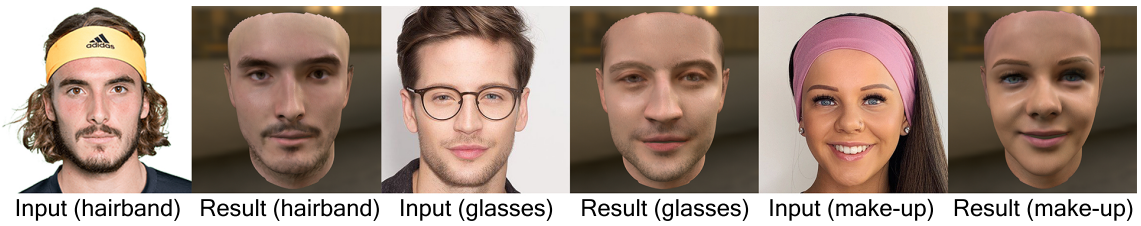


Figure 6.6: Results of our method on challenging examples.

6.4.2 Single-Image Reconstruction

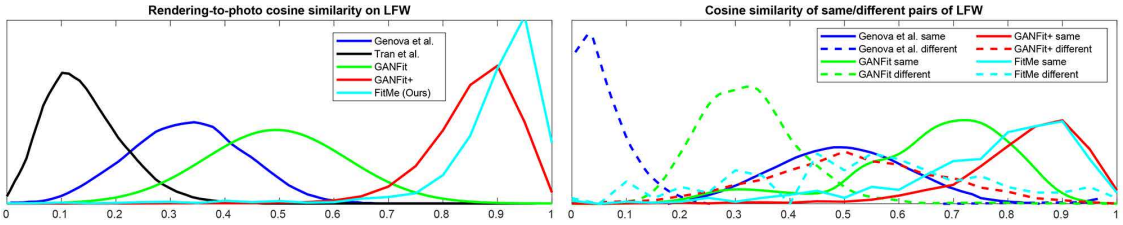


Figure 6.7: Left: Cosine similarity distributions between real image and rendered reconstruction of LFW [249], using VGG-Face [250]. Right: Cosine similarity distribution between same and different pairs of reconstructions. Compared with [19, 211, 216, 219].

We evaluate the identity preservation of our single-image reconstruction by comparing our results in the LFW dataset [249] with previous literature [19, 211, 216, 219]. Using a pre-trained face recognition network [250], we measure the cosine similarity between real image and rendering, and plot the results in Fig. 6.7, which clearly illustrates the effectiveness of our method. We additionally plot the cosine similarity of between reconstruction pairs of the same identity versus different identity. Moreover, in Fig. 6.8, we show a qualitative comparison with prior work on single-image reconstruction and in Fig. 6.9, with single-image reflectance acquisition methods [23, 24, 188, 223]. Finally, in Fig. 6.6 we show our results on challenging examples. All the above show our method's ability to preserve the identity of the subject and produce high-quality shape and reflectance.

Additionally, to show the capabilities of the learned latent space, we perform interpolations between different fittings, which we visualize in Fig. 6.10, by linearly blending their parameters. Our model is able to transition smoothly between fittings, even after generator tuning. In supplemental, we also show PCA-based editing of the latent space.

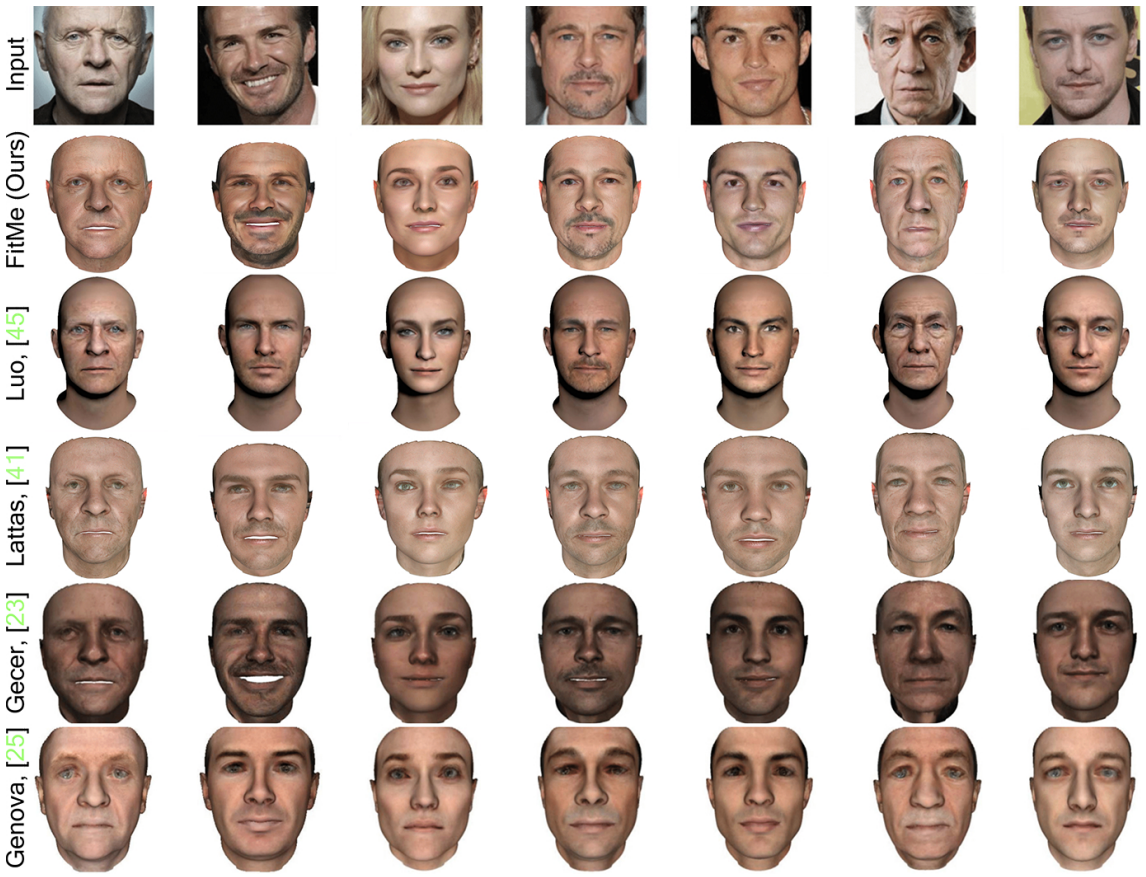


Figure 6.8: Comparison with single-image reconstruction methods [19, 24, 185, 245] Rows 3,5,6 acquired from [185] benchmark. FitMe achieves higher likeness and acquires relightable reconstructions.

6.4.3 Multi-Image Reconstruction

So far, we have been assuming a single target image, however, our method can be used as is for multi-view reconstruction. To do so, we optimize along a batch of N images $\mathbf{I}_{0_i}, i = 0, 1, \ldots, N$. For each batch, we separately optimize the camera $\mathbf{p}_{e_i}$, and illumination $\mathbf{p}_{l_i}$ parameters, while optimizing a single latent vector $\mathbf{W}$, and shape vector $\mathbf{p}_s$. The expression vector $\mathbf{p}_e$ can be varied if needed. Moreover, we only use one instance of the generator $\mathcal{G}$. During optimization, we average the loss across the batch.

We find that a frontal and two side images produce high quality reconstructions that resemble facial capture. Fig. 6.5 shows a comparison of our 3-image reconstruction, and our 1-image reconstruction, with a Light Stage, captured Digital Emily [244], and prior work [24, 223, 224]. As can be seen, our method can successfully be used for fast shape and reflectance acquisition from multi-view sets. Finally, Fig. 6.1 and multiple supplemental figures show examples of our reconstruction using unconstrained mobile phone images.

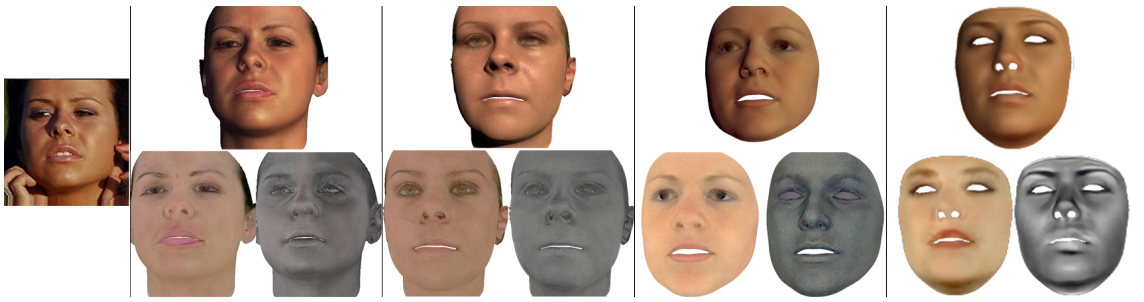


Figure 6.9: Comparison with single-image reflectance acquisition. Top row shows re-rendering and bottom shows rasterized diffuse and specular albedo. From left to right: i) input image, ii) FitMe (ours) using face model, iii) AvatarMe [23], iv) AlbedoMM [223], and v) Dib et al. [224]. Our powerful model GAN accurately reconstructs reflectance in much higher detail than any other method.

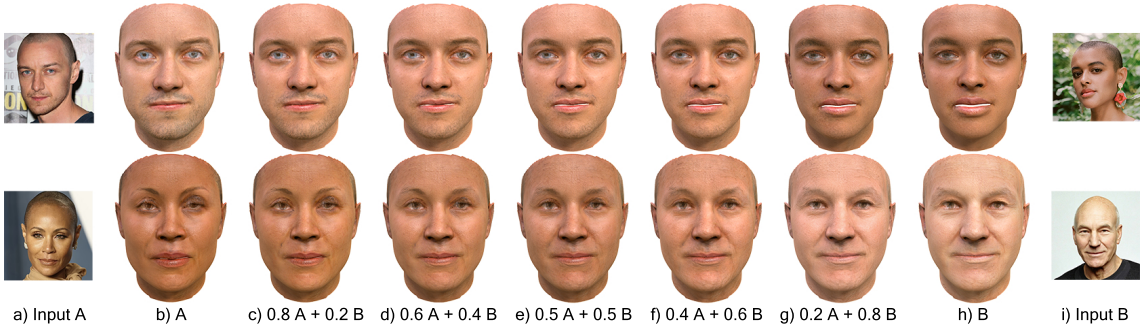


Figure 6.10: Interpolation of the latent parameters between fittings. For each row, we interpolate the latent space $\mathbf{W}$, and the shape parameters $\mathbf{p}_s$, $\mathbf{p}_e$, between the left-most (A) fitting and the rightmost fitting (B), as a linear combination. The smoothly interpolated results show that BRDF-GAN learns a meaningful latent space.

