Reconstruction de formes 3D à partir de plusieurs vues

Pierre Zins

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique
Spécialité : Informatique
Unité de recherche : Laboratoire Jean Kuntzmann

Reconstruction de formes 3D à partir de plusieurs vues

3D shape reconstruction from multiple views

Présentée par :

Pierre ZINS

Direction de thèse :

Stefanie WUHRER CR, INRIA	Directrice de thèse
Edmond BOYER INRIA	Co-directeur de thèse

Rapporteurs :

Sylvie CHAMBON
MAITRE DE CONFERENCES, Toulouse INP
Mohamed DAOUDI
PROFESSEUR, IMT Nord Europe

Thèse soutenue publiquement le **25 avril 2023**, devant le jury composé de :

Sylvie CHAMBON MAITRE DE CONFERENCES, Toulouse INP	Rapporteure	
Mohamed DAOUDI PROFESSEUR, IMT Nord Europe	Rapporteur	
Jean-Sébastien FRANCO MAITRE DE CONFERENCES, Grenoble INP	Examinateur	
Florence BERTAILS-DESCOUBES DIRECTEUR DE RECHERCHE, INRIA	Examinatrice	Présidente

Invités :

Stefanie Wuhrer
CHARGE DE RECHERCHE, Inria de l'UGA
Edmond Boyer
DIRECTEUR DE RECHERCHE, Meta

Abstract

Recent technological progress in terms of hardware and software have given rise to a growing need for 3D content that can be used in several domains. In this thesis, we focus on geometric 3D content creation from multi-view 2D image data. Although specialized hardware such as depth sensors can help to capture 3D data, the prevailing strategy is to use only RGB images as input. Accurate 3D models captured from real data are useful in a wide variety of domains such as the entertainment industry to incorporate them in movies or video games, cultural heritage to preserve fragile objects or scenes, healthcare for diagnostics or virtual surgical simulations and virtual and augmented reality to provide immersive and realistic experiences for novel applications such as telepresence or virtual try-on. As seen with all these different applications, the 3D reconstruction task can take place in different contexts with variable size of the reconstructed content and different numbers of input images. In this thesis, we explore and contribute to two distinct scenarios.

First, we consider the reconstruction of dressed humans from a limited number of input views. This scenario is particularly interesting as digital humans are at the center of a large majority of visual content that we have access to today and the limited number of input views increases the applicability of the method with a simplified capture configuration. However, in such context, the problem becomes very challenging and ill-posed because redundant photometric information within the input images is insufficient to infer a complete 3D model. In this context, we improve over the state-of-the-art with a new data-driven method built on top of a neural implicit representation that proposes accurate and spatially consistent 3D reconstructions of dressed humans from only a few sparse input views. We demonstrate in our experiments a higher reconstruction accuracy than existing methods, and even a good generalization capability to real data while training on synthetic data only. Despite these impressive results, reconstructing complete and accurate models from only a limited number of views remains very challenging and methods that employ more input views are still very relevant.

We therefore consider such context in a second contribution which involves dense input viewpoints to reconstruct the visible surface. In this case, photometric redundancy is leveraged to estimate the surface position and the main challenges concern the 3D representation which must capture fine 3D details and the appearance matching in different views that can be difficult due to non-Lambertian surfaces, noise from the cameras or visibility issues. In particular, we contribute with a novel efficient strategy that combines the benefits of Multi-View Stereopsis (MVS) methods that can yield pixel-wise geometric accuracy with local depth predictions along viewing rays and the volumetric integration used in recent differentiable rendering-based reconstruction methods. In our experiments we demonstrate more accurate surface estimations and a good generalization ability of the method.

Finally, in a third contribution we leverage the first two contributions and investigate how to incorporate multi-view constraints in the data-driven reconstruction method that we developed. In particular, this is possible when the input images share some redundancy and improves the generalization ability

of the method, increases the level of details that can be captured and offers the possibility to use higher resolution images as input.

Résumé

Les récents progrès technologiques d'un point de vue logiciel et matériel ont donné naissance à un besoin croissant de contenu 3D pouvant être utilisé dans plusieurs domaines. Dans cette thèse, nous nous concentrons sur la création de contenu 3D géométrique à partir de données d'images 2D multi-vues. Bien que du matériel spécialisé, tel que des capteurs de profondeur, puisse aider à capturer des données 3D, la stratégie dominante consiste à utiliser uniquement des images RGB en entrée. Des modèles 3D précis capturés à partir de données réelles sont utiles dans une grande variété de domaines tels que l'industrie du divertissement pour les films ou jeux vidéo, le patrimoine culturel pour la préservation d'éléments fragiles, la santé pour le diagnostique ou les simulations chirurgicales et la réalité virtuelle et augmentée pour offrir des expériences immersives et réalistes. Ainsi, la tâche de reconstruction peut prendre place dans différents contextes en fonction de la taille du contenu 3D aini que du nombre d'images considérées en entrée. Dans cette thèse, nous explorons et contribuons sur deux scénarios distincts.

Tout d'abord, nous explorons la reconstruction 3D complète d'humains et de leur vêtements à partir d'un nombre limité de vues. Ce scénario est particulièrement intéressant puisque l'humain est au centre d'une grande majorité d'applications et qu'un nombre limité de vues facilite la mise en place d'une méthode avec une configuration de capture simplifiée. Cependant, dans un tel contexte le problème devient difficile et mal posé car les informations photométriques redondantes parmi les images d'entrée ne peuvent pas être exploitées seules pour déduire un modèle 3D complet. Dans ce contexte, nous améliorons l'état de l'art avec une nouvelle méthode basée sur un apprentissage et construite sur une représentation neuronale implicite qui propose des reconstructions 3D précises et spatialement cohérentes d'humains à partir de seulement quelques vues éparses en entrée. Nous démontrons dans nos expériences une précision de reconstruction supérieure à celle des méthodes existantes, et même une bonne capacité de généralisation aux données réelles. Malgré ces résultats impressionnants, la reconstruction de modèles complets et précis à partir d'un nombre limité de vues reste très difficile et les méthodes qui utilisent plus de vues d'entrée sont toujours très pertinentes.

Nous considérons donc dans une seconde contribution un tel contexte comportant des points de vue d'entrée denses. Dans ce cas, la redondance photométrique est exploitée pour estimer la position de la surface et les principaux défis concernent la représentation 3D qui doit permettre de capturer des détails 3D fins et la correspondance d'apparence dans différentes vues qui peut être difficile en raison de surfaces non-Lambertiennes, du bruit des caméras ou de problèmes de visibilité. En particulier, nous apportons une nouvelle stratégie efficace qui combine les avantages des méthodes de stéréopsie multi-vues (MVS) qui peuvent donner une précision géométrique au niveau du pixel avec des prédictions de profondeur locales le long des lignes de vue et l'intégration volumétrique utilisée dans les récentes méthodes de reconstruction basées sur le rendu différentiable. Dans nos expériences, nous démontrons des estimations de surface plus précises et une bonne capacité de généralisation de la méthode.

Enfin, dans une troisième contribution, nous tirons profit des deux premières contributions et étudions comment incorporer des contraintes multi-vues dans la méthode de reconstruction basée sur un apprentissage que nous avons développée. En particulier, cela est possible lorsque les images d'entrée partagent une certaine redondance et permet d'améliorer la capacité de généralisation de la méthode, le niveau de détails qui peut être capturé et offre la possibilité d'utiliser des images de plus haute résolution comme entrée.

Acknowledgements

My research work was supported by Meta Reality Labs, Sausalito, USA and the 3D data acquisition was supported by a French government funding managed by the National Research Agency under the Investments for the Future program (PIA) with the grant ANR-21-ESRE-0030 (CONTINUUM project).

First, I would like to thank my supervisors, Edmond and Stefanie, for giving me the opportunity to do my PhD at Inria, for their unwavering support, their teachings, and their continuous guidance that made it possible for me to succeed in this thesis. They were able to find a very satisfactory work rhythm between freedom and necessity of results, and allowed me to discover the world of research in ideal conditions. Also, I would like to thank Tony Tung and Yuanlu Xu from Meta Reality Labs, Sausalito (USA) for their involvement, interesting discussions, and valuable advice all along my PhD journey.

Besides my advisors, I would also like to thank the rest of my thesis committee: Florence Bertails-Descoubes, Senior Researcher at Inria, Sylvie Chambon, Associate Professor at Toulouse INP, Mohamed Daoudi, Professor at IMT Nord Europe and Jean-Sébastien Franco, Associate Professor at Grenoble INP, for their helpful and insightful comments. I would also like to thank Professor Gerard Pons-Moll for being my CSI expert and giving his external opinion on my work at several points of my PhD.

I would also like to thank all the people I had the chance to work with during these almost four years. In particular, I would like to express my gratitude to the other permanent members of the Morpheo team: Jean-Sébastien and Sergi for the interesting discussions during which I received valuable advice, Nathalie for her assistance with all the administrative tasks and Laurence and Julien from the Kinovis platform and our volunteer subjects, Sonia and Edmond, for their help with the 3D data acquisition.

My thanks also go to all the other members of the Morpheo team: Matthieu and Jean for their help and advice at the beginning of my thesis, Julien, Vincent, Matthieu, Abdel and Tomas for the very pleasant runs to the "reservoirs" during lunch breaks and the trail trips in Belledonne or the Chambéry-Grenoble in the Chartreuse. A special thank to Mathieu my office neighbor since the beginning of our PhD that made this journey much more pleasant, Boyao, Anil and Abdel for the interesting discussions during breaks and all the other PhD students, postdocs, engineers, interns and visitors that I had the chance to meet: Stephane, Di, Nitika, Victoria, Sanae, Haroon, Roman, Nicolas, Joao, Diego, Rim, Aymen, Mattia, Briac, Maxime, Kristijan, David, Shivam, Anne-Flore, Samara, Hector and Itzel.

I would also like to thank my family, especially my parents, for their unconditional support and love over the years. I would like to thank my brother, Matthieu, for his help and motivation and for proofreading important parts of this document, as well as my aunt Nicole and my uncle Jeannot for their support and encouragement.

Finally, I dedicate the rest of this thesis to Gaëlle, whose presence, encouragement and motivation helped me face the stressful events that occur during

the life of a PhD student. You made me discover the mountains around Grenoble and new activities like trekking and Nordic skiing and my PhD would have been so much harder without you.

Contents

List of Figures

List of Tables

1 Introduction

1.1 General Context

3D reconstruction refers to the automatic process of creating a digital model of a physical object or environment from multiple 2D images or measurements. This task represents a very long-standing problem in Computer Vision that has been explored for more than 40 years. Digitizing an object or a scene includes the 3D geometry, the appearance and possibly the motion if a fourth temporal dimension is considered. Accurate 3D reconstructions are important in many domains and allow a wide range of new applications.

First, it is widely used for entertainment purposes, particularly in the fields of film, television and video games. By creating an accurate digital model of a real-world object or location, it is possible to use them in a film or television production without the need to physically build or recreate them. The same applies for video games in which real-world objects, humans or locations are often incorporated into immersive and realistic worlds.

With all the recent progress in terms of hardware, new types of devices appeared in particular Head Mounted Devices (HMD) for Virtual Reality such as the Occulus Rift, Meta Quest and the HTC Vive or glasses for Augmented Reality such as Hololens and Google glass (see Figure 1.1). New development kits were also released such as ARKit and ARCore which target the developement of virtual and agumented reality applications on mobile devices. These new devices and software enable a much richer experience in 3D than videos or images that are only a 2D projection of the real world. As a result, to provide realistic and immersive experiments, the need for accurate and complete 3D models is vital.

Other domains also benefit from 3D reconstruction such as cultural heritage preservation by creating a detailed and accurate record of an object or site, including its dimensions, shape and surface features. This can be useful for preserving the physical characteristics of the object or site for future generations and to have access to a virtual version that can be studied and analyzed

a) b) c)

Figure 1.1: New virtual and augmented reality devices. a) Meta Quest 2 [1] *b) HTC Vive Pro* [2] *c) Microsoft Hololens 2* [3]

from any angle and at any scale. This is especially useful for objects or sites that are difficult to access or view in person or that are too fragile to visit or manipulate.

When applied to large scenes, 3D reconstruction is also very important and is used by a huge range of industries such as agriculture or inspection. By obtaining a 3D aerial map of the crops, farmers can optimize crop yields and reduce the use of pesticides and chemicals. Accurate 3D models can also help for inspection of sites that can be too dangerous or time-consuming to be performed in real-life, such as electricity pylons, telecommunication towers or even nuclear power plants. At an even larger scale, 3D reconstruction is also used by Google Earth to create a 3D model of the full 3D virtual globe which can be used for exploration, education or planning for example.

Finally, 3D reconstruction is very useful in the medical domain for diagnosis and treatment planning. Reconstruction techniques can be used to create detailed digital models of a patient's anatomy, which can then be used for diagnostic purposes by identifying abnormalities or to plan surgical procedures. Accurate 3D models of a patient's anatomy, can also be used to perform virtual surgical simulations. This can be useful for training surgeons or for preparing complex surgeries, as it allows surgeons to practice and test different approaches in a virtual environment before performing the surgery in real-life.

As shown with all these examples, 3D reconstruction is widespread and its range goes from small objects to larger scenes or even worldwide reconstructions.

All the methods that digitize the 3D world share a common first acquisition step based on specific technologies. Existing capture technologies can be can be categorized in two main groups. First, active systems use an energy source, such as lasers or lights, to actively scan the object or scene. These systems have the advantage of being able to capture high-resolution 3D data accurately, but they may be more expensive and require more specialized equipment. On the other hand, the passive systems (see Figure 1.2) do not involve the use of an active energy source and only rely on existing light or other environ-

[1] https://fr.wikipedia.org/wiki/Meta_Quest_2
[2] https://en.wikipedia.org/wiki/HTC_Vive
[3] https://fr.wikipedia.org/wiki/Fichier:HoloLens_2.jpeg

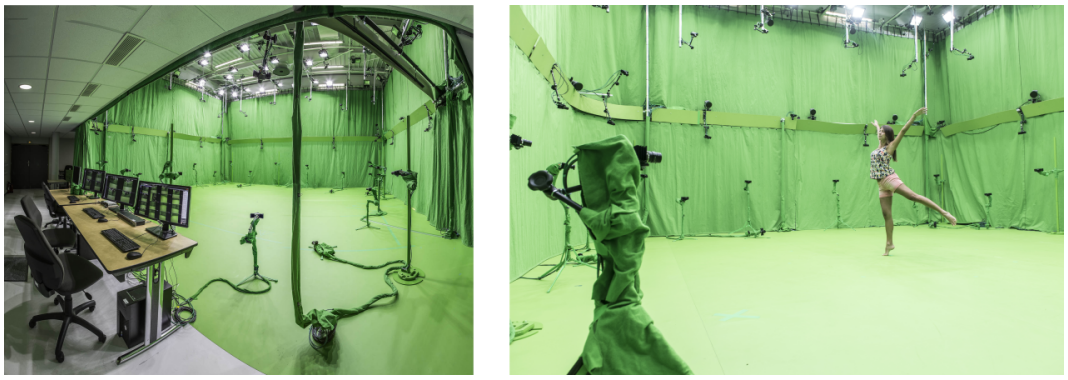


Figure 1.2: The Kinovis passive capture platform at Inria Grenoble [4].

mental factors to capture 3D data. With a simplified setup, passive capture systems composed of several RGB cameras become the prevailing choice for 3D reconstruction and are also the acquisition system considered in this thesis.

By using such passive sensors, the problem of reconstructing the 3D geometry of an object or a scene can be approached in many different reconstruction contexts. It can go from reconstructing small objects to larger scenes and from static capture scenarios to dynamic or even real-time ones. More recently, a large amount of research also tried to simplify even more the passive systems for multi-view reconstruction by considering fewer input views. In this thesis, we consider two different scenarios. First, we explore 3D reconstruction from only a few views specifically for humans and in a second time we develop a more general method for 3D reconstruction that can be applied to any type of objects by using more input views.

1.2 3D Reconstruction of Humans

Humans represent probably the central component in a large majority of visual content that we have access today through pictures, movies, TV shows, video games or even sports. More recently with the progress of virtual and augmented reality, accurate 3D digitizations of humans become even more important to obtain an immersive and realistic experience. As shown in Figure 1.3, Humans are the central key of several applications such as virtual try-on, telepresence or in the virtual world called *Metaverse* that is expected to grow in the future. Although less expensive and complex than active capture platforms, setups composed of calibrated multi-camera still limit the applicability of the reconstruction methods to controlled laboratory setups. To alleviate such limitations and increase the applicability of the 3D reconstruction methods from images, a large amount of research investigated reconstruction from much less input views. In that case, the problem becomes very challenging and ill-posed, especially in the monocular case, and the methods cannot use photo-metric redundant information among the input images to infer 3D. As

[4] https://kinovis.inria.fr/inria-platform/

a) b)

Figure 1.3: Virtual humans applications. a) Telepresence [5] *b) Virtual try-on* [6]

a solution, all the methods rely on prior information that can take several forms such as the depth information provided by an active RGB-D camera or statistical human body parametric models such as SCAPE [6] or SMPL [116]. These low-dimensional parametric models were used a lot in the context of 3D human reconstruction from one or a few images. The earliest methods such as SMPLify [13] proposed optimization techniques to infer statistical human model parameters from the 2D detections of the human joints in one or a few images. More recently, with the progress of deep learning, several methods proposed end-to-end trainable architectures [85, 90, 149] that directly estimate the human model parameters from one or a few images. Low-dimensional statistical human models offer a robust prior information for the human body but only represent a naked human which prevents the reconstruction of clothing and accessories. To overcome these limitations, alternative representations were explored and combined with deep learning such as meshes [4, 207], voxels [58, 189] and depthmaps [52, 175]. These representations bring more flexibility to represent clothing but still include several limitations. Meshes are limited to a single topology, voxels cannot be considered for high resolution 3D data as the spatial complexity grows cubically and depthmaps are still a 2.5D representation and additional steps are required in order to obtain a complete 3D reconstruction of a human. To go even further, implicit continuous neural representations were explored by various methods [162, 163] and offer several advantages such as theoretical infinite resolution, no topology limitation, memory efficiency and easy integration with deep learning frameworks. More specifically, it consists in implementing an implicit function as a Multi Layer Perceptron (MLP) that takes as input a 3D coordinate and outputs either the occupancy of that point (inside/outside) or the signed shortest distance from the point to the surface. In particular, digitizing a human from a

[5] https://www.flickr.com/photos/wolfvision_vsolution/14472241106

[6] https://commons.wikimedia.org/wiki/File:Virtual_clothes_trying_%289507550174%29.jpg

single or only a few images brings several challenges:

- The reconstruction should be of high quality with accurate and complete geometry. Otherwise, the *Uncanny Valley* [130] phenomenon is possible when the human reconstructions can be perceived as unsettling or disturbing as they approach a level of realism that is close to, but not quite, human. This occurs in particular because the human brain is very sensitive to subtle deviations from what is expected to be a human.
- A full reconstruction should be possible from only a limited number of input views. Photo-metric redundancy alone cannot be used in that context and some parts are not even visible due to self-occlusion or the low number of input views. For that, good prior information should be leveraged efficiently by the method.
- The method should also be robust to a wide variety of clothing types from tight clothes such as a t-shirt or tights to looser clothing such as coats or dresses. This also involves being able to reconstruct shapes with varying topologies for example between someone wearing pants or a dress.
- As a human is always interacting with its surrounding environment, objects also play a important role. In many situations, a human is holding an object and it is therefore a strong advantage if the method can reconstruct a dressed human in 3D and also potential accessories.

1.3 3D Reconstruction From Many Views

As mentioned previously, reconstructing dressed humans from only a few sparse views requires prior information and today the most efficient methods use deep learning techniques to directly learn this prior information from data. Thanks to these techniques, a significant improvement was possible in the past few years and results are now impressive. However, the reconstruction problem from only a limited number of views remains very challenging and generalization to never-seen-before data is difficult. For a human, in particular, complex poses or clothing that are unusual and not sufficiently present in the training dataset cause reconstruction problems. Moreover, by leveraging prior information specific to human, the method becomes specialized for this type of reconstruction and cannot deal well with other types of data.

Based on these limitations, the reconstruction techniques that use many input views are still very relevant. These latter have inherently more information to perform the reconstruction and, as a result, they have the potential to achieve higher reconstruction accuracy, much better generalization and are not restricted to a single type of data. This reconstruction scenario of objects or humans from multiple input views is the second scenario considered in this thesis. For this problem, a large number of methods were already proposed in the past and we can distinguish roughly three main categories. The first type represents the more traditional methods that proposed optimization approaches. The earliest methods suggest to carve volumetric grids [98] or

directly deform meshes [49] to obtain the correct 3D geometry. Later, the tendency switched to the depthmap-based approaches as they alleviate the memory limitation issues of volumetric grids and allow for higher resolution reconstructions. With the recent progress of deep-learning techniques, a whole line of reserach tried to benefit from them for the problem of 3D reconstruction. Several methods proposed to learn some parts of the reconstruction pipeline such as the image feature matching [66, 104, 224] or the standard depthmaps fusion step [39, 157], while others even proposed to learn the full pipeline in an end-to-end manner [83, 215]. These methods constitute the second category and more recently a third one appeared which uses differentiable rendering to seek for observation fidelity. These methods approximate the rendering process with a differentiable function that allows to modify an internal 3D representation for the geometry and appearance until it matches with the observed images. This strategy was first explored with various shape representations including meshes [70, 92, 112, 146], volumetric grids [53, 86, 139, 188, 235] or even point clouds [80, 84, 135] and more recently combined with neural implicit representations [129, 197, 217] to offer impressive 3D reconstructions and novel view renderings. As all the methods rely, explicitly or implicitly, on accurate point matching across different images, several challenges can arise:

- Non-lambertian surfaces: when the Lambertian assumption is not respected, the light is not reflected equally in all directions which can change the appearance of a same 3D point in different images. This makes the appearance matching more challenging. The appearance can vary a lot, especially with specular surfaces that create strong highlights or reflections.

- The appearance matching is also dependent on the visibility of a point in an image. However, the true visibility can only be obtained from the true geometry which creates a cyclic dependency.

- Noise and image blur coming from the cameras can also affect the matching between several images which can in turn decrease the accuracy of the reconstruction.

1.4 Outline and Contributions

In Chapter 2, we review some background and related works about the topics discussed in this thesis. In particular, we present the different acquisition technologies that exist to capture the 3D world and introduce the different representations used for 3D data. Then, we elaborate on the problem of 3D reconstruction of humans with a specific focus on the methods that use a limited number of views as input. In the second part of the chapter we discuss the problem of 3D reconstruction in a more standard context, by considering many input RGB images. In particular, we introduce the concept of photo-consistency which is a major component of all the methods and then we present existing works for this type of problem. We group the approaches in three main categories, the traditional methods, the methods that incorporate deep

learning into the MVS pipeline and the more recent methods that are based on differentiable rendering.

In Chapter 3, we present a novel method for 3D reconstruction of dressed humans from a few sparse views. For that, we leverage the recently introduced neural implicit representations used for monocular reconstruction and propose to lift the single-view input with additional views. This allows in particular to alleviate the depth ambiguities and strong occlusions inherent to single view inputs. We explore a strategy to efficiently combine the contributions of the different input views and present a technique to obtain spatially consistent reconstructions, which allows for arbitrary placement of the person in the input views.

In Chapter 4, we consider the more traditional scenario for 3D reconstruction with many views as input. Inspired, by the three categories of works which tackle that problem, we propose a novel optimization-based method which combines advantages from each of them. In particular, we present a volumetric signed ray distance representation that we parameterize with depths along viewing rays. This representation makes the shape surface explicit with depths, keeps the benefit of better distributed gradients with a volumetric discretization and retains pixel-accuracy by optimizing depthmaps.

In Chapter 5, we study again the 3D reconstruction of dressed humans from a few sparse views. In contrast to chapter 3, we assume that these views include some redundancy and we explore two strategies to combine multi-view constraints with prior-based reconstructions to obtain accurate and full digitization of dressed humans. In particular, we first present an end-to-end learnable pipeline that incorporates a deep learning-based architecture for MVS, and then, an optimization method based on recent progress in differentiable rendering.

Finally, in the conclusion (chapter 6), we discuss the limitations and possible future directions for the work in this thesis.

In particular, we can summarize the following contributions in this thesis:

- We propose a spatially consistent 3D reconstruction framework that allows for arbitrary placement of the human in the scene, achieved by learning the model in a canonical coordinate system and by accounting for the transformation of each input view to this system.
- We introduce a learnable attention-based fusion layer that efficiently weighs the view contributions. This layer implements a multi-head self-attention mechanism inspired by the Transformer network [190].
- We propose a local 3D context encoding layer that better generalizes over the local geometric configurations, which is implemented through randomized 3D local grids.
- We demonstrate how to train our end-to-end pipeline on a large synthetic dataset of dressed humans and show that it can even generalize to real data.
- We introduce a novel optimization framework for 3D reconstruction from multiple images that combines depth optimization, as performed in the

latest MVS strategies, with volumetric integration, as used in more recent methods based on differentiable rendering. In particular, this framework proposes pixel-wise accuracy by construction, does not require color decisions, offers strong geometric consistency over different depthmaps and significant parallelism.

- Our optimization framework is agnostic to the photo-consistency metric that is used and we demonstrate that such metric can be efficiently learned from data with a deep neural network and provides good generalization abilities.

- We present two strategies to incorporate multi-view constraints in the data-driven reconstruction method that considers only a few input views.

The work in the thesis has led to the following publications:

– Pierre Zins, Yuanlu Xu, Edmond Boyer, Stefanie Wuhrer, and Tony Tung. Data-driven 3d reconstruction of dressed humans from sparse views. In *2021 International Conference on 3D Vision (3DV)*, pages 494–504. IEEE, 2021

HAL page: https://hal.science/hal-03385107v3

– *(Accepted at CVPR 2023)* Pierre Zins, Yuanlu Xu, Edmond Boyer, Stefanie Wuhrer, and Tony Tung. Multi-view reconstruction using signed ray distance functions (srdf). *arXiv preprint arXiv:2209.00082*, 2022

HAL page: https://hal.science/hal-03766943v1

The associated code is available at: https://gitlab.inria.fr/pzins/.

2

Background and Related Works

In this chapter we review the background of the topics discussed in this thesis and introduce relevant related works. In Sections 2.1, and 2.2 we start by presenting the different acquisition technologies to capture the 3D world and the main representations used for 3D data. Then, in Section 2.3, we present different strategies that were proposed by the community for reconstructing humans from a few views. In Section 2.4, we present some background as well as related works for the problem of 3D reconstruction in a more general setting with many input images. In particular, in Section 2.4.2, we review the more traditional approaches for Multi-View Stereopsis, in Section 2.4.3, the more recent data-driven techniques and in Section 2.4.4, another line of works that explore differentiable rendering approaches.

2.1 Capture Technologies

The existing technologies to capture the 3D world can be categorized in two main groups: active and passive.

Active systems involve some kind of energy source (such as lasers or lights) to actively scan the object or scene. They provide more prior information for 3D and were used a lot to create high fidelity digitizations in the past years. Dou et al. [40] and Collet et al. [32] integrated for example infrared cameras in their pipelines to propose respectively real-time high quality reconstructions and streamable free-viewpoint videos. However, this type of systems also has several limitations. First, active systems usually require more expensive hardware and complex setups. Interference issues can also appear when combining many active sensors and the frame-rate is usually lower than RGB cameras. Finally, the scalability is also more limited as the size of an active capture platform is usually much smaller than the passive ones.

On the contrary, passive systems only require RGB cameras that gather the natural light coming from the environment to capture the appearance

of the 3D world. RGB cameras offer the advantage of being relatively low-cost compared to active sensors and do not introduce interference issues when many of them are combined. Moreover, with the recent progress in terms of hardware, RGB cameras can capture high resolution images (4K, 8K) at very high frame-rates, going from around thousands frames per second for a smartphone to millions or trillions for specialized cameras. For all these reasons, passive capture systems composed of several RGB cameras become the prevailing choice for 3D reconstruction and are also the acquisition system considered in this thesis.