6.4.4 Dataset Augmentation

Our dataset augmentation achieves comparable performance to physics-based skin models [28, 238], without requiring the calculation of such complex interactions. Moreover, compared to the more practical method [239], we avoid inverse-rendering-based noise and achieve similar results in 2.5 seconds per 1024×1024 albedo map, compared to the 46.1 seconds of [239]. We provide a comparison in the supplemental materials.

We perform a comparison of our proposed histogram matching albedo augmentation against a LightStage captured melanin-hemoglobin manipulation method [28] and a melanin-hemoglobin manipulation based on inverse rendering [239]. Our method achieves comparable performance to physics-based skin models [28, 238], without requiring the calculation of such complex interactions, as shown in Fig. 6.11.

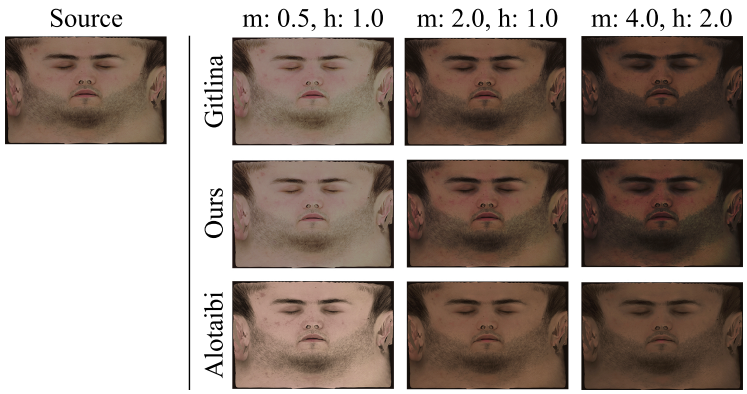


Figure 6.11: Skin tone augmentation comparisons, with adjusted melanin m and hemoglobin h, between (top) LightStage-measured look-up table melanin-hemoglobin of Gitlina et al. [28], (middle) our proposed masked histogram matching and (bottom) inverse-rendering-based manipulation of Alotaibi and Smith [223].

6.5 Model Manipulation

When trained with enough data, the latent space **W** of a style-based generator, can be analyzed with PCA [251]. We expect the first principal components to expose interpretable controls over features of the reconstruction. In Fig. 6.12 we show how the first three components correspond roughly to the skin tone (given our augmentation), gender and age variations. These show that our model learns a meaningful latent space and enable us to perform semantic manipulations directly on the reflectance UV maps.

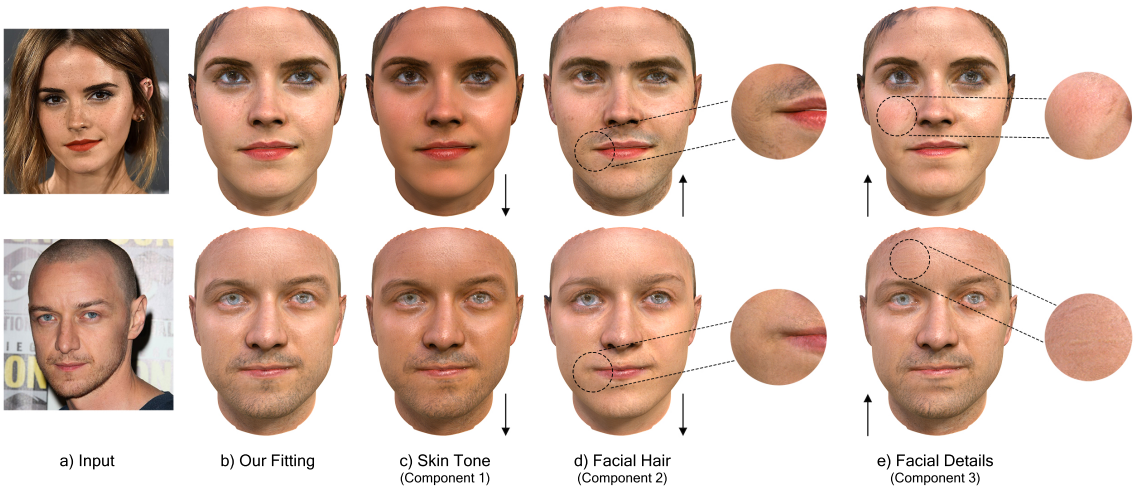
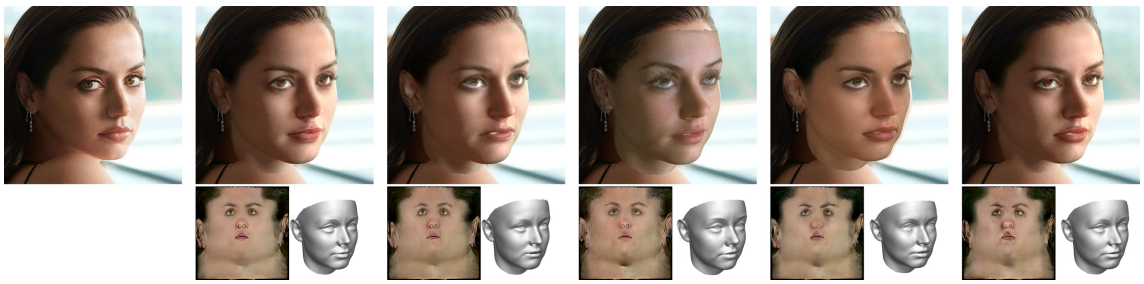


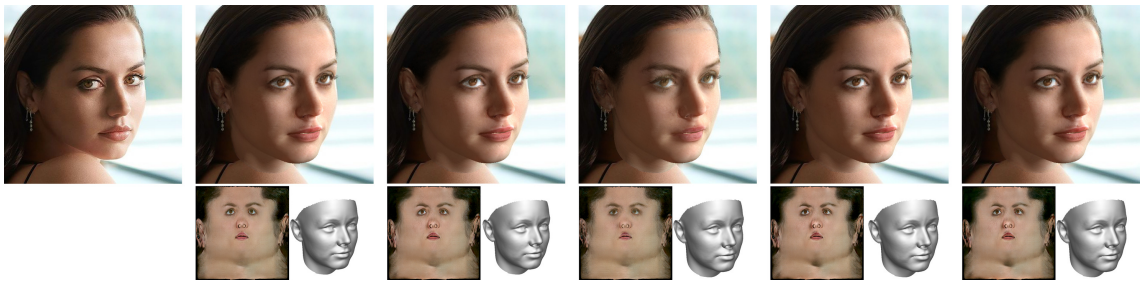
Figure 6.12: Manipulation of a BRDF-GAN projection. By performing PCA on the latent space of BRDF-GAN, we can manipulate the most critical components, which roughly correspond to c) skin tone, d) gender, and e) details (wrinkles, freckles). The arrows indicate the direction of the manipulation of the PCA component.

To better show the meaningful latent representation of facial reflectance that our model learns, we perform interpolations between fittings, which we visualize in Fig. 6.10. Specif-



a) Input b) Ours c) no $\mathcal{L}_{ID}$ d) no $\mathcal{L}_{per}$ e) no $\mathcal{L}_{ph}$ f) no $\mathcal{L}_{\mathbf{W}}$

Figure 6.13: Ablation study on the losses of the **GAN inversion** optimization. For each example, we remove the shown loss from the optimization, and show the rendered result $\mathbf{I}_R$ (top), the diffuse albedo $\mathbf{A}_D$ (bottom-left) and the shape $\mathbf{S}$ (bottom-right). It is apparent how all the proposed losses are required, in order to obtain facial shape and reflectance with high identity and visual similarity while also maintaining an albedo without scene illumination.



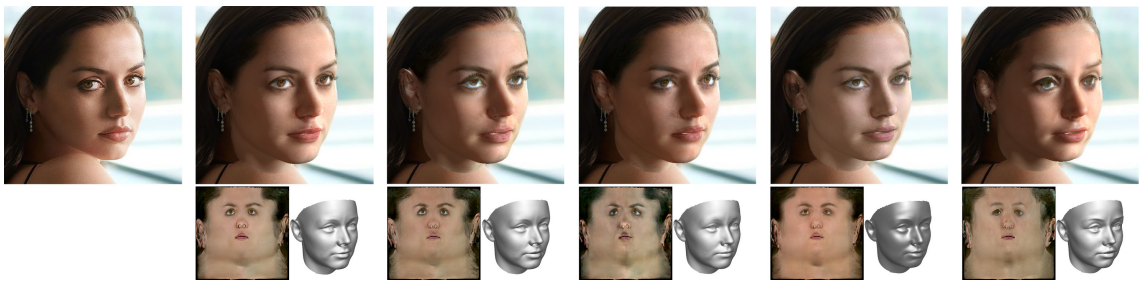
a) Input b) Ours c) no $\mathcal{L}_{ph}$ d) no $\mathcal{L}_{LP}$ e) no $\mathcal{L}_{flip}$ f) no $\mathcal{L}_{\kappa}$

Figure 6.14: Ablation study on the losses of the **GAN tuning** optimization. For each example, we remove the shown loss from the optimization, and show the rendered result $\mathbf{I}_R$ (top), the diffuse albedo $\mathbf{A}_D$ (bottom-left) and the shape $\mathbf{S}$ (bottom-right). Despite most cases showing a highly optimized rendered result, without the proposed losses combination, the albedo absorbs residual scene illumination, not captured by our rendering, in cases c), e), f), Also, since we do not optimize the shape $\mathbf{S}$ during tuning, the shape remains the same in these examples.

ically, we concurrently fit our model and perform generator tuning to two images (A and B), and generate reconstructions by linearly blending their latent parameters, namely the $\mathbf{W}$ space of the generator and the shape $\mathbf{p}_s$ and expressions parameters $\mathbf{s}_e$. Our model is able to generate realistic-looking latent blends of the fitted examples, and transition smoothly between them, even when finetuned.

6.6 Ablation Study

We perform three ablation studies to validate our architectural and optimization choices. The first ablation, shown in Fig. 6.13, shows the effect of the losses used in the GAN



a) Input b) Ours c) no GAN tuning d) **W**+, [185] e) Pytorch3D [50] f) **z** fitting

Figure 6.15: Ablation study on **method choices**. For each example, make the shown change during the optimization, and show the rendered result $\mathbf{I}_R$ (top), the diffuse albedo $\mathbf{A}_D$ (bottom-left) and the shape $\mathbf{S}$ (bottom-right). c), e), and f) show the importance of $\mathbf{W}$ fitting, our photorealistic facial shading, and the GAN tuning. d) shows that $\mathbf{W}+$ optimization, as performed by [185], achieves great likeness in the albedo but does not maintain the semantics of the facial reflectance.

inversion part of our method. For each example, we remove one of the following losses: the identity loss $\mathcal{L}_{ID}$, the perceptual loss $\mathcal{L}_{per}$, the photometric loss $\mathcal{L}_{ph}$ and the $\mathbf{W}$ regularization loss $\mathcal{L}_{\mathbf{W}}$. Removing the landmark loss $\mathcal{L}_{lan}$ or the shape and expression regularization losses $\mathcal{L}_s, \mathcal{L}_e$, fails the optimization, as the shape fitting is misplaced.

Moreover, in Fig. 6.14, we perform a similar ablation for the losses used during the GAN tuning part of our method. The removal of the photometric $\mathcal{L}_{ph}$ and LPIPS $\mathcal{L}_{LP}$ losses, shows their importance in achieving higher likeness to the input image, The removal of the flip $\mathcal{L}_{flip}$ and chromaticity $\mathcal{L}_{\kappa}$ losses shows their need to maintain an albedo without scene illumination, when compared to vanilla pivotal tuning [242].

Finally, in Fig. 6.15, we show an ablation study on choices regarding the optimization method. In Fig. 6.15 c), we show the optimized result, without our GAN tuning, Fig. 6.15 d) shows an extended latent space $\mathbf{W}+$ optimization, rather than $\mathbf{W}$, which means that each slice of $\mathbf{W}$ is separately optimized for each layer of the generator network and is suggested by [185]. Such an approach also achieves great likeness in the optimized rendering, but disturbs the statistics of the generator, so that scene illumination is absorbed by the albedo, and the network cannot be accurately manipulated (Fig. 6.12). Similar observations are also reported by [242]. In Fig. 6.15 e) we show the full optimization (inversion and tuning), while using the Pytorch3D [50] implementation of Phong shader, which also only supports the diffuse albedo $\mathbf{A}_D$, no spatial variation in specular roughness and no subsurface scattering optimization. Fig. 6.15 f) shows the optimization of the latent variable $\mathbf{z}$, which is passed through a mapping network before being given to the generator.

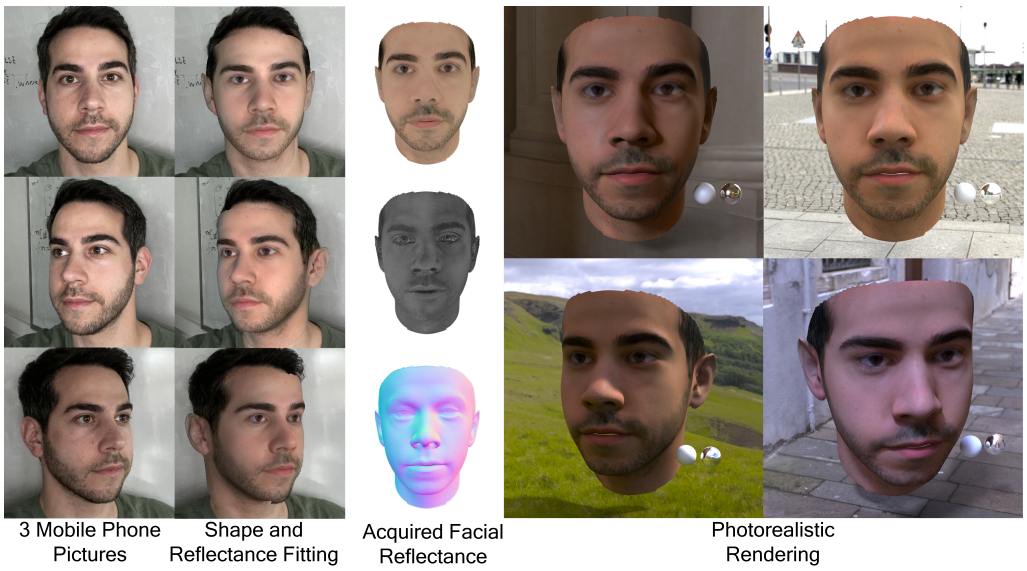


Figure 6.16: Additional multi-view capture results using our method. From left to right: a) three input images, b) rendered fitting using our method, c) diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and normals $\mathbf{N}_S$, d) rendered shape $\mathbf{S}$ and e) rendered results in various environments.

Such a fitting is more constrained and cannot reach a satisfying identity similarity.

6.7 Additional Results

We present additional results of multi-view capturing in Figs. 6.16,6.17,6.18. In each case we ask the subject to take three unconstrained mobile phone images, from the front and the side. We show that these are enough to create an accurate photorealistic avatar of the subject, using our method. In total, capturing takes less than 10 seconds, and processing less than a minute.

Additional to the quantitative comparison on identity similarity presented in the main manuscript, we also perform and present here a quantitative comparison of shape reconstruction. Following the benchmark proposed by GANFit [19], we reconstruct the 53 subjects from the MICC Florence 3D Faces dataset (MICC) [252] with our facial shape and measure their distance (in nm) from the dataset shapes, using the open-source benchmark code of GANFit [19]. MICC includes three subcategories, “cooperative”, “indoor” and “outdoor”, on which we report findings separately. Instead of 5 images [19], we only use three random images from each video for the reconstruction. We then perform dense alignment and measure the point-to-plane distance and present our findings in Tab. 6.1.

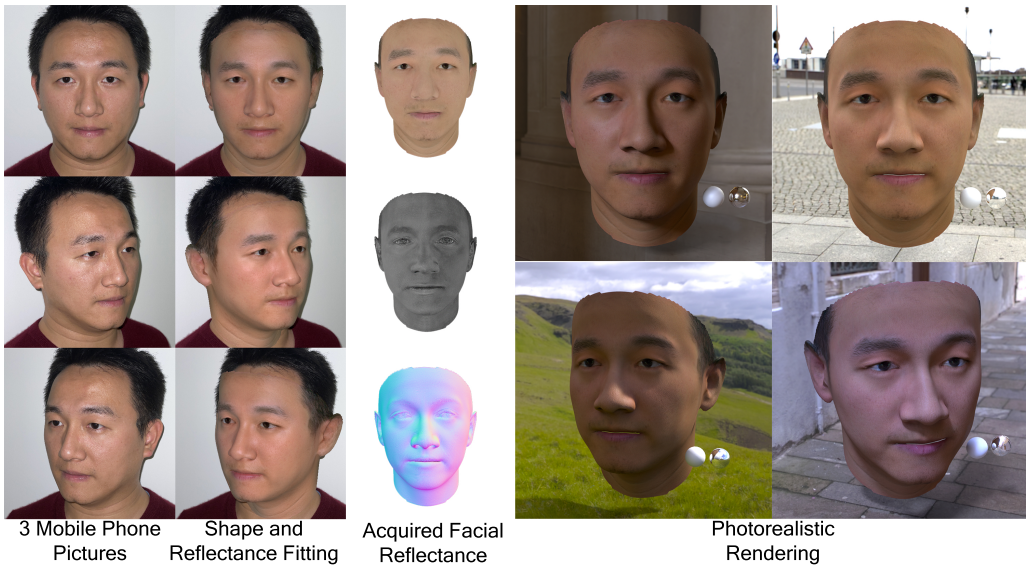


Figure 6.17: Additional multi-view capture results using our method. From left to right: a) three input images, b) rendered fitting using our method, c) diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and normals $\mathbf{N}_S$, d) rendered shape $\mathbf{S}$ and e) rendered results on various environments.

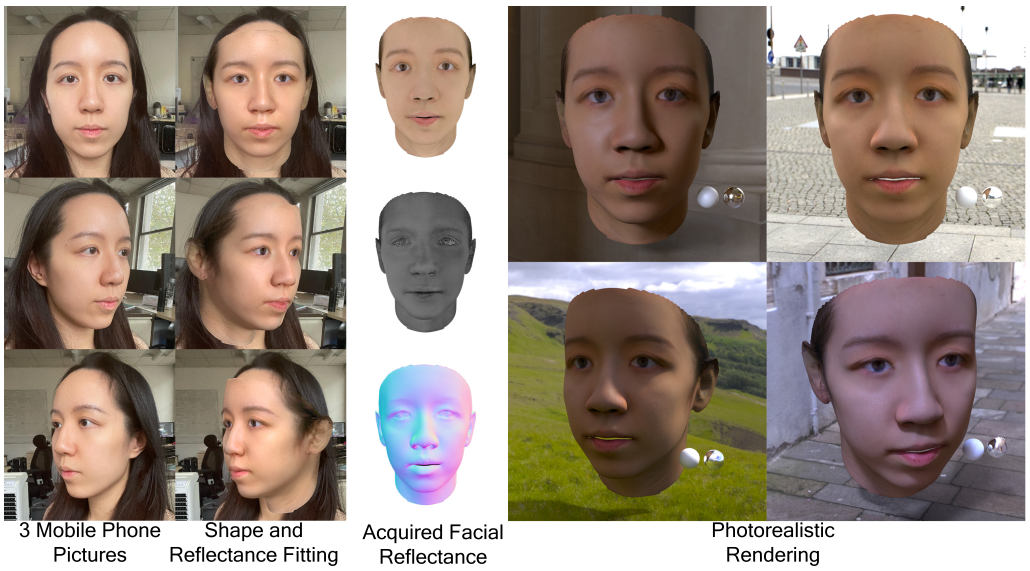


Figure 6.18: Additional multi-view capture results using our method. From left to right: a) three input images, b) rendered fitting using our method, c) diffuse albedo $\mathbf{A}_D$, specular albedo $\mathbf{A}_S$ and normals $\mathbf{N}_S$, d) rendered shape $\mathbf{S}$ and e) rendered results in various environments.