2.1.1 Active Systems

Marker triangulation A first active system consists in equipping a subject with several infrared markers. When the scene is illuminated with infrared light, the position of these markers can be captured with infrared cameras. From the projection of these markers and the calibration parameters of the cameras, the 3D position of the markers can be deduced by triangulation. This system can reconstruct a sparse representation of a shape by capturing a set of interest keypoints but cannot provide realistic dense reconstructions. It was however widely used in the industry for human motion capture.

Laser point triangulation This method is also based on the triangulation principle. A lased beam is cast into the scene and the dot that represents the intersection with the surface is detected in several images. Again, by using the calibration, the 3D position of that point can be determined. By casting numerous beams, it is possible to recover a dense point cloud of the scene.

Structured light This method projects a pattern of light with known geometry (usually light stripes) onto the object or scene, and uses a camera to capture the deformation of the pattern on the object. The resulting distortion of the pattern in the image can be used to calculate a relative depth at every pixel. In particular, this technology is used in the first version of the Microsoft Kinect [229].

Time-of-Flight (ToF) cameras These cameras measure the time it takes for a pulse of light to travel from the camera to the object and back, and use this information to calculate the distance to each point on the surface. In the end, depthmaps are obtained by casting a ray at each pixel of an image. Time-of-Light technology was used in the second version of the Microsoft Kinect [165] and is part of a broader family of range sensors that can estimate the camera-object distance based on a round-trip time. For example the SONAR for Sonic ranging sensors is another example which sends out a pulse of sound and waits for the echo to return. The time it takes for the echo to return is used to determine the distance to the obstacle.

Photometric stereo By capturing a scene with different lighting conditions, normals of the observed surface can be obtained by using the shading

variations. The famous Shape-from-Shading [74] strategy corresponds to this type of techniques when a single input view is used.

Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) scanners CT-scanners and MRI scanners are two other examples of active systems that are used for 3D reconstruction especially in the medical domain. They respectively rely on a measure of the absorption of X-rays by the human body tissues and strong magnetic fields.

2.1.2 Passive Systems

Silhouettes Silhouette-based 3D reconstruction involves using the silhouette or outline of an object as input data to estimate its 3D shape. Usually, multiple input views are considered, and the silhouettes can be obtained by 2D segmentation methods. Given the camera parameters, a 2D silhouette can be backprojected in the 3D space to create a conical volume. By considering several cameras, the intersection of all these volumes creates the final reconstruction, called Visual Hull [100]. This technique is relatively robust to noise and occlusions, as the silhouette of an object tends to be well-defined even when other parts of the object are not visible. This technique is also quite efficient and can even run in real-time. However, as it relies on the silhouette information, it cannot capture concavities which greatly limits the reconstruction accuracy and also heavily depends on the quality of the 2D silhouettes.

Shape from defocus Shape from defocus [45] is another passive technique for estimating the 3D shape of an object or scene from 2D images. It takes advantage of the amount of blur in an image which is related to the distance of the objects in the scene from the camera. By analyzing the blur in an image captured with different focus setting, it is possible to estimate the depth of each point on the object or scene. As this technique involves capturing several images of the same scene, it is not suitable for dynamic scenes.

Stereopsis or Multi-View Stereopsis (MVS) This method involves capturing a scene or an object under two slightly different viewpoints. 2D pixel correspondences between the two images can then be found and used in combination with the calibration parameters to compute the 3D position of the corresponding surface point by triangulation. This technique is similar to the human visual perception system with the two eyes acting as cameras. Stereopsis always considers two viewpoints but the extension to multiple ones is possible and known as Multi-View Stereopsis (MVS).

Structure-from-Motion Structure-from-Motion is a related technique that in contrast to MVS does not assume to have access to the calibration parameters. It usually involves a single camera capturing a static object or a scene under different viewpoints and also uses 2D pixel correspondences to simultaneously estimate the camera parameters and a sparse 3D reconstruction.

2.2 3D Data Representations

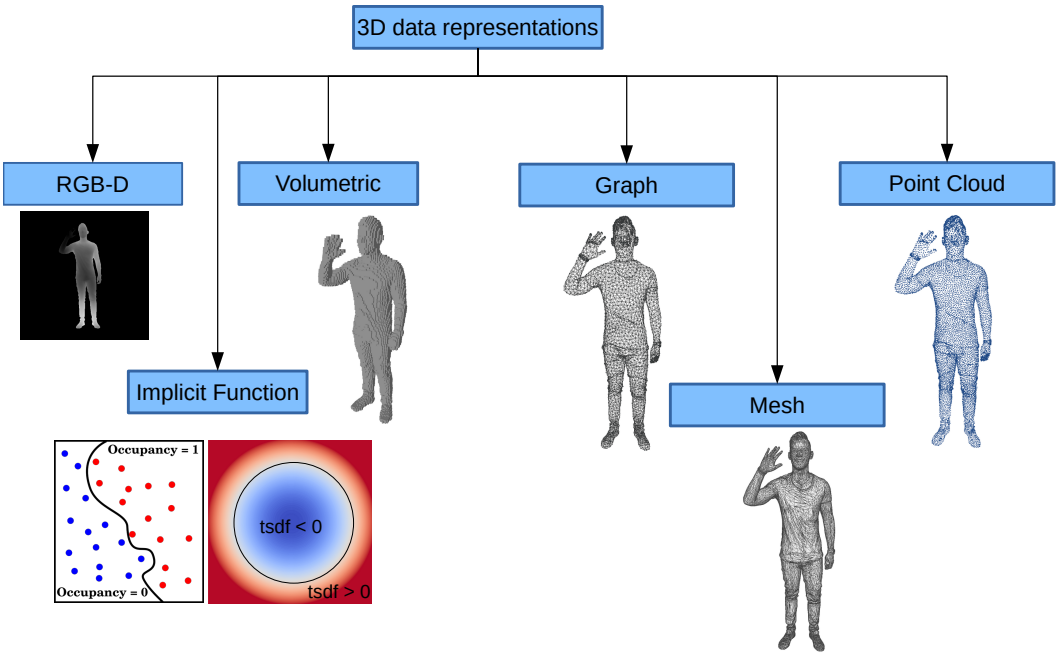


Figure 2.1: Existing representations for 3D data.

Recently 3D data become more and more common with the progress in the domains of Computer Vision, Computer Graphics and Robotics. Depending on the acquisition devices and the applications, many representations exist with different properties and structures, as shown in Figure 2.1. Each of them has advantages and limitations which make them more suitable for some tasks or scenarios. A particularly important point is the suitability of a representation with the recent deep learning techniques. In the following, we present each of these representations.

2.2.1 RGB-D Data

RGB-D data represents a surface as a 2D pixel grid where each pixel stores color and depth information. It can provide information about the 3D geometry from a fixed point of view but can also be captured from different viewpoints and combined to obtain a more complete 3D information.

This representation offers various advantages such as memory efficiency, easy processing and easy combination with deep learning techniques thanks to its regular grid structure. However, it is not a true 3D representation as it is sensitive to the viewpoints and only represents the visible surfaces.

It gains a lot in popularity with the release of powerful and low-cost depth sensors such as the Microsoft Kinect or Intel RealSense and, as a result, RGB-D datasets [19, 47] also become very common compared to other 3D datasets that propose meshes or point clouds. Many computer vision tasks are developed using that representation such as identity recognition [136], pose estima-

tion [43, 97, 194], object detection [42, 154], scene understanding [22, 176] and reconstruction [3, 240].

2.2.2 Volumetric Data

A regular 3D voxel grid is a data structure that divides the 3D space into a grid of small cubic units called voxels which are the direct extension to 3D of pixels in the 2D domain. Although, various information such as colors, densities or materials can be assigned to each voxel, two main definitions exist to represent a 3D geometry.

First, the occupancy representation assigns a binary value to each voxel V_i depending if it is inside or outside a closed shape Ω,

$$V_i = \begin{cases} 0 & \text{if } V_i \notin \Omega \\ 1 & \text{if } V_i \in \Omega \end{cases} \tag{2.1}$$

Instead of binary occupancy values, it is also possible to assign continuous values between 0 and 1 that represents the probability of occupancy of a voxel V_i.

The Signed Distance Function (SDF) is the second possibility in which each voxel V_i contains the signed distance between the center of each voxel and the nearest surface of the closed shape Ω,

$$V_i = \begin{cases} dist(V_i, \partial\Omega) & \text{if } V_i \in \Omega \\ -dist(V_i, \partial\Omega) & \text{if } V_i \notin \Omega \end{cases} \tag{2.2}$$

where $dist()$ computes the shortest distance from the center of V_i to the surface $\partial\Omega$ of the shape Ω. Several variations of the Signed Distance Function also exist with the Unsigned Distance Function (UDF) that defines only positive distances or Truncated Signed Distance Function (TSDF) that truncates the values at a specific threshold.

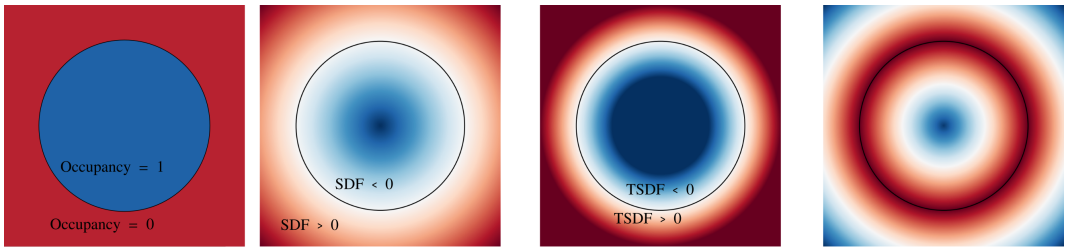
Thanks to its regular structure, voxel grids can be used directly in convolutional network architectures by performing convolutions on a 3D grid instead of a 2D image grid and as a result became very popular for several computer vision tasks such as shape classifications [17] or multi-view object reconstruction [30].

However, the spatial complexity of a voxel grid of size $(N \times N \times N)$ is $\mathcal{O}(n^3)$ which makes this representation expensive in memory and computation time. This is a major drawback of voxel grids and limits their suitability to represent high resolution data. As a solution, some more efficient 3D volumetric representations were also proposed such as Octrees [124, 183] or KD-trees [11, 200]. These hierarchical tree data structures subdivide the 3D space and allow to save a lot of space and control the resolution, i.e., high resolution around the surface and low resolution in empty space. Finally, another limitation of these volumetric representations is that the geometric surface properties are not preserved as the surface is not explicitly represented.

2.2.3 Implicit Surfaces

Implicit functions can also be used to represent 3D data and form a continuous extension of the volumetric representations described previously. They are defined with an equation $f(X) = c$ that associates a function value c to each 3D point $X = (x, y, z)$ in space. As for volumetric data and as shown in Figure 2.2, these functions can encode a probabilistic or binary occupancy [126] of a 3D point in space or a distance [202] which corresponds to the signed, unsigned or truncated distance from the point X to the nearest surface. Depending on the exact definition of the implicit function, the final surface of a shape is modeled as a level-set of the function, usually 0.5 for occupancy and 0 for a distance function. A final transformation step is necessary to recover an explicit surface by using an iso-surfacing algorithm such as Marching Cubes [117].

This type of representation allows to represent shapes with varying topologies and its continuous nature offers a theoretical infinite resolution [126] while keeping a low memory consumption. In addition, it is highly compatible with deep learning frameworks and several works implemented implicit functions with MLP to represent 3D data for different tasks such as classification [89], shape completion [29, 234] or shape reconstruction [126, 202].



a) Occupancy | b) Signed Distance Function (SDF) | c) Truncated Distance Function (TSDF) | d) Unsigned Signed Distance Function (USDF)

Figure 2.2: A 2D visualization of a circle (in black) and its associated implicit functions of occupancy a) or distances b) c) and d). It works similarly in 3D.

2.2.4 Point Cloud

A point cloud is a discrete set P of 3D data points X_i sampled on the surface of a 3D shape and specified by their coordinates in the three-dimensional space: $P = \{X_i = (x_i, y_i, z_i) | i = 1, ..., n\}$. This set is unordered and unstructured by construction.

This data representation is very popular for various tasks in Robotics, Computer Vision or Autonomous Driving thanks to its relative ease of capture with the available technology such as structured-light scanners. Another strong advantage is its compactness as only the surface of a 3D object is modeled.

On the other hand, their unstructured and unordered natures and lack of connectivity information make the processing of point cloud more difficult than regular structured data such as depthmaps or voxel grids. Extending the very efficient 2D deep learning operations to point cloud is also not straightforward, but a lot of works tried to address this issue. Convolutions for example

were extended to point clouds using different types of kernels: 3D continuous kernels [14, 113, 114] or 3D discrete kernels [75, 99, 101].

2.2.5 3D Graphs and Meshes

3D graphs and meshes form another popular representation for 3D data. A graph G can be defined as a pair $G = (V, E)$ where

- $V = \{v_i | i = 1, ..., n\}$ is a finite set of vertices, each represented with their 3D coordinates in the Euclidean space $v_i = (x_i, y_i, z_i)$
- $E = \{e_j | j = 1, ..., m\}$ is a finite set of edges that represent the connections between vertices.

A graph can be further transformed into a mesh by considering faces F and additional constraints to verify the 2D manifold properties. A bijective mapping should exist between the local neighbourhood of any point of the mesh and a 2D disc. A mesh is then defined as a triplet $M = (V, E, F)$ where $F = \{f_k | k = 1, ..., t\}$ is a finite set of faces, each defining a polygonal surface connecting together several vertices. The most common mesh structure is called triangle mesh in which all the polygons are triangles.

3D meshes are the standard representation in Computer Graphics as the hardware (GPUs for Graphics Processing Units) is specifically optimized for this 3D representation and the connectivity allows to have the normal information for any surface point, which is very useful to compute lighting effects.

Similar to point cloud, applying deep learning techniques to graphs or meshes is challenging due to the irregularity nature of this data, but important progress has been made recently. Many approaches tried to learn 3D shape properties from graphs with the introduction of Graph Convolutional Neural Networks (GCNN) that can be categorized into two main families: spatial filtering methods [166] and spectral filtering methods [18]. For meshes, Feng et al. [46] proposed a mesh neural network that performs very well on shape classification and retrieval and Hanocka et al. [65] used edges to define specialized pooling and convolution operations and demonstrated good performance for mesh classification and segmentation.