Method	Cooperative Mean ± Std.	Indoor Mean ± Std.	Outdoor Mean ± Std.
Tran et al. [211]	1.93 ± 0.27	2.02 ± 0.25	1.86 ± 0.24
Booth et al. [154]	1.82 ± 0.29	1.85 ± 0.22	1.63 ± 0.16
Genova et al. [219]	1.50 ± 0.14	1.50 ± 0.11	1.48 ± 0.11
GANFit [19]	0.95 ± 0.10	0.94 ± 0.10	0.94 ± 0.10
GANFit++ [216]	0.94 ± 0.17	0.92 ± 0.14	0.94 ± 0.19
Fast-GANFit [216]	1.11 ± 0.25	0.98 ± 0.15	1.16 ± 0.18
Ours	0.95 ± 0.18	0.97 ± 0.20	0.98 ± 0.21

Table 6.1: Quantitative benchmark comparison of shape reconstruction, on the MICC Florence 3D Faces dataset, using point-to-plane distance, as used in the open-source benchmark of [216]. we compare our results with the reported results of Tran et al. [211], Booth et al. [154], Genova et al. [219], GANFit [19], GANFit++ [216] and Fast-GANFit [216].

Our method performs similarly in shape reconstruction to GANFit [19] and GANFit++ [216], and significantly better when compared to previous methods [154, 210, 245] and Fast-GANFit [216]. GANFit reconstructions are slightly closer to the ground truth shapes. This can be explained by the fact that we optimize both the shape mesh and the texture normals, and thus a part of the shape information is explained in the normals domain rather than in the actual shape space. Furthermore, for a fair comparison, we only compared the meshes, potentially missing details. Finally, as shown in the main manuscript, our method scores first in identity similarity while acquiring relightable reflectance textures.

6.8 Implementation Details

As briefly described in the main manuscript, FitMe implementation builds on the public repository of StyleGAN2-ADA [16], in Pytorch, both for the generator and the discriminator. However, we have made the following changes. a) The generator $\mathcal{G}$ is branched on the last convolutional blocks, which is achieved by feeding the output of the last single-branch layer, to 3 different copies of the last branched module. The generator follows the skip-connections architecture of [16]. b) The discriminator is also branched and follows the resnet architecture of [16]. The output of each branch is concatenated and fed to the last convolutional block and the fully connected layers.

For the differentiable photorealistic rendering, we create a shader based on the Blinn-Phong model [58], following AvatarMe++[24]. The model is implemented by extending the shader and shading classes in Pytorch3D [50].

Our optimization is based on a number of hyperparameters λ_i, each corresponding to a loss described in the paper. We find these empirically and present them below. For the inversion, we use a learning rate of $l_{inv} = 1 \times 10^{-2}$ and for the tuning we use a learning rate of $l_{pti} = 8 \times 10^{-4}$. In Table 6.2 and Table 6.3 we present the values which we find most optimal, when optimizing for crop and rendering at 512×512 pixels, 250 iterations for inversion and 30 iterations for tuning.

hyper-param	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
corresponding loss	$\mathcal{L}_{lan}$	$\mathcal{L}_{ph}$	$\mathcal{L}_{ID}$	$\mathcal{L}_{per}$	$\mathcal{L}_{W}$	$\mathcal{L}_{s}$	$\mathcal{L}_{e}$
value	100	0.5	1.0	25.0	5×10^{-2}	5×10^{-4}	5×10^{-4}

Table 6.2: Hyper-parameter values and their corresponding loss terms, used for the GAN-inversion part of the proposed method.

hyper-param	λ_8	λ_9	λ_{10}	λ_{11}
corresponding loss	$\mathcal{L}_{LP}$	$\mathcal{L}_{ph}$	$\mathcal{L}_{flip}$	$\mathcal{L}_{\kappa}$
value	2.0	0.5	0.8	0.35

Table 6.3: Hyper-parameter values and their corresponding loss terms, used for the GAN-tuning part of the proposed method.

6.9 Limitations

Our method is limited in certain aspects pertaining the data and the fitting process. The data limitations include a) the imbalance of features in the training data, which we attempt to overcome by the proposed augmentation and generator tuning. Moreover, the reflectance data are captured with the assumption of skin-like materials and therefore the eyes exhibit noisy reflections. Finally, there exists ambiguity between skin-tone and illumination intensity during fitting, which could be alleviated by combining our method with the findings of TRUST [225] in future work.

6.10 Conclusion

In this paper we introduced FitMe, a method that is able to produce highly accurate, renderable human avatars based on a single or multiple "in-the-wild" images. To achieve this, we introduced a deep facial reflectance model that consists of a multi-branched style-based GAN and a PCA-based shape model, as well as an easy skin-tone augmentation

method. Moreover, we presented a novel iterative optimization procedure that is based on a differential rendering process. As we showed in a series of experiments, our method helps to bridge the uncanny valley and creates pleasing results, directly renderable in common renderers.

Chapter 7

Conclusion

Contents

7.1 Future Work 143

In this Thesis, we presented several improvements in facial scanning, in the direction of simplifying the process and enabling a more significant number of subjects, and introduced a scanned dataset of high-resolution facial reflectance. Using these data, we introduced a novel method for acquiring facial reflectance from "in-the-wild" images, and created a deep 3DMM with high-resolution facial reflectance textures, both of which enable the fast and accurate acquisition of relightable facial avatars.

In Chapters 3 and 4, we introduced improvements in facial reflectance capture with scanning devices. Specifically, we improved the automation and efficiency of the Imperial College Light Stage, and made its diffuse-specular separation algorithm multi-view, while capturing a large dataset of facial reflectance. Then, we introduced two novel facial scanners composed of commodity devices, that are portable and significantly cheaper than the Light Stage. Despite this, we show that they are capable of capturing high-quality diffuse and specular facial reflectance, democratizing facial capture for wider audiences.

In Chapters 5 and 6, we introduced three novel methods of facial reflectance and shape acquisition from unconstrained facial images. AvatarMe and AvatarMe^{++} transform captured and inferred facial textures into high-resolution relightable diffuse and specular reflectance textures, that can be rendered as-is in standard rendering engines. On the other hand, FitMe is a deep 3DMM capable of producing high-resolution facial reflectance, which can

also be fit on one or more facial images, taken without constraints. All these methods pave the way for fully flexible facial digitization, so that people can capture themselves with personal devices, i.e. mobile phones, and insert themselves in digital environments, such as games, telepresence, and virtual or augmented reality.

7.1 Future Work

In this Thesis, we present several improvements around two ways of acquiring facial avatar assets, that can be photorealistically rendered in standard rendering engines. On the one hand, we explore and further simplify the requirements for the previously cumbersome facial reflectance and geometry scanning, and on the other hand, we develop deep-learning models and fitting methods, for inferring such assets from arbitrary facial images. Despite the novelties we introduce, various aspects of these methods could be further improved, which we describe below, as well as new optimization and machine learning methods that could be utilized in this direction.

Despite the improvements introduced in this Thesis, in capturing facial shape and reflectance, one still must purchase and combine several devices to construct the proposed facial scanner. It is perhaps possible and desirable to further simplify the capturing process to require a single mobile device and no additional hardware. The complexity of human skin and the presence of arbitrary illumination and shadows, in non-controlled environments and setups, make this attempt challenging, however, it is an exciting and helpful direction.

Moreover, the work in this Thesis focuses on facial and head skin acquisition. However, several other body parts are required for an accurate digital human representation. Firstly, several head parts need particular attention due to their different nature, especially the eyes and the hair. Even though several works exist for both, there still needs to be a unified approach. In the same manner, although there has been tremendous research in full body capture, such works place less importance on the realisticity of the acquired head. Again, a unified approach would be of great interest for future work.

Moreover, in this work, we consider computer graphics and machine learning methods

to solve the human digitization problem. The components selected for our methods produce explicit representations of facial shape and reflectance, that are directly usable in standard rendering engines and applications. Recently, Neural Radiance Fields (NeRFs) [253], initially designed for photorealistic novel-view synthesis from a set of static views of a scene, have paved the way for a new graphics pipeline. Specifically, NeRFs optimize several MLPs that implicitly define a scene and can be queried to provide samples for the volumetric rendering of that scene. In addition, follow-up works have shown NeRFs capable of capturing and rendering humans in static scenes, e.g. [254]. Without having to capture different material properties explicitly, such methods show potential in capturing and rendering skin and non-skin areas simultaneously , as well as accessories, with accurate diffuse and view-dependent specular rendering. Some works have even modeled the diffuse and specular components of facial assets captured with methods, such as those presented in this thesis [35, 228, 255, 256]. Even more, various follow-up works have explored dynamic NeRFs [254, 257, 258], where additional MLPs learn to transform a conditional canonical space to the implicit space of a NeRF.

Despite the success of such methods in producing impressive renderings, there remain certain drawbacks. Firstly, one can acquire an explicit shape representation of a NeRF space, for example, by using marching cubes [253], or optimizing details on an existing geometry [33, 259]. However, at the writing of this Thesis, it remains a challenge to extract shape or reflectance at the details and quality of current explicit methods. Secondly, standard NeRF rendering requires significant rendering times, although recent works manage to reduce these requirements [65, 260, 261]. Thirdly, relighting optimized scenes remains a significant research challenge, especially for complex materials such as human skin or hair, and comes as a trade-off to rendering speed. Finally, the available rendering engines or applications are yet to support rendering implicit models broadly, reducing their usability. Despite the above, Neural Radiance Fields provide a fascinating research topic, and a powerful alternative for certain parts of the existing graphics pipelines, and its possibilities in avatar reconstruction should certainly be examined.

Last but not least, there have recently been tremendous improvements in image generation, using Denoising Probabilistic Diffusion Models (Diffusion Models) [262, 263], which can be shown to be more capable than GANs in multiple scenarios [264]. Diffusion Models learn

a degradation function to add noise to an image gradually, and a function that gradually removes this noise to generate an image from the training distribution. Typically, a UNet model can learn this transformation [262], while this diffusion process can be performed at the latent space of a GAN [263]. This architecture enables more stable training while making it easy to condition the diffusion process on embeddings of modes such as text [263], while also enabling the diffusion to be guided by the gradients of another network, e.g. a classifier [264]. While the denoising process is significantly slower (1-2 minutes) than a forward pass in a large GAN model (less than one second), follow-up works in implicit Diffusion Models [265] speed up this process (10-20 seconds). In this work, we mainly base our proposed inference methods on the well-studied PCA for shape modeling, and the robust Generative Adversarial Networks for texture modeling. Given the possibilities of Diffusion Models and the fast-paced research of the community on the topic, it is an indeed promising future research direction is to investigate improved facial acquisition methods, based on Diffusion Models for facial texture acquisition as well as editing based on text and other attributes. An early attempt at such a method is Relightify [36], which uses 3DMM fitting and multi-modal diffusion models, for facial reflectance acquisition from a single image, trained on data such as those captured in Chapter 3 and Chapter 4 and based on ideas from AvatarMe^{++} and FitMe from Chapter 5 and Chapter 6.

Bibliography

[1] M. Mori, "Bukimi no tani (the uncanny valley)," *Energy*, vol. 7, no. 4, pp. 33–35, 1970.

[2] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH*, pp. 145–156, ACM Press, 2000.

[3] P. Debevec, "The light stages and their applications to photoreal digital actors," *SIGGRAPH Asia*, vol. 2, no. 4, 2012.

[4] W.-C. Ma, T. Hawkins, P. Peers, C.-F. Chabert, M. Weiss, and P. Debevec, "Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*, EGSR'07, pp. 183–194, Eurographics Association, June 2007.

[5] A. Ghosh, G. Fyffe, B. Tunwattanapong, J. Busch, X. Yu, and P. Debevec, "Multiview face capture using polarized spherical gradient illumination," *ACM Transactions on Graphics*, Dec. 2011.

[6] 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, 1999.

[7] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2, no. 11, pp. 559–572, 1901.

[8] E. Antonakos, J. Alabort-i Medina, G. Tzimiropoulos, and S. Zafeiriou, "Hog active appearance models," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 224–228, IEEE, 2014.

[9] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3D Morphable Model Learnt From 10,000 Faces," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[10] J. Booth, E. Antonakos, S. Ploumpis, G. Trigeorgis, Y. Panagakis, and S. Zafeiriou, "3D Face Morphable Models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 48–57, 2017.

[11] W. A. P. Smith, A. Seck, H. Dee, B. Tiddeman, J. B. Tenenbaum, and B. Egger, "A Morphable Face Albedo Model," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020.

[12] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, "Handwritten digit recognition with a back-propagation network," *Advances in neural information processing systems*, vol. 2, 1989.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 27* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 2672–2680, Curran Associates, Inc., 2014.

[14] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.

[15] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," *arXiv:1812.04948 [cs, stat]*, Mar. 2019.

[16] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12104–12114, 2020.

[17] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and

improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

[18] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-Resolution Image Synthesis and Semantic Manipulation With Conditional GANs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[19] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, "Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1155–1164, 2019.

[20] A. Lattas, M. Wang, S. Zafeiriou, and A. Ghosh, "Multi-view facial capture using binary spherical gradient illumination," in *ACM SIGGRAPH 2019 Posters*, pp. 1–2, 2019.

[21] A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera, G. Chawla, and A. Ghosh, "Desktop-based high-quality facial capture for everyone," in *SIGGRAPH Talks*, 2022.

[22] A. Lattas, Y. Lin, J. Kannan, E. Ozturk, L. Filipi, G. Guarnera, G. Chawla, and A. Ghosh, "Practical and scalable desktop-based high-quality facial capture," in *Proceedings of the European Conference on Computer Vision*, 2022.

[23] A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "AvatarMe: Realistically renderable 3d facial reconstruction" in-the-wild"," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 760–769, 2020.

[24] A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, A. Ghosh, and S. Zafeiriou, "AvatarMe++: Facial shape and BRDF inference with photorealistic rendering-aware GANs," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2021.

[25] A. Lattas, S. Moschoglou, S. Ploumpis, B. Gecer, J. Deng, and S. Zafeiriou, "Fitme: Deep photorealistic 3D morphable model avatars," in *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8629–8640, June 2023.

[26] P. Tzirakis, A. Papaioannou, A. Lattas, M. Tarasiou, B. Schuller, and S. Zafeiriou, "Synthesising 3D facial motion from "in-the-wild" speech," in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pp. 265–272, IEEE, 2020.

[27] B. Gecer, A. Lattas, S. Ploumpis, J. Deng, A. Papaioannou, S. Moschoglou, and S. Zafeiriou, "Synthesizing coupled 3D face modalities by trunk-branch generative adversarial networks," in *European conference on computer vision*, pp. 415–433, Springer, 2020.

[28] Y. Gitlina, G. C. Guarnera, D. S. Dhillon, J. Hansen, A. Lattas, D. Pai, and A. Ghosh, "Practical measurement and reconstruction of spectral skin reflectance," in *Computer Graphics Forum*, pp. 75–89, Wiley Online Library, 2020.

[29] J. Deng, J. Guo, J. Yang, A. Lattas, and S. Zafeiriou, "Variational prototype learning for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11906–11915, 2021.

[30] J. Guo, J. Deng, A. Lattas, and S. Zafeiriou, "Sample and computation redistribution for efficient face detection," in *International Conference on Learning Representations*, 2021.

[31] S. Li, U. Waheed, M. Bahshwan, L. Z. Wang, L. M. Kalossaka, J. Choi, F. Kundrak, A. Lattas, S. Ploumpis, S. Zafeiriou, *et al.*, "A scalable mass customisation design process for 3D-printed respirator mask to combat covid-19," *Rapid Prototyping Journal*, 2021.

[32] Y. Miao, A. Lattas, J. Deng, J. Han, and S. Zafeiriou, "Physically-based face rendering for NIR-VIS face recognition," in *Advances in Neural Information Processing Systems* (A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, eds.), 2022.

[33] X. Ren, A. Lattas, B. Gecer, J. Deng, C. Ma, X. Yang, and S. Zafeiriou, "Facial geometric detail recovery via implicit representation," in *2023 17th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2023)*, IEEE, 2023.

[34] J. Guo, J. Yu, A. Lattas, and J. Deng, "Perspective reconstruction of human faces by joint mesh and landmark regression," *European Conference on Computer Vision (ECCV) Workshops*, 2022.

[35] S. Galanakis, B. Gecer, A. Lattas, and S. Zafeiriou, "3DMM-RF: Convolutional radiance fields for 3D face modeling," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3536–3547, 2023.

[36] F. P. Papantoniou, A. Lattas, S. Moschoglou, and S. Zafeiriou, "Relightify: Relightable 3D faces from a single image via diffusion models," in *Proc. IEEE International Conference on Computer Vision*, 2023.

[37] A. Ghosh, C. Kampouris, A. Lattas, and S. Zafeiriou, "Image processing for diffuse-specular separation," 2021. US Patent 10,964,099.

[38] S. Zafeiriou, A. Lattas, S. Moschoglou, S. Ploumpis, and B. Gecer, "Three-dimensional facial reconstruction." UK Patent App GB2593441A (Pending).

[39] G. Patow and X. Pueyo, "A Survey of Inverse Rendering Problems," *Computer Graphics Forum*, vol. 22, no. 4, 2003.

[40] T. Akenine-Möller, E. Haines, and N. Hoffman, *Real-Time Rendering*. CRC Press, Jan. 2019.

[41] M. M. Loper and M. J. Black, "OpenDR: An Approximate Differentiable Renderer," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), Lecture noteComments in Computer Science, Springer International Publishing, 2014.

[42] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D Mesh Renderer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3907–3916, 2018.

[43] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, "Unsupervised Training for 3D Morphable Model Regression," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8377–8386, IEEE, June 2018.

[44] S. Liu, T. Li, W. Chen, and H. Li, "Soft Rasterizer: A Differentiable Renderer for Image-Based 3D Reasoning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[45] W. Chen, H. Ling, J. Gao, E. Smith, J. Lehtinen, A. Jacobson, and S. Fidler, "Learning to Predict 3D Objects with an Interpolation-based Differentiable Renderer," in *Advances in Neural Information Processing Systems 32* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 9609–9619, Curran Associates, Inc., 2019.

[46] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable Monte Carlo ray tracing through edge sampling," *ACM Transactions on Graphics*, vol. 37, pp. 222:1–222:11, Dec. 2018.

[47] G. Loubet, N. Holzschuch, and W. Jakob, "Reparameterizing discontinuous integrands for differentiable rendering," *ACM Transactions on Graphics*, vol. 38, pp. 228:1–228:14, Nov. 2019.

[48] K. M. Jatavallabhula, E. Smith, J.-F. Lafleche, C. F. Tsang, A. Rozantsev, W. Chen, T. Xiang, R. Lebaredian, and S. Fidler, "Kaolin: A PyTorch Library for Accelerating 3D Deep Learning Research," *arXiv:1911.05063 [cs]*, Nov. 2019.

[49] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5541–5550, 2017.

[50] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv:2007.08501*, 2020.

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

[52] C. Donner and H. W. Jensen, "A spectral bssrdf for shading human skin.," *Rendering techniques*, vol. 2006, pp. 409–418, 2006.

[53] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pp. 143–150, 1986.

[54] C. Schlick, "An inexpensive brdf model for physically-based rendering," in *Computer graphics forum*, vol. 13, pp. 233–246, Wiley Online Library, 1994.

[55] J. F. Blinn and M. E. Newell, "Texture and reflection in computer generated images," *Communications of the ACM*, vol. 19, pp. 542–547, Oct. 1976.

[56] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," *ACM Transactions on Graphics (ToG)*, vol. 1, no. 1, pp. 7–24, 1982.

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

[58] 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.

[59] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, "Microfacet models for refraction through rough surfaces," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pp. 195–206, 2007.

[60] B. Gecer, J. Deng, and S. Zafeiriou, "Ostec: One-shot texture completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7628–7638, 2021.