2.3 3D Reconstruction of Humans From a Few Input Views

In this section, we study the recent advances for the task of reconstructing the complete 3D geometry of a human from only a few input views.

When much more input views are available, traditional reconstruction pipelines described in Section 2.4 can be used to capture the 3D geometry of a human. However, they tend to fail when only a few input viewpoints are considered. Indeed, if these latter are too far from each other the appearance matching is impossible and if they are close, only a small part of the shape is reconstructed.

As a result, a significant amount of research was dedicated to 3D human reconstruction from a few views. This context is also more conceivable in practice and at a lower price. To perform 3D reconstruction from a reduced number of views, all the methods rely on 3D human priors that can be statistically learned with parametric models or learned from data with the recent advances in deep learning. In the next sections, we present these approaches grouped in categories depending on the 3D representation that is involved.

2.3.1 Parametric Models

In the past, many works proposed parametric models to represent the geometry of the human body by using dimensionality reduction. It started with the use of a set of simple geometric primitives such as planar rectangles [137], cylinders [88, 122, 159, 171, 172, 193] or ellipsoids [196] and then statistical human models were learned directly from 3D scan data such as SCAPE [6], S-SCAPE [81] or SMPL [116]. The common idea of these latter is to obtain a low-dimensional representation that includes two sets of parameters to control the pose and shape deformations of the human body and learn them directly from a multi-pose and a multi-shape datasets.

To obtain more complete models, SMPL-H [160] combines the human body model SMPL with the hand model MANO and SMPL-X [150] extends SMPL with the MANO model and the FLAME model [106] to jointly represent the human body with the hands and the facial expressions.

Works that estimate the pose and shape parameters of these parametric human models from images can be categorized in two families: optimization-based and regression-based methods.

Optimization With Parametric Models

Balan et al. [7] leveraged the low-dimensional parametric model SCAPE and defined a cost function between the hypothesized mesh and the image observations. An optimization based on a stochastic search is then used to estimate the pose and shape parameters of SCAPE. Guan et al. [61] leveraged user-supplied information to obtain an initial segmentation and pose and shape parameters. Then, the 3D body shape is optimized using different cues such as silhouette overlaps, edge distance, and smooth shading. More recently in SMPLify, Bogo et al. [13] optimized the SMPL pose and shape parameters by using the 2D human joint detections estimated by a Convolutional Neural Network (CNN) from a single image.

All these methods provide reliable results but as an optimization problem, they usually strongly depends on the initialization, may fall in local minima and are time-consuming.

Regression With Parametric Models

More recently, many works used deep learning techniques to directly train a network for the task of estimating the pose and shape parameters of a parametric human body model from images. The training phase is usually long and requires a large amount of annotated data, but the inference is just a

simple forward pass in the neural network. Pavlakos et al. [149] proposed an end-to-end method for 3D human pose and shape estimation from a single RGB image. One of their main contribution is the incorporation of the parametric model SMPL inside the end-to-end framework. In particular, it allows to predict shape and pose parameters from silhouettes and 2D keypoints and to recover a 3D mesh during training. With this predicted mesh, they can use a per-vertex loss and integrate a differentiable renderer to refine it with respect to 2D annotations. This supervision based on 2D annotations is used to balance the relative lack of 3D ground truth data for training. In the concurrent work HMR, Kanazawa et al. [90] proposed an adversarial strategy with a discriminator that ensures the validity of a predicted mesh. It allows training the deep neural network only with 2D labels which are relatively easy to obtain compared to 3D ground truth, and they showed superior performance for the setting in-the-wild. Kolotouros et al. [96] chose a different strategy and regressed the SMPL template vertex positions instead of SMPL parameters. The motivation is that even if the predicted mesh is consistent with the input image when SMPL parameters are regressed, the performance in terms of pose estimation is lower that non-parametric solutions. The authors used a fixed template mesh with a Graph-CNN architecture and demonstrated state-of-the-art results in terms of pose and shape estimation.

While parametric models allow for robust and possibly fast reconstructions of the human body from only one or a few images it is limited to a coarse naked human body. The level of detail and variability of the reconstructed clothing and accessories remain inherently limited.

2.3.2 Meshes

To alleviate some limitations of parametric models and represent clothing and details such as wrinkles or facial expressions, several works proposed to deform a 3D template mesh. Xu et al. [207] proposed, in particular, to deform mesh vertex positions using a warp field parameterized by an embedded deformation graph and supervised by silhouette segmentation. Habermann et al. [63] proposed a two-stage strategy where a human mesh template is first fitted to the input frame, and then, the vertices are non-rigidly deformed using dense photometric and silhouette constraints over multiple frames. Habermann et al. [64] introduced a method that combines a pose-estimation network which regresses the skeletal pose of a human with a deformation network that estimates non-rigid deformation of the dense surface. Zhu et al. [236] created a large dataset of basketball players meshes derived from the NBA2K19 video game. From that, they can train an identity and a skinning neural networks that predict respectively vertex offsets and skinning weights to transform a rest pose template into a personalized mesh in a specific pose. An interesting alternative strategy for the problem of full human body prediction from a single image was proposed by Alldieck et al. [5]. The authors considered an image-to-image translation neural network that estimates normal and vector displacement maps from a single RGB image and which can be applied on top of a coarse initial mesh reconstruction.

Even if deforming a template mesh provides more opportunities to capture details such as facial expressions and clothing wrinkles, a mesh remains limited to a single topology and cannot generalize to various types of clothing (*e.g.* pants, skirt, dress, coat, ...).

2.3.3 Volumetric Representations

As explained in Section 2.2.2, volumetric representations can take the form of voxels or more efficient versions with Octrees or KD-trees. In contrast to meshes, they are not limited to a single topology and their regular 3D grid-based structure makes them very suitable to be processed with recent 3D deep learning techniques. A whole line of research was dedicated to reconstruct humans in 3D from images. BodyNet [189] was one of the first methods to integrate a volumetric representation in an end-to-end trainable neural network for 3D human body shape prediction from a single colored image. Several losses are used for supervision: 2D segmentation and pose losses on intermediate representations, a 3D pose loss on 3D joints, a volumetric loss on each voxel, and a multi-view reprojection loss on the silhouettes to increase the importance of boundary voxels. Gilbert et al. [58] proposed a coarse-to-fine strategy to reconstruct 3D humans from sparse multi-view videos. For that, they voxelize an initial Visual Hull reconstruction and encode it in a latent representation. A volumetric decoder is then used to obtain an improved version of the initial reconstruction with higher fidelity with respect to the input images. Similarly, Zheng et al. [232] also proposed to start from an initial volumetric representation and use a voxelized version of an SMPL mesh for that. Their volume-to-volume translation network uses multi-scale features extracted from the input images to refine the initial volume representation.

Despite good results for the task of 3D human reconstruction from a few images, the volumetric representation remains limited in terms of resolution and computation costs which prevent the reconstruction of high quality details.

2.3.4 Depthmaps

In order to reduce the large memory requirements of the volumetric representation, some methods explored alternative representations such as depthmaps. In Moulding Humans, Gabeur et al. [52] predicted two depthmaps from a single RGB image, one for the visible surface, usually the front of the human, and one for the hidden part, usually the back of a human. These depthmaps can be later fused into a point cloud and converted to a mesh. In FACSIMILE, Smith et al. [175] also introduced a depth inference network that predicts front and back depthmaps from a single RGB image. A key component of their depth network is a spatial differentiation module that converts depth into normals and allows supervision with normals even in the absence of depth ground truth. Finally, they proposed a mesh alignment technique to match a SMPL template mesh with the point cloud obtained from the depthmaps.

Depthmaps provide an interesting alternative for voxel-based representations with a much lower memory requirement and their regular-grid structure also makes them very suitable for deep learning techniques. However, a

depthmap is still a 2.5D representation and additional steps are required to recover a full human avatar such as a fusion and meshing steps or a fitting process of a template mesh.

2.3.5 Implicit Functions

More recently, several works also explored implicitly-defined continuous neural representations that offer several advantages compared to explicit representations, as mentioned in Section 2.2.3.

A seminal work that used this representation to reconstruct humans from monocular images is PIFu [162], which learns pixel-aligned implicit functions to locally align image pixels with the global location of the 3D human. It can also deal with sparse camera views by simply averaging the contribution of each camera. Thanks to the underlying implicit representation, PIFu achieves complete and more detailed reconstructions of a 3D human from a single or a few images compared to the other methods based on different 3D representations.

In PIFuHD, Saito et al. [163] built on top of PIFu to deal with high resolution images in order to capture more geometric details. For that, they proposed a multi-resolution pipeline that learns two implicit functions, one for a low resolution occupancy and one for a high resolution occupancy. PIFuHD also includes a pretrained image-to-image translation network that predicts front and back normal maps which are given as additional input to the fine reconstruction network. With GeoPIFu, He et al. [67] proposed to improve the geometry predicted by PIFu by using latent voxel features that are processed by a 3D U-Net [31] network. These features allow to resolve some ambiguities in the query point encoding and also serve as a global 3D human prior and regularizer to increase robustness. In StereoPIFu, Hong et al. [73] proposed to integrate the geometric constraints of stereo vision with the implicit representation of PIFu. It allows to recover the 3D shape of clothed humans from a pair of rectified images. With their experimental results they demonstrated improved robustness, completeness and accuracy. In PAMIR (Parametric Model-Conditioned Implicit Representation), Zheng et al. [233] proposed to combine a parametric body model with the free-form deep implicit representation used in PIFu. The authors explained that this combined representation is a key contribution to address the ill-posed problem of reconstructing a 3D human from a single RGB image. In particular, they use the Graph-CNN proposed by Kolotouros et al. [96] to predict a initial SMPL estimation from the input image which is then voxelized and processed by a 3D encoder to extract voxel-aligned features. The latter are combined with the standard pixel-aligned features and given as input of the MLP decoder that implements the implicit function. Some works also built on top of PIFu to obtain an animation-ready avatar from a monocular image such as ARCH [78] or ARCH++ [68] in which the authors introduce a learnable semantic space and deformation field in the PIFu pipeline that can transform any human body to a canonical rest pose. Yang et al. [213] also reconstruct an animatable 3D human from an image by learning three implicit functions to obtain a skinning field, a pose field and the occupancy field from PIFu.

Another interesting research direction for methods based on implicit functions is the efficiency in terms of computation time. Even if an implicit representation offers the advantage of low memory footprint, the inference requires to evaluate the neural network at every position in a 3D grid which prevents the deployment for real-time applications. To address this point, MonoPort [105] introduced a novel hierarchical surface localization algorithm to drastically reduce the number of queried points during surface reconstruction without sacrificing final geometry quality. They also proposed a direct rendering technique from novel viewpoints of the captured human that is possible without reconstructing explicit geometry. More recently, NeuralHumanFVV [181] proposed a coarse-to-fine multi-stage pipeline to reconstruct the 3D human geometry from a few images. In particular, they show how to generate the geometry explicitly only in the novel views instead of the whole human geometry which is expensive and unnecessary for real-time applications.

In this thesis, and more particularly in Chapters 3 and 5, we also leverage an implicit representation to reconstruct accurate and complete models of dressed humans. In contrast to the existing works, we focus on a reconstruction scenario with a few input views and contribute on several points including an efficient combination of the information coming from the different views, spatially consistent reconstructions, improved generalization with a local context encoding and better performances with the integration of multi-view constraints.

2.4 3D Reconstruction From Multi-View Images

In this section, we discuss 3D reconstruction of arbitrary objects from dense viewpoints. We start by presenting the photo-consistency, a common key component of all Multi-View Stereopsis approaches, and then we review the three main categories of works that address this problem: traditional MVS methods in 2.4.2, data-driven approaches in 2.4.3 and more recent methods based on differentiable rendering in 2.4.4.

2.4.1 Photo-consistency

In the following, we present the concept of multi-view photometric consistency, also known as photo-consistency in short, which is a key component of every MVS algorithm. It consists in measuring the agreement between the projected appearances of a 3D point in several images. This concept is known for a long time [123], but the term consistency was first introduced by Seitz and Dyer [168] as voxel consistency and later renamed photo-consistency [98]. Originally, this term was used under the Lambertian assumption which assumes that a 3D point of a shape that is visible in multiple images is photoconsistent if the colors corresponding to the 2D projection of that point in the images are exactly the same. This assumption is however difficult to guarantee in real scenarios as it depends on the viewpoints, materials, illumination and sensor

noise. Another critical aspect for the photo-consistency measure is to have access to the visibility information such that appearances computed from a set of cameras actually see the same 3D point. However, this introduces a chicken-and-egg dependency as the photo-consistency requires visibility to reconstruct the geometry and the visibility information can only be accurately computed from the true geometry. Several approaches exist to overcome these limitations.

Kutulakos and Seitz [98] introduced a very simple photo-consistency measure by considering the color variances of the pixels corresponding to the 2D projections of a 3D point across the views. This strategy provided very limited robustness to the different types of noise that appear in real life scenarios such as reprojection errors, sensor noise, blur, vignetting or when the Lambertian assumption is not verified. To account for this noise, many photo-consistency metrics consider a local region where the appearance should match instead of a single pixel. Given a set of N input RGB images and a 3D point X that is seen by the N images, the photo-consistency function can be defined as follows by considering each pair of images I_i and I_j:

$$C_{ij}(X) = \rho(I_i(D(\pi_i(X))), I_j(D(\pi_j(X)))), \tag{2.3}$$

where $\rho()$ defines a similarity function computed between vectors, $\pi_i(X)$ represents the projection of a 3D point X into the image I_i, and $D()$ defines the local region. The simplest way to define this local region is to use a 2D patch of pixels around the 2D projection of the 3D point. The size of this region is an important parameter as it defines a trade-off between uniqueness and invariance. The appearance of a large region is more unique which simplifies matching with the other images but invariance with respect to illumination and viewpoints is harder to guarantee. Many similarity functions ρ can be computed between the appearance information of two image patches I and J of the same size such as Sum of Squared Differences (SSD) which is defined as

$$\rho_{SSD}(I, J) = \sum_u \sum_v (I_{uv} - J_{uv})^2, \tag{2.4}$$

where u and v index each pixel of the two patches. Sum of Absolute Differences (SAD) is a variant that considers the absolute difference between the pixels

$$\rho_{SAD}(I, J) = \sum_u \sum_v |I_{uv} - J_{uv}|. \tag{2.5}$$

Normalized Cross-Correlation (NCC) is defined as the cross-correlation between the two patches I and J normalized by the product of their standard deviations σ_I and σ_J,

$$\rho_{NCC}(I, J) = \frac{\sum_u \sum_v (I_{uv} - \hat{I}_{uv})(J_{uv} - \hat{J}_{uv})}{\sigma_I \sigma_J} \tag{2.6}$$

where $\hat{I}_{uv}$ and $\hat{J}_{uv}$ are respectively the mean intensity values of the two patches I and J. Various other metrics also exist such as Sum of Hamming Differences (SHD), Census transform (CT), Rank (R) or Mutual Information (MI) and more information is available in [51]. The general objective of these similarity metrics is to provide some robustness to various types of noise coming from

the camera sensor or the change of illumination or brightness across viewpoints. Another trend for photo-consistency functions is to use gradient-based image descriptors such as SIFT [118], GLOH [128], SURF [10], BRIEF [20] or Daisy [186]. More extensive studies and evaluations of these image descriptors are available in [72, 118, 128, 186, 199]. These descriptors combine the response of several 2D filters, usually at multiple scales, to obtain a descriptor invariant to the changes of viewpoint or illumination or other transformations such as scaling or rotation. Such gradient-based image descriptors demonstrated better robustness in case on noisy photometric information [102] but are usually more complex to compute compared to the standard photo-consistency functions such as NCC.