[61] X. Wu, J. Zhang, Y. Chen, and X. Huang, "Real-time mid-wavelength infrared scene rendering with a feasible BRDF model," *Infrared Physics & Technology*, vol. 68, pp. 124–133, Jan. 2015.

[62] C. Kampouris, S. Zafeiriou, and A. Ghosh, "Diffuse-Specular Separation using Binary Spherical Gradient Illumination," *Eurographics Symposium on Rendering*, p. 10, 2018.

[63] P.-W. Grassal, M. Prinzler, T. Leistner, C. Rother, M. Nießner, and J. Thies, "Neural head avatars from monocular rgb videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18653–18664, 2022.

[64] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis,"

in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), Lecture noteComments in Computer Science, pp. 405–421, Springer International Publishing, 2020.

[65] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, "Plenoctrees for real-time rendering of neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5752–5761, 2021.

[66] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10318–10327, 2021.

[67] P. P. Srinivasan, B. Deng, X. Zhang, M. Tancik, B. Mildenhall, and J. T. Barron, "Nerv: Neural reflectance and visibility fields for relighting and view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7495–7504, 2021.

[68] M. Niemeyer and A. Geiger, "Giraffe: Representing scenes as compositional generative neural feature fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11453–11464, 2021.

[69] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi, "Efficiently combining positions and normals for precise 3D geometry," *ACM Transactions on Graphics*, vol. 24, pp. 536–543, July 2005.

[70] C. Hernandez, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla, "Non-rigid photometric stereo with colored lights," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2007.

[71] M. Klaudiny and A. Hilton, "High-detail 3d capture and non-sequential alignment of facial performance," in *Proceedings of 3DIMPVT*, 2012.

[72] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 145–156, 2000.

[73] T. Weyrich, W. Matusik, H. Pfister, B. Bickel, C. Donner, C. Tu, J. McAndless, J. Lee, A. Ngan, H. W. Jensen, and M. Gross, "Analysis of human faces using a

measurement-based skin reflectance model," *ACM Trans. Graphics (TOG)*, vol. 25, pp. 1013–1024, July 2006.

[74] W.-C. Ma, T. Hawkins, P. Peers, C.-F. Chabert, M. Weiss, and P. Debevec, "Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pp. 183–194, Eurographics Association, 2007.

[75] A. Ghosh, T. Hawkins, P. Peers, S. Frederiksen, and P. Debevec, "Practical modeling and acquisition of layered facial reflectance," *ACM TOG*, vol. 27, Dec. 2008.

[76] A. Ghosh, G. Fyffe, B. Tunwattanapong, J. Busch, X. Yu, and P. Debevec, "Multiview face capture using polarized spherical gradient illumination," in *ACM Transactions on Graphics (TOG)*, vol. 30, p. 129, ACM, 2011.

[77] G. Fyffe, T. Hawkins, C. Watts, W.-C. Ma, and P. Debevec, "Comprehensive facial performance capture," *Computer Graphics Forum (CGF)*, vol. 30, no. 2, 2011.

[78] C. A. Wilson, A. Ghosh, P. Peers, J.-Y. Chiang, J. Busch, and P. Debevec, "Temporal upsampling of performance geometry using photometric alignment," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 2, p. 17, 2010.

[79] G. Fyffe and P. Debevec, "Single-shot reflectance measurement from polarized color gradient illumination," in *International Conference on Computational Photography (ICCP)*, IEEE, 2015.

[80] C. Kampouris, S. Zafeiriou, and A. Ghosh, "Diffuse-specular separation using binary spherical gradient illumination," in *EGSR (EI&I)*, pp. 1–10, 2018.

[81] G. Fyffe, P. Graham, B. Tunwattanapong, A. Ghosh, and P. Debevec, "Near-Instant Capture of High-Resolution Facial Geometry and Reflectance," *Computer Graphics Forum*, 2016.

[82] 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. 3, pp. 40:1–40:9, 2010.

[83] D. Bradley, W. Heidrich, T. Popa, and A. Sheffer, "High resolution passive facial performance capture," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, p. 41, 2010.

[84] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, R. W. Sumner, and M. Gross, "High-quality passive facial performance capture using anchor frames," *ACM Transactions on Graphics (ACM)*, vol. 30, pp. 75:1–75:10, August 2011.

[85] P. Gotardo, J. Riviere, D. Bradley, A. Ghosh, and T. Beeler, "Practical dynamic facial appearance modeling and acquisition," *ACM Trans. Graph.*, vol. 37, Dec. 2018.

[86] J. Riviere, P. Gotardo, D. Bradley, A. Ghosh, and T. Beeler, "Single-shot high-quality facial geometry and skin appearance capture," *ACM Trans. Graph.*, vol. 39, July 2020.

[87] A. E. Ichim, S. Bouaziz, and M. Pauly, "Dynamic 3D avatar creation from hand-held video input," *ACM Transactions on Graphics*, vol. 34, pp. 45:1–45:14, July 2015.

[88] P. Bérard, D. Bradley, M. Nitti, T. Beeler, and M. Gross, "High-quality capture of eyes," *ACM Transactions on Graphics*, vol. 33, pp. 223:1–223:12, Nov. 2014.

[89] C. Wu, D. Bradley, P. Garrido, M. Zollhöfer, C. Theobalt, M. Gross, and T. Beeler, "Model-based teeth reconstruction," *ACM Transactions on Graphics*, vol. 35, pp. 220:1–220:13, Nov. 2016.

[90] Z. Velinov, M. Papas, D. Bradley, P. Gotardo, P. Mirdehghan, S. Marschner, J. Novák, and T. Beeler, "Appearance capture and modeling of human teeth," *ACM Transactions on Graphics*, vol. 37, pp. 207:1–207:13, Dec. 2018.

[91] T. Beeler, B. Bickel, G. Noris, P. Beardsley, S. Marschner, R. W. Sumner, and M. Gross, "Coupled 3D reconstruction of sparse facial hair and skin," *ACM Transactions on Graphics*, vol. 31, pp. 117:1–117:10, July 2012.

[92] L. Luo, H. Li, and S. Rusinkiewicz, "Structure-aware hair capture," *ACM Transactions on Graphics*, vol. 32, pp. 76:1–76:12, July 2013.

[93] L. Hu, C. Ma, L. Luo, and H. Li, "Robust hair capture using simulated examples," *ACM Transactions on Graphics*, vol. 33, pp. 126:1–126:10, July 2014.

[94] L. Hu, C. Ma, L. Luo, and H. Li, "Single-view hair modeling using a hairstyle database," *ACM Transactions on Graphics*, vol. 34, pp. 125:1–125:9, July 2015.

[95] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[96] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, pp. 541–551, Dec. 1989.

[97] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov. 1998.

[98] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[99] I. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks," *arXiv:1701.00160 [cs]*, Apr. 2017.

[100] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *arXiv:1511.06434 [cs]*, Jan. 2016.

[101] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, pp. 448–456, PMLR, 2015.

[102] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, pp. 214–223, PMLR, 2017.

[103] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved Training of Wasserstein GANs," in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), pp. 5767–5777, Curran Associates, Inc., 2017.

[104] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *arXiv:1411.1784 [cs, stat]*, Nov. 2014.

[105] 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.

[106] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, Oct. 2017.

[107] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8789–8797, IEEE, June 2018.

[108] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.

[109] B. Egger, W. A. P. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, C. Theobalt, V. Blanz, and T. Vetter, "3D Morphable Face Models—Past, Present, and Future," *ACM Transactions on Graphics*, vol. 39, pp. 1–38, June 2020.

[110] H. Li, T. Weise, and M. Pauly, "Example-based facial rigging," *ACM Transactions on Graphics*, vol. 29, pp. 32:1–32:6, July 2010.

[111] D. Vlasic, M. Brand, H. Pfister, and J. Popovic, "Face transfer with multilinear models," in *ACM SIGGRAPH 2006 Courses*, pp. 24–es, 2006.

[112] J. Booth and S. Zafeiriou, "Optimal UV spaces for facial morphable model construction," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 4672–4676, Oct. 2014.

[113] S. Moschoglou, E. Ververas, Y. Panagakis, M. A. Nicolaou, and S. Zafeiriou, "Multi-Attribute Robust Component Analysis for Facial UV Maps," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, pp. 1324–1337, Dec. 2018.

[114] S. Alotaibi and W. A. Smith, "A biophysical 3d morphable model of face appearance,"

in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 824–832, 2017.

[115] J. R. Tena, F. De la Torre, and I. Matthews, "Interactive region-based linear 3d face models," in *ACM SIGGRAPH 2011 papers*, pp. 1–10, 2011.

[116] C. Wu, D. Bradley, M. Gross, and T. Beeler, "An anatomically-constrained local deformation model for monocular face capture," *ACM Transactions on Graphics*, vol. 35, pp. 115:1–115:12, July 2016.

[117] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and y. Vetter, "A 3D Face Model for Pose and Illumination Invariant Face Recognition," in *Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 296–301, Sept. 2009.

[118] C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo, "A dictionary learning-based 3d morphable shape model," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2666–2679, 2017.

[119] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4D scans," *ACM Transactions on Graphics*, vol. 36, Nov. 2017.

[120] H. Dai, N. Pears, W. A. P. Smith, and C. Duncan, "A 3D Morphable Model of Craniofacial Shape and Texture Variation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3085–3093, 2017.

[121] 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*, pp. 1–1, 2020.

[122] 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, July 2017.

[123] X. Zhu, J. Yan, D. Yi, Z. Lei, and S. Z. Li, "Discriminative 3D morphable model

fitting," in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 1, pp. 1–8, May 2015.

[124] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 6, pp. 681–685, 2001.

[125] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A semi-automatic methodology for facial landmark annotation," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 896–903, 2013.

[126] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-shot multi-level face localisation in the wild," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5203–5212, 2020.

[127] 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, June 2007.

[128] Z. Fan, X. Hu, C. Chen, and S. Peng, "Dense semantic and topological correspondence of 3d faces without landmarks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 523–539, 2018.

[129] L. Tran and X. Liu, "On Learning 3D Face Morphable Model from In-the-wild Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.

[130] F. Cole, D. Belanger, D. Krishnan, A. Sarna, I. Mosseri, and W. T. Freeman, "Synthesizing Normalized Faces from Facial Identity Features," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3386–3395, IEEE, July 2017.