With the objective of improving the robustness of the hand-crafted photo-consistency functions several works explored data-driven methods. They usually build on top of deep learning techniques and learn a similarity function directly from real images. In that case, the neural network can choose what information to consider in order to obtain invariance for the changes of illumination or brightness due to non-Lambertian surfaces and robustness with respect to noise or occlusions. Several works in the domain of Stereo Matching [121, 224, 225] proposed to learn how to compare 2D image patches by training CNN. They usually consider two rectified patches as input and use a Siamese neural network to extract features from these patches. Then, a final decision network predicts the similarity score. This technique was also explored by MVS approaches. Hartmann et al. [66] proposed a similar architecture but consider multiple input patches as input instead of pairs. Leroy et al. [103] proposed to replace the 2D patches by the projection of a 3D volume back-projected from a reference image. The authors explained that this strategy allows to capture more local geometric patterns, takes into account the relative positions of the cameras and is more efficient under complex dynamic conditions.

2.4.2 Traditional Multi-View Stereo Methods

A first category of seminal methods [16, 35, 98, 168] for MVS reconstruction used a voxel grid representation and tried to estimate occupancies and colors. One notable technique is Space Carving [98] in which matter is iteratively removed based on the consistency of pixel colors in multiple views. While efficient their reconstruction precision is inherently limited by the memory requirement of the 3D grid when increasing resolution.

Another line of works explored global optimization methods. Faugeras and Keriven [44] proposed a global optimization based on a variational principle that must be satisfied by the surfaces of the objects and the images. Similarly, Fua and Leclerc [49] and Fua [48] used a photometric criterion that is minimized in a global optimization of meshes or particles. Inspired by these works, several other methods were then developed, in which the authors tried to deform an initial shape, usually a mesh, by minimizing a reprojection error [36, 152].

In contrast to the global methods, another tendency gathers more local methods. Furukawa and Ponce [50] proposed an interesting strategy that gen-

erates and propagates a set of dense patches followed by a Poisson Surface reconstruction [94]. Hiep et al. [71] combined a point cloud generation based on local detections with a more global graph-cut based optimization.

Later the tendency switched to depthmap-based MVS methods that usually try to match image features from several views to estimate depths. Additional post-processing fusion and meshing steps [33, 93, 94, 125] are required to recover a surface from the multi-view depthmaps. Despite the usually complex pipeline, multi-view depthmap estimation gives access to pixel-accuracy, a strong feature for the reconstruction quality which has made this representation common in MVS approaches. Campbell et al. [21] used a volumetric graph-cut strategy to improve estimated depthmaps by removing outliers. With improved depthmaps the authors relax the requirement of strong redundancy among input depthmaps for the fusion stage which allows to work with sparse datasets. More recently in GIPUMA, Galliani et al. [54] proposed an extension of the PatchMatch Stereo algorithm [12] to directly search for correspondences in multiple views. They also adapted their method such that it can be efficiently parallelized on GPUs. Schönberger et al. [167] extended the probabilistic framework from Zheng et al. [230] by jointly estimating depth and normal information per-pixel, introducing a pixelwise view selection strategy that uses both photometric and geometric priors. They also added a multi-view geometric consistency term for depths and normals fusion. This work is now integrated into a popular Structure-from-Motion and MVS framework called COLMAP. Several improvements were also proposed such as an adaptive checkerboard sampling to propagate good depth hypotheses as soon as possible [204] and a planar prior in the PatchMatch multi-view stereo framework to solve some ambiguities in low-textured areas [205].

2.4.3 Data-Driven Multi-View Stereo Methods

With the recent advances in deep learning, several methods proposed to learn some parts of the MVS pipeline such as the image feature matching [66, 104, 224] or the standard depthmaps fusion step [39, 157]. Others even proposed to learn the full pipeline in an end-to-end manner. Here, we focus on these latter and distinguish two categories depending on the representation they use. Similarly to the traditional methods, several deep learning-based approaches consider depthmaps which are filtered, fused and potentially converted to a mesh in post-processing steps. Another trend of works considers a volumetric representation for which the deep neural network directly predicts occupancy or signed distance values. It simplifies the post-processing steps as no fusion is required, but a 3D volumetric representation is expensive in memory which can hinder the capture of high quality details. For that reason, the majority of recent deep learning-based strategies consider depthmap predictions. Here, we start by reviewing volumetric-based approaches, and then, present methods that predict depthmaps.

Volume-based Methods

SurfaceNet [83] was the first attempt to propose an end-to-end approach for MVS. Their deep neural network directly processes pairs of volumetric grids, called Colored Voxel Cubes (CVC), that are constructed for each input view by back-projecting pixel values in 3D and predicts binary surface presence for any voxel in a CVC. Kar et al. [91] also used an internal 3D voxel representation to learn the MVS process. In particular, they built per-view 3D feature volumes by back-projecting 2D image features in 3D and used a recurrent neural network to match the different volumes together. The main limitation of these two approaches comes from the 3D discrete grid representation as the number of voxels grows cubically with the size or the resolution of the scene.

More recently, some works of this category tackled the problem of indoor scene reconstruction from a sequence of images. In Atlas, Murez et al. [132] directly regressed a 3D Truncated Signed Distance Function (TSDF) for the full scene. For that, they accumulated features in a 3D grid volume by back-projection and processed them with a 3D CNN to predict per-voxel TSDF values. The full 3D volumetric representation and the 3D CNN limits the size and the level of detail of the reconstructed scene and prevents from real-time applications. In NeuralRecon, Sun et al. [179] proposed a solution to these issues by considering a sparse TSDF representation, sparse 3D convolutions, a Gated Recurrent Unit (GRU) and a coarse-to-fine strategy. Their framework can reconstruct in 3D an indoor scene from a monocular video in real-time. More recently Bozic et al. [15] proposed a similar coarse-to-fine strategy combined with a transformer-based feature fusion module. In particular, the latter learns which coarse and fine features are the most relevant for the reconstruction problem.

Depthmap-based Methods

A large number of deep learning-based methods for MVS predict depthmaps which are then filtered and fused into a point cloud. The depthmaps are predicted for a reference view by using other views called the source views. During inference, each input view is sequentially considered as the reference view to predict the corresponding depthmap. Different strategies exist to efficiently select source views with sufficient parallax, overlap and reduced occlusions. They usually rely on geometric information such as camera positions or angles between camera principal axes and photometric information.

The majority of depthmap-based methods share a standard pipeline introduced first in MVSNet [214]. The main component is a cost volume constructed in the frustum of a reference camera by a sweeping strategy called *plane sweeping*. Virtual fronto-facing parallel planes are positioned at multiple depth hypotheses, in which, features extracted from the reference and the sources views are matched by a differentiable homography. To aggregate these features coming from the different views, MVSNet uses a cost metric computed as the element-wise variance of all the features. Then, the raw cost volume is regularized by a 3D Unet to account for noise coming from different sources such as occlusions or non-Lambertian surfaces. Finally, a $Softmax$ operation is used to obtain a probability volume and the final depthmap is computed as

the expectation value of the depth hypotheses along each ray.

A very large number of methods were inspired by MVSNet and proposed improvements on each component of the standard pipeline.

2D Feature Extraction Several works proposed to replace the 2D feature extractor of MVSNet with more efficient modules or networks such as Spatial Pyramid Pooling (SPP) [79], Feature Pyramid Network [60, 208] or attention mechanisms [211, 222, 227].

Cost Volume Construction Several alternatives to the variance-based strategy from MVSNet were explored to aggregate the features from the different source and reference views in the cost volume. DPSNet [79] directly concatenated the features from the reference and each source views and averaged over all such pairs. Luo et al. [119] proposed to construct patch-wise matching cost volumes which improves matching accuracy and robustness. Several approaches also consider more information than just local perceptual features when creating the cost volume such as contextual [120], semantic [76] or visibility [26, 195, 206, 226, 227, 228] information.

Cost Volume Regularization The original cost volume regularization strategy of MVSNet based on 3D convolutions is very memory-consuming which limits the scalability of the pipeline. To overcome this, several works introduced a Recurrent Neural Network (RNN), such as an LSTM [209] or a GRU [110, 215] to sequentially regularize each 2D slice of the cost volume along the depth direction. PatchmatchNet [195], an end-to-end learnable pipeline inspired from the seminal Patchmatch algorithm [8], also avoided the costly 3D cost volume regularization by learning adaptive propagation and spatial cost aggregation modules.

Coarse-to-fine Architectures Coarse-to-fine architectures were introduced in end-to-end MVS pipelines by several works [25, 28, 60, 212, 227] to reduce the memory consumption and the computational cost. These architectures take as input a pyramid of images or features and first construct a low-resolution cost volume. At this coarse level, a very large depth range is used to create the depth hypotheses and coarse depthmaps are then predicted and used to narrow the depth range for the finer level. The coarse level better captures the global shape and the fine level captures high frequency details.

All these works on data-driven multi-view stereo, have progressed a lot recently and constitute a very promising direction for the future. However, some challenges still remain such as the generalization to never-seen-before data or the large memory consumption when high resolution images are used.

2.4.4 Differential Rendering

Another line of works for 3D reconstruction from multiple images has explored differentiable rendering approaches. A large majority of these works are used

for novel view rendering applications, however, several also focus on 3D shape geometry reconstruction.

They build on a rendering process that is differentiable and henceforth enable shape model optimization by differentiating the discrepancy between generated and observed images. These methods were originally applied to various shape representations including meshes [70, 92, 112, 146], volumetric grids [53, 86, 139, 188, 235] or even point clouds [80, 84, 135]. In association with deep learning, new neural implicit representations have also emerged. As described in Section 2.2.3, their continuous nature and light memory requirements are attractive. They have been successfully applied to different tasks: 3D reconstruction [126, 143, 151, 162, 163, 202] or geometry and appearance representations [57, 127, 141, 174, 182]. Most of these methods solve for 3D shape inference and require 3D supervision, however, recent works combine implicit representations with differentiable rendering.

Nerf [129] is the pioneer work that combines neural representations with volumetric differentiable rendering to produce high quality photo-realistic synthesis of novel views. It requires a long optimization phase and a large number of input views. It also inspired a very large number of works that tried to extend its capabilities or overcome some limitations. Some works improve the rendering quality [9, 62, 192], deal with dynamic scenes [107, 144, 145, 153], incorporate additional sparse [38, 203] or dense [158, 198] depth supervision, generalize to novel scenes [23, 87, 221] or even accelerate the optimization with more efficient representations or optimization strategies [24, 69, 131, 155, 178, 220]. More details can be found in different very complete surveys [37, 55, 184, 201]. However, they all target the novel view rendering problem and the quality of the associated geometry as encoded with densities is however not perfect and often noisy. The estimated geometry still lacks precision as the methods are not primarily intended to perform surface reconstruction.

More related to this thesis, several works also focused on that aspect to obtain better 3D surface reconstructions. They roughly belong to two categories depending on the shape representation they consider.

Volume-based Methods The methods in this category follow the same principle as the original Nerf but replaced the geometry representation based on the density with a more suitable one based on a Signed Distance Function (SDF). In NeuS, Wang et al. [197] compute the density based on a transformation of the SDF and in VolSDF, Yariv et al. [218] model the volume density as a Laplace cumulative distribution function applied to an SDF. Both proposed more accurate surface reconstructions than the methods using the original Nerf density. Darmon et al. [34] even extended VolSDF and proposed a fine-tuning strategy based on image warping to take advantage of high-frequency textures. This method allows to incorporate traditional patch matching techniques from MVS into a differentiable rendering-based method. To capture accurate surfaces, all these methods replaced the original density-based representation of Nerf with an SDF-based representation. However, very recently, Toussaint et al. [187] showed that the original representation of Nerf can be used for accurate surface reconstruction if appropriate losses are added. They also

demonstrated that by combining a non-neural parameterization, a coarse-to-fine scheme and an explicit sparse storage they obtain very fast surface capture.

Surface-based Methods On the other hand, several works proposed to use a surface renderer that estimates 3D locations where viewing rays enter the surface and their colors. By making the shape surface explicit, these approaches obtain usually better geometries [95, 111, 140, 217]. Nevertheless, they are more prone to local minima during the optimization since gradients are only computed near the estimated surface as opposed to volumetric strategies. Interestingly, Oechsle et al. [142] proposed a hybrid approach that combines the advantages of both volumetric and surface rendering and obtained good surface reconstructions.

In Chapter 4 we propose a novel optimization method for this reconstruction problem of 3D shapes from dense input viewpoints. In particular, our method combines the benefits of MVS methods that can yield pixel-wise geometric accuracy with local depth predictions along viewing rays, the volumetric integration used in recent differentiable rendering-based reconstruction methods and a data-driven strategy to learn a photo-consistency criterion.

3

Data-Driven 3D Reconstruction of Dressed Humans From Sparse Views

3.1 Introduction

In this chapter, we examine the problem of 3D reconstruction of dressed humans from a limited number of views. The ability to produce accurate visual models of real humans in every-day context, in particular with their clothing and accessories, is useful in a wide range of applications that deal with captured human avatars, typically in the virtual and augmented reality or telepresence domains. Using images for that purpose has been an active field of research for decades, with issues that result, in part, from the high dimensionality of the space of human shapes and appearances, especially with dressed people. The challenge is accentuated when only few viewpoints are considered, a situation that is, on the other hand, common in many practical contexts, for instance with mobile devices. While model-based strategies (*e.g.*, SMPL [115]) have shown impressive results in case of undressed bodies, they cannot easily generalize to generic humans with clothing and accessories.

Acquiring 3D human models from images is a long-standing research topic in computer vision. When images from several viewpoints are available, multi-view stereo approaches (*e.g.* [50, 169]), and their learning-based extensions (*e.g.* [83, 103]), allow for highly detailed 3D reconstructions by combining multi-view information with photo-consistency criteria. This generative strategy builds on photo-metric redundancy among input images and tends to fail however in our context that considers only sparse input viewpoints. Besides, data-driven reconstruction methods, that only require a single view, have been proposed. This includes methods based on low-dimensional parametric models (*e.g.* [150]) which are anyway limited with clothing and accessories; methods based on volumetric representations (*e.g.* [189]) with bounded level-of-details by construction; and methods based on implicitly defined continuous neural

representations (*e.g.* [162]). These latter methods have demonstrated their ability to recover humans with clothing and accessories. Yet, the single-view reconstruction problem is highly ambiguous and results easily suffer from artifacts when the input scene differs substantially from the training set. To remedy this, methods accounting for multiple input views have been proposed *e.g.* [77, 162]. These extensions, however, merely combine single-view estimations with simple average pooling. Such ways of fusion do not fully exploit multi-view cues and are still plagued by single-view ambiguities.

In this chapter, we adopt the widely approved implicit neural representations and focus on multi-view fusion. With respect to single-view estimation this task raises several issues. First, single-view reconstruction methods generally assume a person centered and scaled input image. This needs to be compensated for when dealing with sequences of moving humans and in order to obtain spatially consistent reconstruction with coherent localization and scales among the sequence frames. The second question is how to aggregate local information from viewpoints that can differ significantly, for instance front and side-views, and which can therefore predict different occupancy at a given spatial location. The third issue is how to account for local contexts, defined by image color cues around a 3D point, that gain in variability with increasing views but also allow to better differentiate local geometric patterns. To address these issues, we propose a data-driven end-to-end approach that reconstructs a 3D model of the dressed human from sparse camera views using an implicit representation. Specifically, our method has three key components:

- A spatially consistent 3D reconstruction framework that allows for arbitrary placement of the human in the scene that uses the perspective camera model, achieved by learning the model in a canonical coordinate system and by accounting for the transformation of each input view to this system.
- A learnable attention-based fusion layer that weighs view contributions. This layer implements a multi-head self-attention mechanism inspired by the transformer network [190].
- A local 3D context encoding layer that better generalizes over the local geometric configurations, which is implemented through randomized 3D local grids.

In the experiments, we evaluate our approach against the state of the art on public benchmarks. To demonstrate the value of the spatially consistent reconstruction, we apply our method to dynamic scenes with large displacements. Moreover we also contribute with results on new real data obtained with a multi-view platform. They demonstrate the feasibility of data-driven approaches in practical real-world capture scenarios, even trained solely on synthetic data.

3.2 Method

In this section we first give an overview of our method and explain the representation that is used. We then present our strategy to learn and infer humans in a large scene and our contributions with the spatially consistent reconstruction, the attention-based fusion layer and the local context learning.

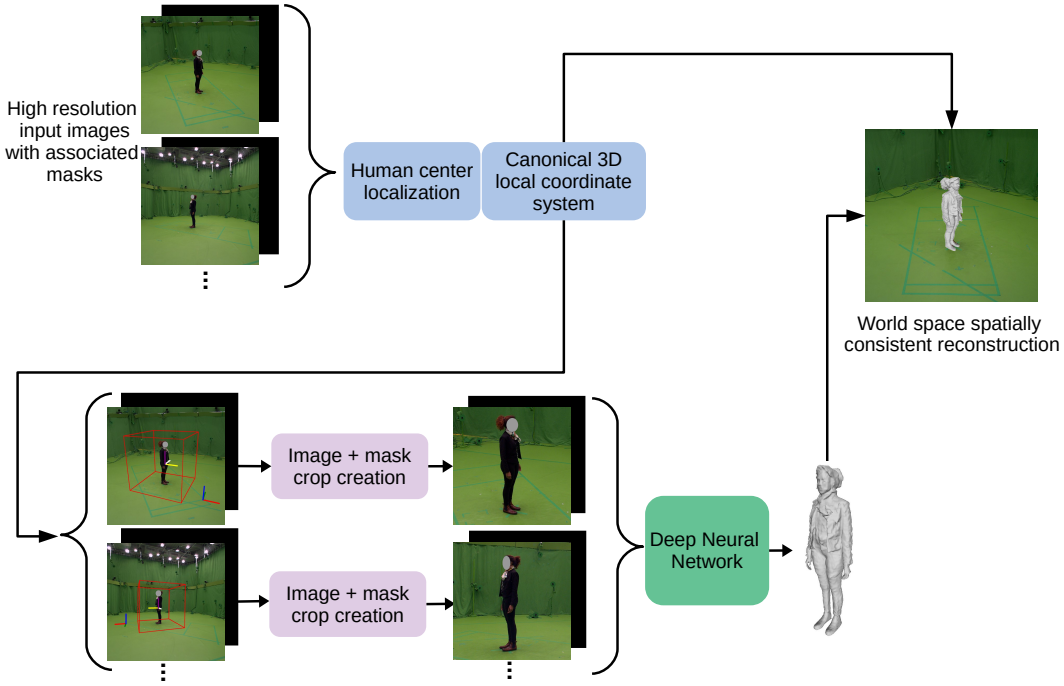


Figure 3.1: Overview of the proposed pipeline. Given a sparse set of input images with associated background masks and known calibration, our method reconstructs a spatially consistent 3D model.

3.2.1 Overview

Our pipeline is described in Figure 3.1. High resolution images of a human and background masks are used as inputs to reconstruct a spatially consistent 3D model using an implicit representation. To allow for a spatially consistent reconstruction with proper scales and localization, we learn the model in a canonical 3D local coordinate system, and transform each observation to this space. This is achieved by localizing the 2D center of the human in each view, by triangulating to find the 3D position of the human center, and by defining a canonical 3D local coordinate system based on this information. This allows to create canonical crops of the input images and background masks so they can be fed to our deep neural network that learns to predict an implicit 3D reconstruction in a canonical space. The result, combined with the canonical 3D local coordinate system, allows to reconstruct a spatially consistent 3D model in the scene by placing the reconstruction in world coordinates.

Fig 3.2 gives an overview of our deep neural network for multi-view 3D reconstruction. Image features are first extracted using a standard multi-scale image encoder. We then sample points by combining two strategies: random sampling in a 3D bounding box and importance sampling close to the surface with half of the points inside the mesh and the other half outside. We also construct a local 3D grid around each sample. Here we describe the method for a single sample but in practice a large number of points are processed in parallel. Using projection and bilinear interpolation, each point of the local grid is associated with a 2D feature, which is concatenated with the depth of the point. It is important to note that the previous steps are performed per-

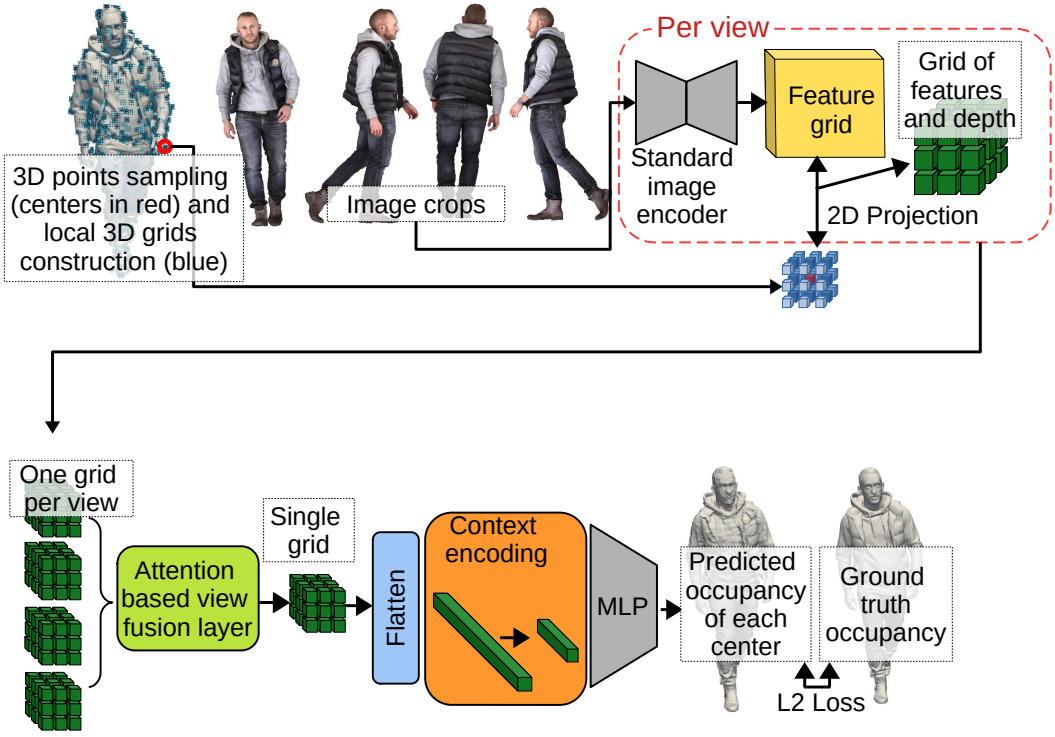


Figure 3.2: Overview of the deep neural network for multi-view training. Image features are extracted per view, and queried for a local grid around each sample. All views are integrated using an attention-based fusion layer, and a context encoding layer based on 3D convolution is applied before predicting occupancy.

view and in the end a 3D local grid of features is obtained for each view. An attention-based module efficiently combines the information from the different views by merging the 3D local grids. A second fully connected fusion layer extracts a final 3D feature from the local grid. At inference time, we define a grid at the desired resolution, evaluate the occupancy function at every grid location, and apply the Marching Cubes algorithm [117] with a pre-defined threshold of 0.5 to recover a 3D mesh.

3.2.2 Multi-View Implicit Surface Representation

Following recent progresses in learning-based shape modeling, we use an implicit 3D surface representation for the reconstruction task. Implicit surface representation converts arbitrary mesh surfaces into a function defined on a volume and allows for geometric details to be represented at arbitrary resolution. Furthermore, the use of neural implicit representations is memory-efficient and solves the main issue in other volumetric representations. Similar to methods like [162, 163], our implicit function takes the combination of pixel-aligned features with depth values as input and predicts an occupancy probability $o \in [0, 1]$.

When reconstructing from a single image, the conditioning on the depth is necessary to differentiate points on the same camera ray as their appearance features are the same. In our case with multiple views, associations of features

can discriminate points of the same view line, but the conditioning on the depth is still helpful to capture details as the spatial resolution of the features is limited. To optimally benefit from this conditioning, training examples should all be aligned so that the network can learn a prior of the depth from the training set. Therefore, even if we consider reconstruction in large scene, we work in a canonical local coordinate system during training and at inference. The origin of the coordinate system is defined at the center of each training mesh and its orientation is the same as the world coordinate system, so we have the following equation: $X^j_{local} = X_{world} + T_j$, where T_j is the translation between the world origin and the center of the j-th mesh. The exact definition of the center of a mesh is arbitrary but should be consistent for all the training examples. In practice, we use the median over all mesh vertices for the x and z coordinates and the mean between the highest and the lowest vertices for the vertical coordinate y. For each 3D point, the depth value given as input of the implicit function is its z-coordinate in the local coordinate system aligned with each of the cameras by applying the rotation R_i. The implicit function takes the form:

$$\begin{aligned} &f(E_I(K_i\,[R|t]_i\,X_w), z(R_i X_{local}); \theta) = o, \\ &[|E| \times \mathbb{R}] \mapsto [0,1] \end{aligned} \tag{3.1}$$

where X_w is the 3D point in world coordinates, K_i and $[R|t]_i$ are respectively the intrinsic and extrinsic parameters of the i-th camera, o is the occupancy probability at X_w, and $|E|$ the dimension of the 2D image feature. $E_I(...)$ is defined at any location in the image using bilinear interpolation of the values of E_I at pixel locations.

3.2.3 Spatially Consistent Reconstruction

Most existing works based on pixel-aligned features and implicit representation consider orthographic projection where the appearance of a subject is the same at any position in the scene. In single-view reconstruction, this simplified scenario removes the ambiguity between the size of the subject and its distance from the camera. On the contrary we deal with perspective projection like in real environments with the pinhole camera model. We consider the case where enough views are available to avoid the size versus distance ambiguity. To accommodate for perspective deformations, we augment the data during training by randomly placing the subjects in the scene. As we are learning an implicit representation in a canonical 3D local coordinate system, the reconstruction at inference is inconsistent with the world space. Previous work tackles this problem with a neural network that estimates the spatial transformation of humans from a single image [133]. In our context, we propose to take advantage of the multiple views and triangulate the 3D coordinates from multiple 2D detections of the center of the human as shown in Figure 3.1. The 2D center positions are known at training time and predicted during inference using a convolutional deep neural network. The exact definition of the center of a human should be coherent with the point used to define the origin of the canonical coordinate systems. To supervise this network, we can use a similar dataset as in the remaining pipeline. Knowing the 3D center position, we can

define a canonical 3D local coordinate system, perform the inference in that space and replace the result in world coordinates. Note that the height of the subject is preserved as we do not apply any normalization on the size of the meshes during training.

3.2.4 Attention-based Fusion Layer

Image-based reconstruction benefits from multi-view cues, *e.g.*, stereo vision, which should be combined before the reconstruction is carried out in order to avoid premature single-view decisions and therefore limit ambiguities. Each view provides a feature and the question is how to aggregate them. Concatenating all the features, while simple, does not appear optimal because the fused features may become large when many images are considered, making it impossible to learn from an arbitrary numbers of views. Concatenation also imposes an order between views, which is undesirable in practice.