[131] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1274–1283, 2017.

[132] A. Tewari, M. Zollhöfer, P. Garrido, F. Bernard, H. Kim, P. Pérez, and C. Theobalt, "Self-Supervised Multi-Level Face Model Learning for Monocular Reconstruction at Over 250 Hz," in *Conference on Computer Vision and Pattern Recognition*, 2018.

[133] L. Tran and X. Liu, "Nonlinear 3D Face Morphable Model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7346–7355, 2018.

[134] A. Tewari, F. Bernard, P. Garrido, G. Bharaj, M. Elgharib, H.-P. Seidel, P. Perez, M. Zollhofer, and C. Theobalt, "FML: Face Model Learning From Videos," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, June 2019.

[135] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *Conference on Computer Vision and Pattern Recognition*, 2019.

[136] Y. Wang, L. Zhang, Z. Liu, G. Hua, Z. Wen, Z. Zhang, and D. Samaras, "Face Relighting from a Single Image under Arbitrary Unknown Lighting Conditions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1968–1984, Nov. 2009.

[137] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs, "SfSNet: Learning Shape, Reflectance and Illuminance of Faces 'in the Wild'," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6296–6305, 2018.

[138] A. Chen, Z. Chen, G. Zhang, K. Mitchell, and J. Yu, "Photo-realistic facial details synthesis from single image," in *IEEE/CVF International Conference on Computer Vision*, October 2019.

[139] S. Saito, L. Wei, L. Hu, K. Nagano, and H. Li, "Photorealistic facial texture inference using deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5144–5153, 2017.

[140] L. Huynh, W. Chen, S. Saito, J. Xing, K. Nagano, A. Jones, P. Debevec, and H. Li, "Mesoscopic Facial Geometry Inference Using Deep Neural Networks," in *2018*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8407–8416, IEEE, June 2018.

[141] S. Yamaguchi, S. Saito, K. Nagano, Y. Zhao, W. Chen, K. Olszewski, S. Morishima, and H. Li, "High-fidelity facial reflectance and geometry inference from an unconstrained image," *ACM Transactions on Graphics*, vol. 37, pp. 162:1–162:14, July 2018.

[142] M. Aittala, T. Aila, and J. Lehtinen, "Reflectance modeling by neural texture synthesis," *ACM Transactions on Graphics*, vol. 35, pp. 1–13, July 2016.

[143] Z. Li, K. Sunkavalli, and M. Chandraker, "Materials for Masses: SVBRDF Acquisition with a Single Mobile Phone Image," in *Proceedings of the European Conference on Computer Vision*, 2018.

[144] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, "Single-image SVBRDF capture with a rendering-aware deep network," *ACM Transactions on Graphics*, July 2018.

[145] X. Li, Y. Dong, P. Peers, and X. Tong, "Modeling surface appearance from a single photograph using self-augmented convolutional neural networks," *ACM Transactions on Graphics*, vol. 36, pp. 45:1–45:11, July 2017.

[146] C. Kampouris and A. Ghosh, "Icl multispectral light stage: building a versatile led sphere with off-the-shelf components," in *Proceedings of the Eurographics 2018 Workshop on Material Appearance Modeling*, pp. 1–4, 2018.

[147] C. Kampouris, S. Zafeiriou, and A. Ghosh, "Diffuse-specular separation using binary spherical gradient illumination.," in *EGSR (EI&I)*, pp. 1–10, 2018.

[148] "Scientific & technical awards," 2019.

[149] S. Mallick, T. Zickler, D. Kriegman, and P. Belhumeur, "Beyond Lambert: reconstructing specular surfaces using color," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, pp. 619–626 vol. 2, 2005.

[150] A. Ghosh, T. Chen, P. Peers, C. A. Wilson, and P. Debevec, "Estimating Specular Roughness and Anisotropy from Second Order Spherical Gradient Illumination," *Computer Graphics Forum*, vol. 28, no. 4, pp. 1161–1170, 2009.

[151] J. L. Schonberger and J. Frahm, "Structure-from-motion revisited," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 4104–4113, IEEE Computer Society, jun 2016.

[152] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[153] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[154] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3D Morphable Model Learnt From 10,000 Faces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5543–5552, 2016.

[155] J. Alabort-i Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou, "Menpo: A comprehensive platform for parametric image alignment and visual deformable models," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 679–682, 2014.

[156] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[157] 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*, 2007.

[158] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi, "Efficiently combining positions and normals for precise 3D geometry," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 536–543, 2005.

[159] P. Ekman and W. V. Friesen, "Facial action coding system," *Environmental Psychology & Nonverbal Behavior*, 1978.

[160] O. Klehm, F. Rousselle, M. Papas, D. Bradley, C. Hery, B. Bickel, W. Jarosz, and T. Beeler, "Recent advances in facial appearance capture," *Computer Graphics Forum (CGF)*, vol. 34, pp. 709–733, May 2015.

[161] T. Weyrich, J. Lawrence, H. Lensch, S. Rusinkiewicz, and T. Zickler, "Principles of appearance acquisition and representation," *Foundations and Trends in Computer Graphics and Vision*, vol. 4, no. 2, pp. 75–191, 2008.

[162] K. Guo, P. Lincoln, P. Davidson, J. Busch, X. Yu, M. Whalen, G. Harvey, S. Orts-Escolano, R. Pandey, J. Dourgarian, *et al.*, "The relightables: Volumetric performance capture of humans with realistic relighting," *ACM Transactions on Graphics (ToG)*, vol. 38, no. 6, pp. 1–19, 2019.

[163] S. Sengupta, B. Curless, I. Kemelmacher-Shlizerman, and S. M. Seitz, "A light stage on every desk," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2420–2429, 2021.

[164] J. Jiang, D. Liu, J. Gu, and S. Süsstrunk, "What is the space of spectral sensitivity functions for digital color cameras?," in *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pp. 168–179, IEEE, 2013.

[165] A. L. Yuille, D. Snow, R. Epstein, and P. N. Belhumeur, "Determining generative models of objects under varying illumination: Shape and albedo from multiple images using svd and integrability," *International Journal of Computer Vision*, vol. 35, no. 3, pp. 203–222, 1999.

[166] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-stage dense face localisation in the wild," *arXiv preprint arXiv:1905.00641*, 2019.

[167] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[168] 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 Conference on Computer Vision and Pattern Recognition*, pp. 1097–1106, 2019.

[169] M. Sela, E. Richardson, and R. Kimmel, "Unrestricted Facial Geometry Reconstruction Using Image-To-Image Translation," in *IEEE International Conference on Computer Vision*, 2017.

[170] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[171] B. Gecer, A. Lattas, S. Ploumpis, J. Deng, A. Papaioannou, S. Moschoglou, and S. Zafeiriou, "Synthesizing Coupled 3D Face Modalities by Trunk-Branch Generative Adversarial Networks," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), Lecture noteComments in Computer Science, Springer International Publishing, 2020.

[172] R. Li, K. Bladin, Y. Zhao, C. Chinara, O. Ingraham, P. Xiang, X. Ren, P. Prasad, B. Kishore, J. Xing, and H. Li, "Learning Formation of Physically-Based Face Attributes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[173] W. Zhu, H. Wu, Z. Chen, N. Vesdapunt, and B. Wang, "Reda:reinforced differentiable attribute for 3d face reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[174] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, Georgia Gkioxari, and Nikhila Ravi; Jeremy Reizenstein; David Novotny; Taylor Gordon; Wan-Yen Lo; Justin Johnson; Georgia Gkioxari, "PyTorch3D," 2020.

[175] S. Saito, L. Wei, L. Hu, K. Nagano, and H. Li, "Photorealistic Facial Texture Inference Using Deep Neural Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5144–5153, 2017.

[176] T. Weyrich, W. Matusik, H. Pfister, B. Bickel, C. Donner, C. Tu, J. McAndless, J. Lee, A. Ngan, H. W. Jensen, and M. Gross, "Analysis of human faces using a measurement-based skin reflectance model," *ACM Transactions on Graphics*, vol. 25, July 2006.

[177] P. Gotardo, J. Riviere, D. Bradley, A. Ghosh, and T. Beeler, "Practical dynamic facial appearance modeling and acquisition," *ACM Transactions on Graphics*, vol. 37, pp. 232:1–232:13, Dec. 2018.

[178] A. Lattas, M. Wang, S. Zafeiriou, and A. Ghosh, "Multi-view facial capture using binary spherical gradient illumination," in *ACM SIGGRAPH 2019 Posters*, ACM, July 2019.

[179] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," *Proceedings of the 6th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS) for Security, Safety and Monitoring in Smart Environments*, 2009.

[180] 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.

[181] Y. Guo, J. Zhang, J. Cai, B. Jiang, and J. Zheng, "CNN-Based Real-Time Dense Face Reconstruction with Inverse-Rendered Photo-Realistic Face Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 1294–1307, June 2019.

[182] 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.

[183] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4D scans," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, vol. 36, no. 6, pp. 194:1–194:17, 2017.

[184] S. Ploumpis, H. Wang, N. Pears, W. A. P. Smith, and S. Zafeiriou, "Combining 3D morphable models: A large scale face-and-head model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[185] H. Luo, K. Nagano, H.-W. Kung, Q. Xu, Z. Wang, L. Wei, L. Hu, and H. Li, "Normalized avatar synthesis using stylegan and perceptual refinement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11662–11672, 2021.

[186] L. P. Asselin, D. Laurendeau, and J. F. Lalonde, "Deep svbrdf estimation on real materials," in *2020 International Conference on 3D Vision (3DV)*, pp. 1157–1166, 2020.

[187] Y. Yu and W. A. P. Smith, "InverseRenderNet: Learning single image inverse rendering," in *Proc. CVPR*, 2019.

[188] A. Dib, G. Bharaj, J. Ahn, C. Thébault, P. Gosselin, M. Romeo, and L. Chevallier, "Practical face reconstruction via differentiable ray tracing," in *Computer Graphics Forum*, vol. 40, pp. 153–164, Wiley Online Library, 2021.

[189] M. B. R, A. Tewari, T.-H. Oh, T. Weyrich, B. Bickel, H.-P. Seidel, H. Pfister, W. Matusik, M. Elgharib, and C. Theobalt, "Monocular reconstruction of neural face reflectance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4791–4800, June 2021.