Besides concatenation, fusion approaches based on statistics, such as sum-pooling [41], average pooling [56] or max pooling [177] were proposed in the literature. The advantages are simplicity and invariance to both the order and the number of views. However, pooling operation loses information about individual view contributions. In particular, views in which a point is visible are considered equal to views in which the point is occluded and, more generally, erroneous information from an input view will contaminate the final prediction.

We propose to go one step further by learning the fusion and contextualising the information from different views. Previous work [210] proposes a simple learned fusion layer that computes a normalized score for each view, for each channel of a global feature. The main limitation is that the score of each view is computed individually without taking into account the information from the other views.

Inspired by recent progress in natural language processing to learn from sequences, we propose an architecture based on the transformer network [190] which implements a multi-head self-attention mechanism and is described in Figure 3.3. One key component is the *scaled dot-product attention* which is a mapping function from a query along with a key / value pair to an output. The three vectors query $Q = M^q X$, key $K = M^k X$, and value $V = M^v X$ are the embedding of the original feature X parameterized by matrices M^q, M^k and M^v, respectively. The idea is to compute an attention score for each view based on a compatibility of a query with a corresponding key:

$$\text{Attention}(Q, K, V) = \textbf{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{3.2}$$

where d_k the common dimension of K, Q and V.

To allow the network to attend to different geometric patterns, we propose to use multiple heads. For that, Q, K and V are linearly projected h times and processed in parallel through a scaled dot-product attention layer. The results from the different heads are concatenated and finally projected once

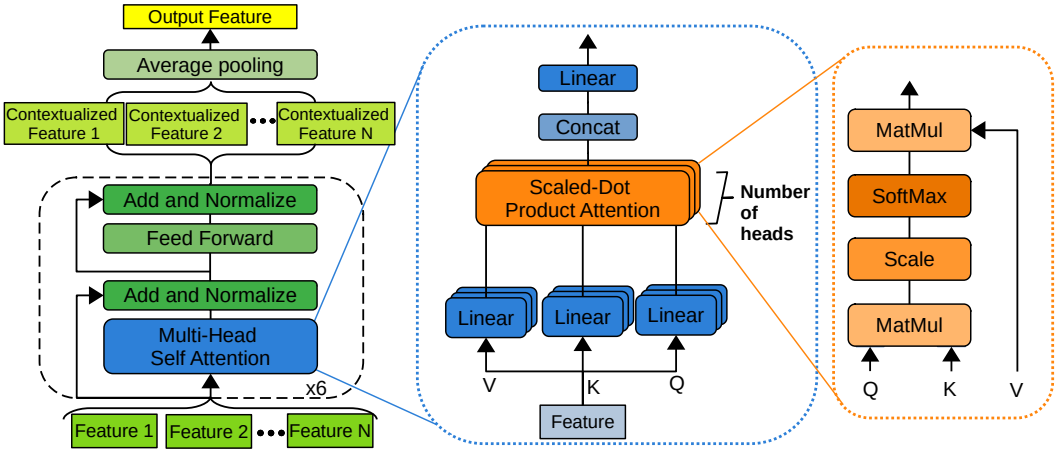


Figure 3.3: (Left) Our view fusion module. (Middle) Multi-Head Attention module. (Right) Scaled Dot-Product Attention.

again to obtain the final output :

$$\begin{aligned} &\text{MultiHead}(Q,K,V) = \textbf{concat}(H_1,...,H_h)W^o \\ &\text{with } H_i = \text{Attention}(QW_i^q, KW_i^k, VW_i^v) \end{aligned} \tag{3.3}$$

where W_i^q, W_i^k, W_i^v are respectively the parameters of the linear mapping of Q, K and V, and W^o the parameters of the final projection.

The output of the attention modules is a set of features. Each of them contains the original information from the corresponding view that now takes into account the information from all the other available views. Finally, we use the mean of these features as output of our view fusion module. Note also, that we do not use any positional encoding on the input feature sequence to remain invariant to the view order. Two options exist to implement multi-head self-attention. The standard narrow version splits Q, K and V into small chunks and each head processes one of them. On the opposite, the wide option propagates entirely Q, K and V to each head. This version provides superior performance at the expense of computation time and memory requirements. In our work, we choose the narrow option which offers a very good compromise.

3.2.5 Local 3D Context Encoding

In the proposed framework, projection is used to associate 3D points with 2D image features for each available view. Then, the attention-based fusion layer weighs the contribution of each view in the fused feature. Finally, a Multi-Layer Perceptron (MLP) predicts an occupancy probability. However, such features do not take the 3D geometric context into consideration since the neighbourhood is only considered in 2D when features are extracted from the images. To include 3D context, we propose to build a local 3D grid around each sampled point and associate each point of the local 3D grid with 2D image features by projection.

The attention-based layer is applied individually on each point of the local grids, after which we add another context fusion module that combines the

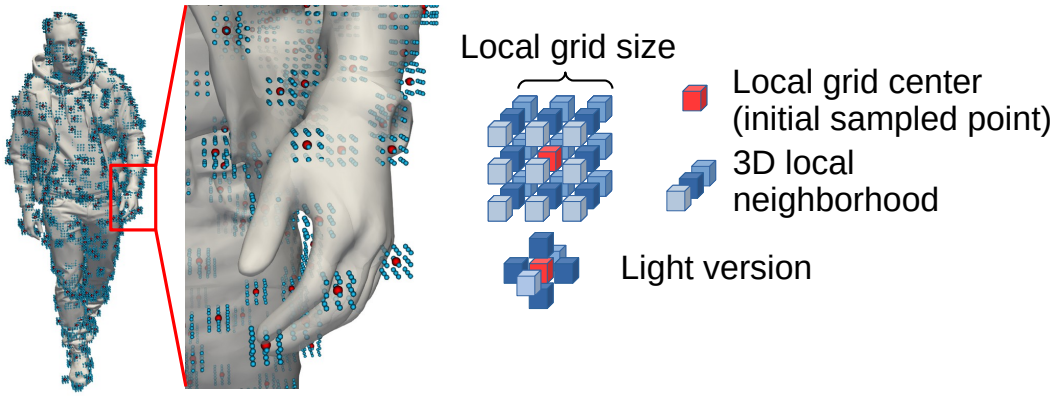


Figure 3.4: A local 3D grid is constructed around each sampled point (in red), and parameterized by a size and an orientation.

information coming from a 3D neighbourhood of a sampled point. This module is shown in orange in Figure 3.2, and is implemented with a fully connected layer. Thanks to this additional layer, the neural network is aware of the local 3D context of a point. In this way, we expect the network to better capture 3D geometric patterns and to increase robustness against nuisance factors (*e.g.*, texture, lighting).

As shown in Figure 3.4, the local grid is parameterized by the size S and orientation R. Empirically, we found that fixing R during training strongly links the local grid to the global coordinate system and the orientation of the human body. To remain invariant to the orientation of the human, during training we randomly align R with one of the available views at each iteration.

The grid size S needs to be chosen based on the training data and the type of the targeted 3D patterns. Our goal is to learn local 3D patterns that typically contain points in the same or close-by body parts. As a full local grid can be expensive in computation time and memory, we propose a variant that uses only the cells along the three grid axes that traverse the center of the grid. In that case, three one-dimensional vectors are considered instead of one three-dimensional grid, which significantly decreases the number of grid points while still allowing to take into account local context along three directions. We call this version "light" and use it in all our experiments.

3.3 Implementation Details

In this section, we give some implementation details about the reconstruction network, the human center localization strategy as well as the view selection during training.

3.3.1 Reconstruction Network

The image encoder of our reconstruction network is a Stacked Hourglass Network, with intermediate supervision, composed of 4 hourglass modules each of depth 2. The size of the output features is $128 \times 128 \times 256$. Since we trained

the network with a small batch size, we also introduced group normalization instead of batch normalization. Our view fusion layer is composed of 6 modules based on multi-head self-attention with 6 heads. The local 3D context fusion maps features from a $3 \times 3 \times 3$ grid into a single feature of size 256. The Multi Layer Perceptron (MLP) is composed of 6 layers of dimensions $256, 1024, 512, 256, 128, 1$ with skip connections between the first layer and all the other layers except the last one. We optimized our network during 100 epochs using the root mean square propagation algorithm with a learning rate of 1×10^{-4} that is divided by 10 at iterations 60 and 80. We implemented our method using Pytorch [148].

3.3.2 Human Center Localization

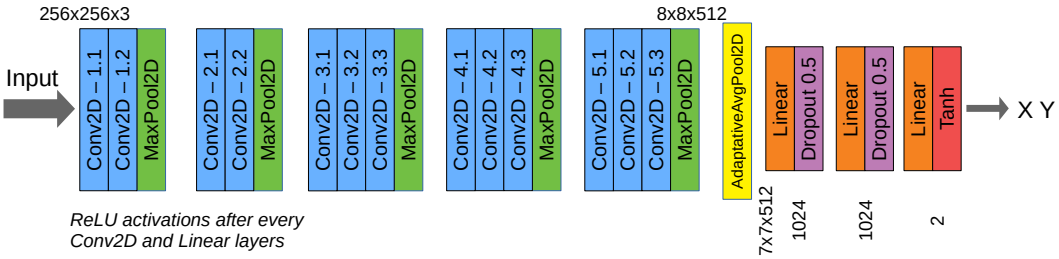


Figure 3.5: Human center detection network based on VGG16 [173].

In Figure 3.5 we show the architecture of our deep neural network based on the standard VGG16 [173] architecture to detect the human center on each of the view. These 2D detections are then used to triangulate the 3D position of the person in the scene. The center of the person is arbitrarily defined but should be coherent with the origin of the canonical coordinate systems used at training. In practice, we defined it as :

$$\begin{bmatrix} median(vertices.x) \\ 0.5 * (max(vertices.y) - min(vertices.y)) \\ median(vertices.z) \end{bmatrix}$$

where y is the up-axis. We do not use the median for the up-axis to account for cases where numerous vertices are grouped at the top or the bottom. Such cases are worth considering since a human is less symmetric with respect to the horizontal plane. In Table 3.4, we compute the L_2 distance between the 2D detections and 2D ground truth as well as the 3D positions triangulated from the 2D detection and the 3D ground truth. Here we used 4 views evenly distributed around the person with a random elevation axis between 0° and 45°.

3.3.3 Training Views

To train our deep neural network, we created a synthetic model view set by rendering 3D models from Renderpeople [2] using 360 cameras located around

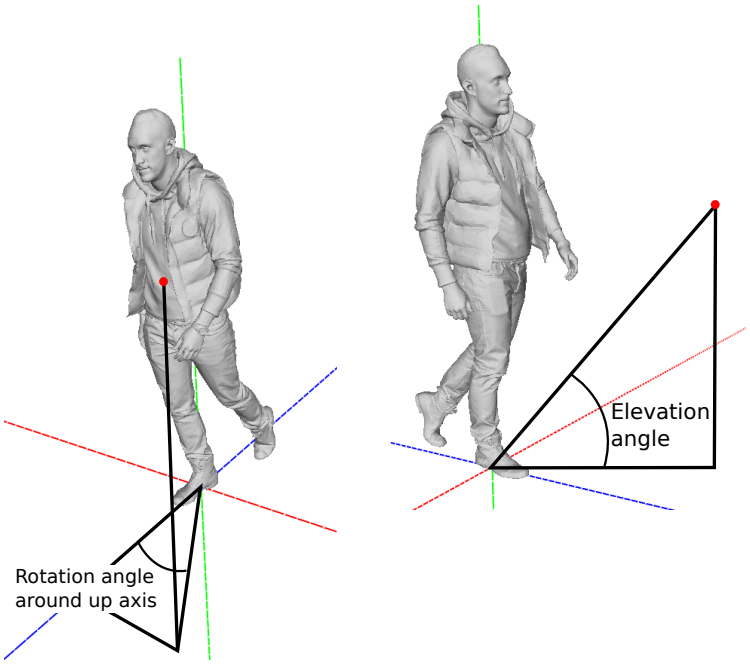


Figure 3.6: View selection angles.

them as explained below. In contrast to Multi-View Stereo methods, only a few of these views are considered at inference (between 2 and 6 in our experiments). The views used at inference should be ideally evenly distributed around the person in order to increase its visibility. At inference, results are most of the time better for parts of the surface that are observed than hidden ones for which the reconstruction relies solely on the prior learned from the training set. To build such image sets for the training we sample the synthetic views of a 3D model and create several model view subsets with few images.

To define the position of our cameras when creating such a subset, we use a rotation angle around the up-axis and an elevation angle, as described in Figure 3.6. For the orientation, we assume that the cameras are always looking at the center of the scene.

In practice, at each training iteration we choose N angles around the up axis that are evenly distributed among $[0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°]$ and add a random offset between $-20°$ and $20°$. The elevation angles are selected randomly between $0°$ and $45°$. Note that we trained our model with a fixed elevation angle when comparing with other methods (*i.e.*PIFu [162] and PIFuHD[163]) that consider a similar scenario.

3.4 Experimental Results

In this section, we evaluate the proposed method and compare it with the state of the art. First, we introduce the training and testing datasets as well as the evaluation metrics. We then compare our approach quantitatively and qualitatively against the current state of the art and provide a ablation studies to justify our contributions. Finally, we show results of spatially consistent

reconstruction and applications on real multi-view stereo data.

3.4.1 Settings

We create our synthetic dataset with Renderpeople [2], a public commercial dataset that provides highly detailed meshes obtained from 3D scans and corrected by artists. Its main advantage is the very high quality of the geometry which is essential to learn geometric details, especially with clothing. The humans from this set are in relatively standard poses and often hold accessories such as bags, cups or other objects. In total, we have 1026 meshes, split into 800 meshes for training, 100 for validation and 126 for testing.