[190] S. Bi, S. Lombardi, S. Saito, T. Simon, S.-E. Wei, K. Mcphail, R. Ramamoorthi, Y. Sheikh, and J. Saragih, "Deep relightable appearance models for animatable faces," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–15, 2021.

[191] C. Kampouris and A. Ghosh, "ICL multispectral light stage: building a versatile LED sphere with off-the-shelf components," in *Proceedings of the Eurographics 2018 Workshop on Material Appearance Modeling*, EG MAM '18, Eurographics Association, July 2018.

[192] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A Semi-automatic Methodology for Facial Landmark Annotation," in *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 896–903, IEEE, June 2013.

[193] K. Nishino and S. K. Nayar, "Eyes for relighting," *ACM Transactions on Graphics*, vol. 23, pp. 704–711, Aug. 2004.

[194] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 286–301, 2018.

[195] G. Borshukov and J. P. Lewis, "Realistic human face rendering for "the matrix reloaded"," in *ACM SIGGRAPH 2005 Courses*, Association for Computing Machinery, July 2005.

[196] J. Hable, "Uncharted 2:, character lighting and shading," in *SIGGRAPH Advances in Real-Time Rendering in Games course*, July 2010.

[197] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," in *4th International Conference on Learning Representations, ICLR*, 2016.

[198] O. Alexander, M. Rogers, W. Lambeth, J.-Y. Chiang, W.-C. Ma, C.-C. Wang, and P. Debevec, "The Digital Emily Project: Achieving a Photorealistic Digital Actor," *IEEE Computer Graphics and Applications*, vol. 30, pp. 20–31, July 2010.

[199] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *2010 20th international conference on pattern recognition*, IEEE, 2010.

[200] H. Yang, H. Zhu, Y. Wang, M. Huang, Q. Shen, R. Yang, and X. Cao, "Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction," in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 601–610, 2020.

[201] S. Berretti, A. D. Bimbo, and P. Pala, "Superfaces: A super-resolution model for 3d faces," in *ECCV Workshops (1)*, 2012.

[202] 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.

[203] 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.

[204] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4d scans.," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 194–1, 2017.

[205] S. Ploumpis, H. Wang, N. Pears, W. A. Smith, and S. Zafeiriou, "Combining 3d morphable models: A large scale face-and-head model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10934–10943, 2019.

[206] S. Ploumpis, E. Ververas, E. O'Sullivan, S. Moschoglou, H. Wang, N. Pears, W. A. Smith, B. Gecer, and S. Zafeiriou, "Towards a complete 3d morphable model of the human head," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4142–4160, 2020.

[207] S. Ploumpis, S. Moschoglou, V. Triantafyllou, and S. Zafeiriou, "3d human tongue reconstruction from single" in-the-wild" images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2771–2780, 2022.

[208] S. Moschoglou, S. Ploumpis, M. A. Nicolaou, A. Papaioannou, and S. Zafeiriou, "3dfacegan: adversarial nets for 3d face representation, generation, and translation," *International Journal of Computer Vision*, vol. 128, no. 10, pp. 2534–2551, 2020.

[209] L. Tran and X. Liu, "Nonlinear 3d face morphable model," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7346–7355, 2018.

[210] L. Tran, F. Liu, and X. Liu, "Towards high-fidelity nonlinear 3d face morphable model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1126–1135, 2019.

[211] 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.

[212] 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/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7763–7772, 2019.

[213] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.

[214] 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.

[215] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 852–863, 2021.

[216] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, "Fast-ganfit: Generative adversarial network for high fidelity 3d face reconstruction," *arXiv preprint arXiv:2105.07474*, 2021.

[217] B. Gecer, J. Deng, and S. Zafeiriou, "Ostec: One-shot texture completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7628–7638, 2021.

[218] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3d reasoning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7708–7717, 2019.

[219] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, "Unsupervised training for 3d morphable model regression," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[220] C. Fuji Tsang, M. Shugrina, J. F. Lafleche, T. Takikawa, J. Wang, C. Loop, W. Chen, K. M. Jatavallabhula, E. Smith, A. Rozantsev, O. Perel, T. Shen, J. Gao, S. Fidler, G. State, J. Gorski, T. Xiang, J. Li, M. Li, and R. Lebaredian, "Kaolin: A pytorch library for accelerating 3d deep learning research." `https://github.com/NVIDIAGameWorks/kaolin`, 2022.

[221] J. Riviere, P. Gotardo, D. Bradley, A. Ghosh, and T. Beeler, "Single-shot high-quality facial geometry and skin appearance capture," in *Association for Computing Machinery (ACM)*, 2020.

[222] L. Bao, X. Lin, Y. Chen, H. Zhang, S. Wang, X. Zhe, D. Kang, H. Huang, X. Jiang, J. Wang, *et al.*, "High-fidelity 3d digital human head creation from rgb-d selfies," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 1, pp. 1–21, 2021.

[223] W. A. Smith, A. Seck, H. Dee, B. Tiddeman, J. B. Tenenbaum, and B. Egger, "A morphable face albedo model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5011–5020, 2020.

[224] A. Dib, C. Thébault, J. Ahn, P.-H. Gosselin, C. Theobalt, and L. Chevallier, "Towards high fidelity monocular face reconstruction with rich reflectance using self-supervised learning and ray tracing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12819–12829, 2021.

[225] H. Feng, T. Bolkart, J. Tesch, M. J. Black, and V. Abrevaya, "Towards racially unbiased skin tone estimation via scene disambiguation," in *European Conference on Computer Vision*, pp. 72–90, Springer, 2022.

[226] R. Li, K. Bladin, Y. Zhao, C. Chinara, O. Ingraham, P. Xiang, X. Ren, P. Prasad, B. Kishore, J. Xing, *et al.*, "Learning formation of physically-based face attributes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3410–3419, 2020.

[227] C. Cao, T. Simon, J. K. Kim, G. Schwartz, M. Zollhoefer, S.-S. Saito, S. Lombardi, S.-E. Wei, D. Belko, S.-I. Yu, *et al.*, "Authentic volumetric avatars from a phone scan," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 4, pp. 1–19, 2022.

[228] D. Wang, P. Chandran, G. Zoss, D. Bradley, and P. Gotardo, "Morf: Morphable radiance fields for multiview neural head modeling," in *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–9, 2022.

[229] A. Papaioannou, B. Gecer, S. Cheng, G. Chrysos, J. Deng, E. Fotiadou, C. Kampouris, D. Kollias, S. Moschoglou, K. Songsri-In, *et al.*, "Mimicme: A large scale diverse 4d database for facial expression analysis," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pp. 467–484, Springer, 2022.

[230] A. Chen, Z. Chen, G. Zhang, K. Mitchell, and J. Yu, "Photo-realistic facial details synthesis from single image," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9429–9439, 2019.

[231] T. Bagautdinov, C. Wu, J. Saragih, P. Fua, and Y. Sheikh, "Modeling facial geometry using compositional vaes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3877–3886, 2018.

[232] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 145–156, 2000.

[233] 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.

[234] 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.

[235] L. Huynh, W. Chen, S. Saito, J. Xing, K. Nagano, A. Jones, P. Debevec, and H. Li, "Mesoscopic facial geometry inference using deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8407–8416, 2018.

[236] S. Yamaguchi, S. Saito, K. Nagano, Y. Zhao, W. Chen, K. Olszewski, S. Morishima, and H. Li, "High-fidelity facial reflectance and geometry inference from an unconstrained image," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[237] R. C. Gonzalez, *Digital image processing*. Pearson education india, 2009.

[238] C. Aliaga, C. Hery, and M. Xia, "Estimation of spectral biophysical skin properties from captured rgb albedo," *arXiv e-prints*, pp. arXiv–2201, 2022.

[239] S. Alotaibi and W. A. Smith, "A biophysical 3d morphable model of face appearance," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 824–832, 2017.

[240] "Monk skin-tone (MST) scale." `https://skintone.google/`. Accessed: 2012-10-11.

[241] 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.

[242] D. Roich, R. Mokady, A. H. Bermano, and D. Cohen-Or, "Pivotal tuning for latent-based editing of real images," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 1, pp. 1–13, 2022.

[243] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

[244] O. Alexander, M. Rogers, W. Lambeth, J.-Y. Chiang, W.-C. Ma, C.-C. Wang, and P. Debevec, "The digital emily project: Achieving a photorealistic digital actor," *IEEE Computer Graphics and Applications*, vol. 30, no. 4, pp. 20–31, 2010.

[245] 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.

[246] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

[247] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, "Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 3, pp. 1–21, 2021.

[248] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[249] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[250] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Association*, 2015.

[251] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9841–9850, 2020.

[252] A. D. Bagdanov, A. Del Bimbo, and I. Masi, "The florence 2d/3d hybrid face dataset," in *Proceedings of the 2011 Joint ACM Workshop on Human Gesture and Behavior Understanding*, J-HGBU '11, (New York, NY, USA), p. 79–80, ACM, 2011.

[253] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[254] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5865–5874, 2021.

[255] P. Rao, B. Mallikarjun, G. Fox, T. Weyrich, B. Bickel, H. Pfister, W. Matusik, A. Tewari, C. Theobalt, and M. Elgharib, "Vorf: Volumetric relightable faces," 2022.

[256] Y. Hong, B. Peng, H. Xiao, L. Liu, and J. Zhang, "Headnerf: A real-time nerf-based parametric head model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20374–20384, 2022.

[257] S. Athar, Z. Shu, and D. Samaras, "Flame-in-nerf: Neural control of radiance fields for free view face animation," in *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 1–8, IEEE, 2023.

[258] A. Chatziagapi, S. Athar, A. Jain, R. M. V. Kumar, V. Bhat, and D. Samaras, "Lipnerf: What is the right feature space to lip-sync a nerf," 2023.

[259] A. Chatziagapi, S. Athar, F. Moreno-Noguer, and D. Samaras, "Sider: single-image neural optimization for facial geometric detail recovery," in *2021 International Conference on 3D Vision (3DV)*, pp. 815–824, IEEE, 2021.

[260] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022.

[261] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5501–5510, 2022.

[262] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[263] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

[264] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[265] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2020.