To evaluate quantitatively the reconstructed human meshes, we first compute the Chamfer Distance (**CD**) between the ground truth mesh and the reconstructed mesh. By considering average distances between meshes, this metric tends to measure the global quality of the reconstructions. To focus more on local details, we also consider surface normal of the reconstructed and ground truth meshes and compute the $\mathbb{L}_2$ and cosine distances between them (**Norm Cosine** and **Norm L2**, respectively). Finally, in order to evaluate accurately the raw predictions of our network before the Marching Cubes post-processing that transforms the occupancy probability grid into a mesh, we compute the average $\mathbb{L}_1$ distance ($\times 10^3$) between predicted and ground truth occupancy (**Occ L1**).

3.4.2 Comparisons

Methods	**CD (cm) ↓**		**Occ L1 ↓**		**Norm Cosine ↓**		**Norm L2 ↓**	
	mean	median	mean	median	mean	median	mean	median
PaMIR [233]	0.554	0.508	1.977	1.754	0.097	0.090	0.361	0.343
PIFu [162]	0.592	0.510	2.079	1.773	0.103	0.093	0.376	0.358
PIFuHD [163]	2.008	1.624	5.837	4.543	0.181	0.162	0.544	0.503
Ours	**0.367**	**0.316**	**1.538**	**1.323**	**0.089**	**0.083**	**0.350**	**0.337**

Table 3.1: Quantitative results and comparisons with PaMIR [233], PIFu [162] and PIFuHD [163] on Renderpeople dataset. PaMIR, PIFu and ours use 4 views as input (see Figure 3.7) and PIFuHD uses a single frontal view. Best scores are in **bold.**

In the context of 3D reconstruction of dressed humans from a few sparse views, PIFu [162] demonstrated state-of-the-art results so we consider it as the baseline result. For the comparison we trained it on our training dataset. This method has proven its benefit against model-based reconstructions and we do not provide comparisons with the latter. PIFuHD [163] extends PIFu to high resolution images and shows impressive single view reconstructions of details for the visible parts. No training code is available, so we use the published pre-trained model for the comparison. PAMIR [233] combines the implicit

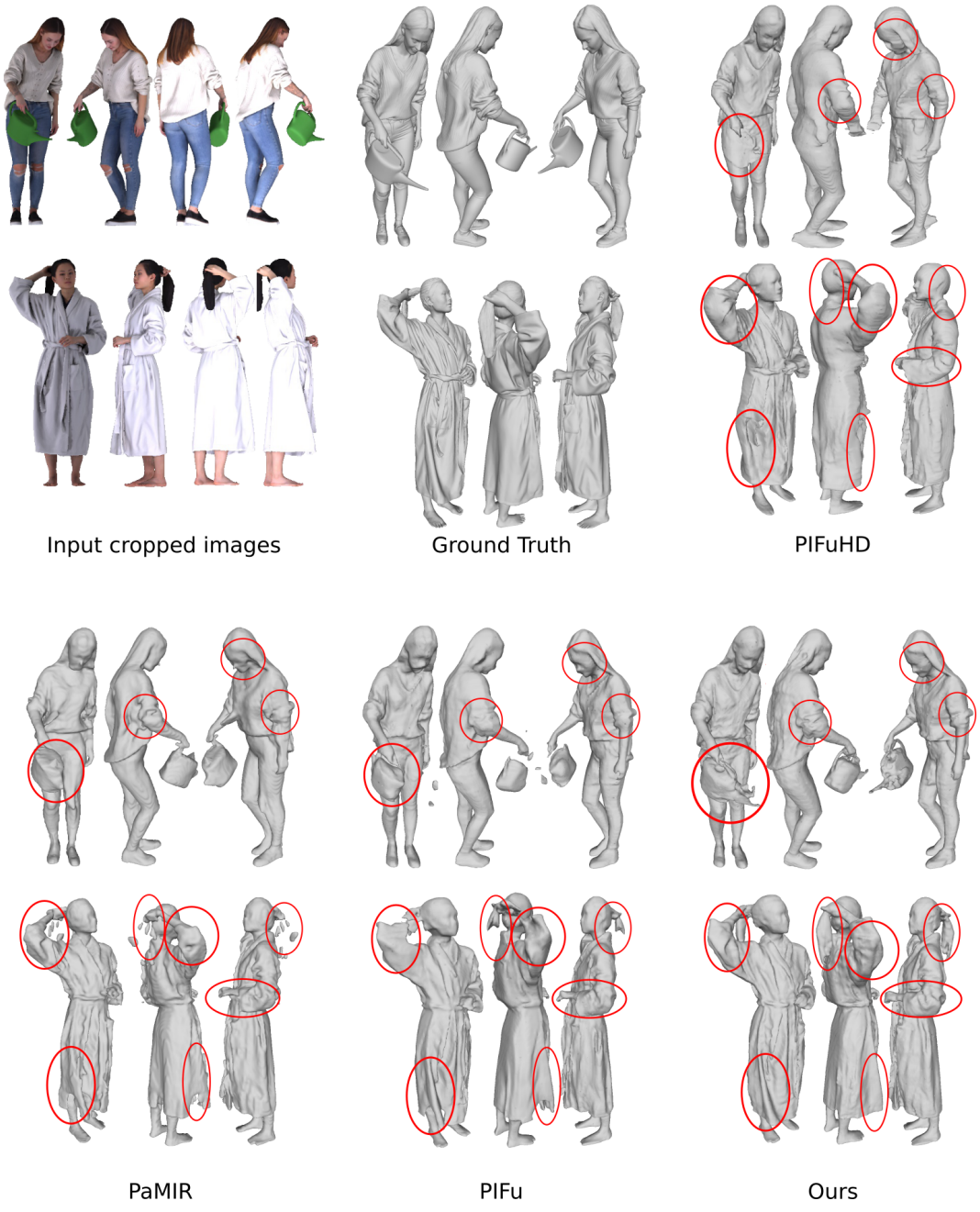


Figure 3.7: Qualitative results and comparisons with multi-view PIFu [162], multi-view PaMIR [233] and PIFuHD [163]. The 4 input images are rendered with the rotations around the vertical axis : 10°, 110°, 150°, 300°. PIFuHD uses a single frontal view as input.

representation with a parametric body model and shows improved single-view and multi-view reconstructions. The released code and pre-trained model are only for single-view reconstruction, so we implemented the missing parts ourselves and trained a multi-view model on our training dataset. We do not provide direct comparisons between our method and multi-view stereo (MVS) methods applied on the exact same input data since MVS methods fail when only few images are available. PIFu, PIFuHD and PaMIR use orthographic

images in which the human is at the center and cannot address the spatial consistency in world space. For a fair evaluation, we create a corresponding training / validation / test dataset composed of meshes from Renderpeople and evaluate all four methods on this data.

Qualitative results on synthetic data are presented in Figure 3.7. PIFu and PaMIR achieve promising reconstructions but fail on some parts like the hair and the arm in the first row, or the watering can and clothing wrinkles in the second row. Our method appears clearly more robust and captures more geometric detail as can be seen on faces and clothing wrinkles. PIFuHD achieves detailed reconstructions for the visible parts like the face but, unlike for our method, the quality decreases significantly for the hidden parts and the global shape is not respected like the head on both rows. This is inherent to single view reconstruction methods and emphasizes the utility of using multiple views.

This intuition is verified by the associated quantitative results in Tab. 3.1 that confirm the benefit of our method on three aspects. First, the global quality of the reconstructions is improved by a large margin with the Chamfer distance. Second, metrics on surface normal are also in line and show that local geometric details are better captured. Third, our method achieves better results on the raw values of the implicit function.

3.4.3 Ablation Studies

To evaluate the impact of our contributions, namely the multi-head self-attention fusion layer and the local 3D context encoding, we conducted qualitative and quantitative ablation studies. To isolate these contributions from eventual human center detection errors, we place here the human person at the center of the scene. For the first contribution, we replaced the view fusion module by a simple average pooling strategy and for the second, individual sample points were considered in place of the proposed local 3D grid.

Quantitatively, disabling the view fusion or the context encoding module both affect the reconstruction performance. From the results shown in Figure 3.8 and Tab. 3.2, we clearly see that the multi-head self-attention view fusion module is crucial for both the global quality and the local geometric details. On the other hand, the local 3D context encoding is not sufficient by itself but when combined with the view fusion module helps the global reconstruction quality and avoids holes or missing parts.

To evaluate the scalability of our method, we compare reconstructions with different numbers of input views. It demonstrates that adding views effectively decreases depth ambiguities and occlusions with a clear improvement in the reconstructions. Visual results in Figure 3.9 show that the global quality of the shape (noise and missing parts) as well as geometric details (face and skirt) are improved as more views are used. It also shows the superiority of our proposed method compared to the baseline. Visual results are confirmed by the quantitative evaluation in Tab. 3.3. In particular, we observe a stronger improvement when using 4 views instead of 2 compared to 6 views instead of 4. This observation seems reasonable since the views used here are distributed evenly around the person and 4 views are sufficient to observe every side.

Variants	**CD (cm) ↓**		**Occ L1 ↓**		**Norm Cosine ↓**		**Norm L2 ↓**	
	mean	median	mean	median	mean	median	mean	median
w./o. fusion	0.553	0.478	2.013	1.755	0.101	0.093	0.373	0.353
w./o. context	0.413	0.363	1.622	1.399	0.091	0.087	0.353	0.342
Ours full	**0.367**	**0.316**	**1.538**	**1.323**	**0.089**	**0.083**	**0.350**	**0.337**

Table 3.2: Ablation studies on the effectiveness of different components. We evaluate our method when deactivating the view fusion module and the local 3d context encoding, respectively. Best scores are in **bold.**

Variants	**CD (cm) ↓**		**Occ L1 ↓**		**Norm Cosine ↓**		**Norm L2 ↓**	
	mean	median	mean	median	mean	median	mean	median
PIFu 2 v.	1.386	1.233	3.206	2.861	0.136	0.130	0.444	0.432
PIFu 4 v.	0.592	0.510	2.079	1.773	0.103	0.093	0.376	0.358
PIFu 6 v.	0.331	0.313	1.499	1.402	0.088	0.083	0.345	0.331
Ours 2 v.	0.870	0.753	2.909	2.474	0.121	0.114	0.407	0.392
Ours 4 v.	0.367	0.316	1.538	1.323	0.089	0.083	0.350	0.337
Ours 6 v.	**0.279**	**0.245**	**1.383**	**1.215**	**0.082**	**0.079**	**0.337**	**0.327**

Table 3.3: Ablation studies on using a different number of views as input. Best scores are in **bold.**

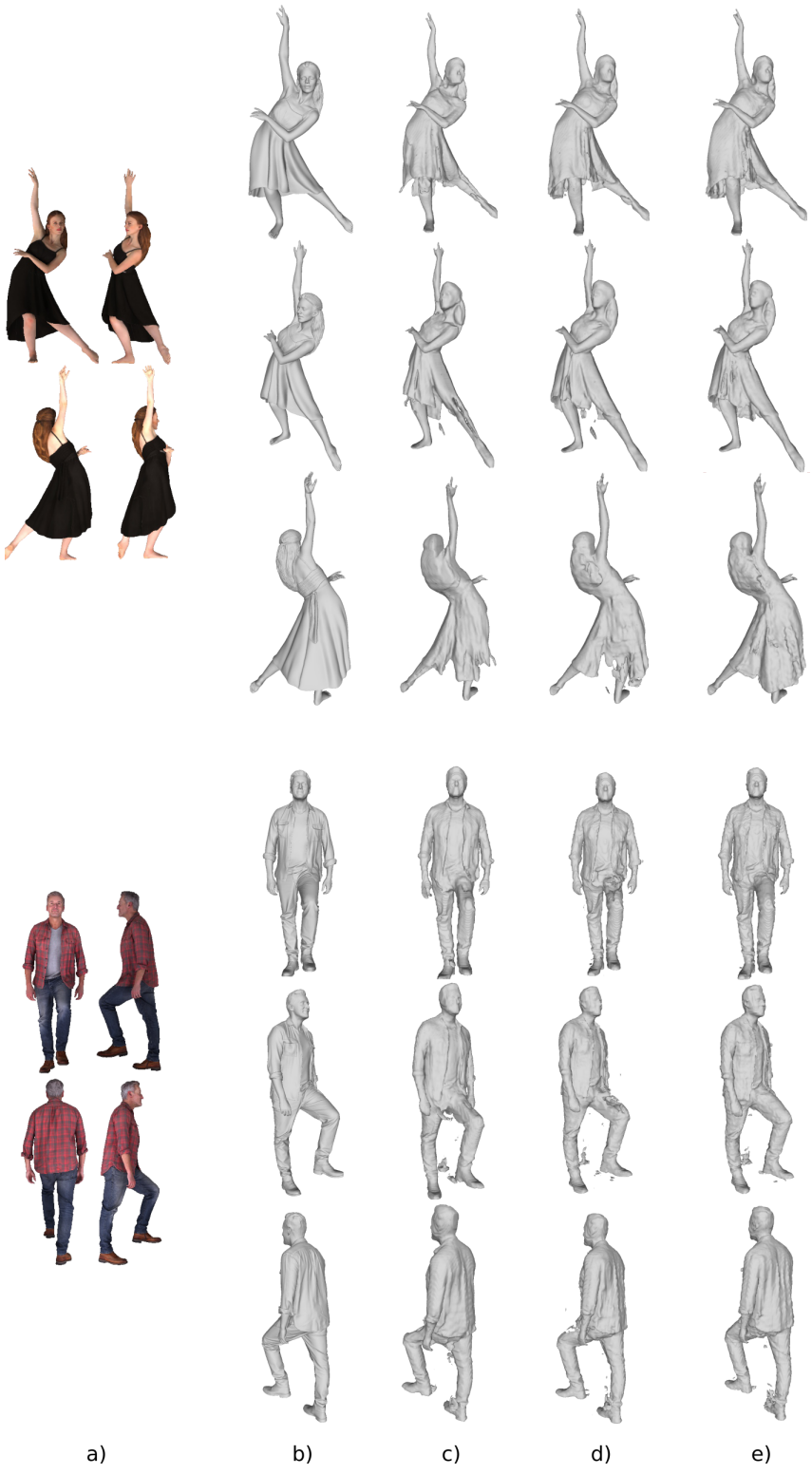


Figure 3.8: Ablation studies of our approach: a) Input cropped images. The 4 input images are rendered with the rotations around the vertical axis : 10°, 110°, 150°, 300°. b) Ground truth models. c) Ours without the attention-based view fusion module. d) Ours without the local 3D context encoding. e) Our full method.

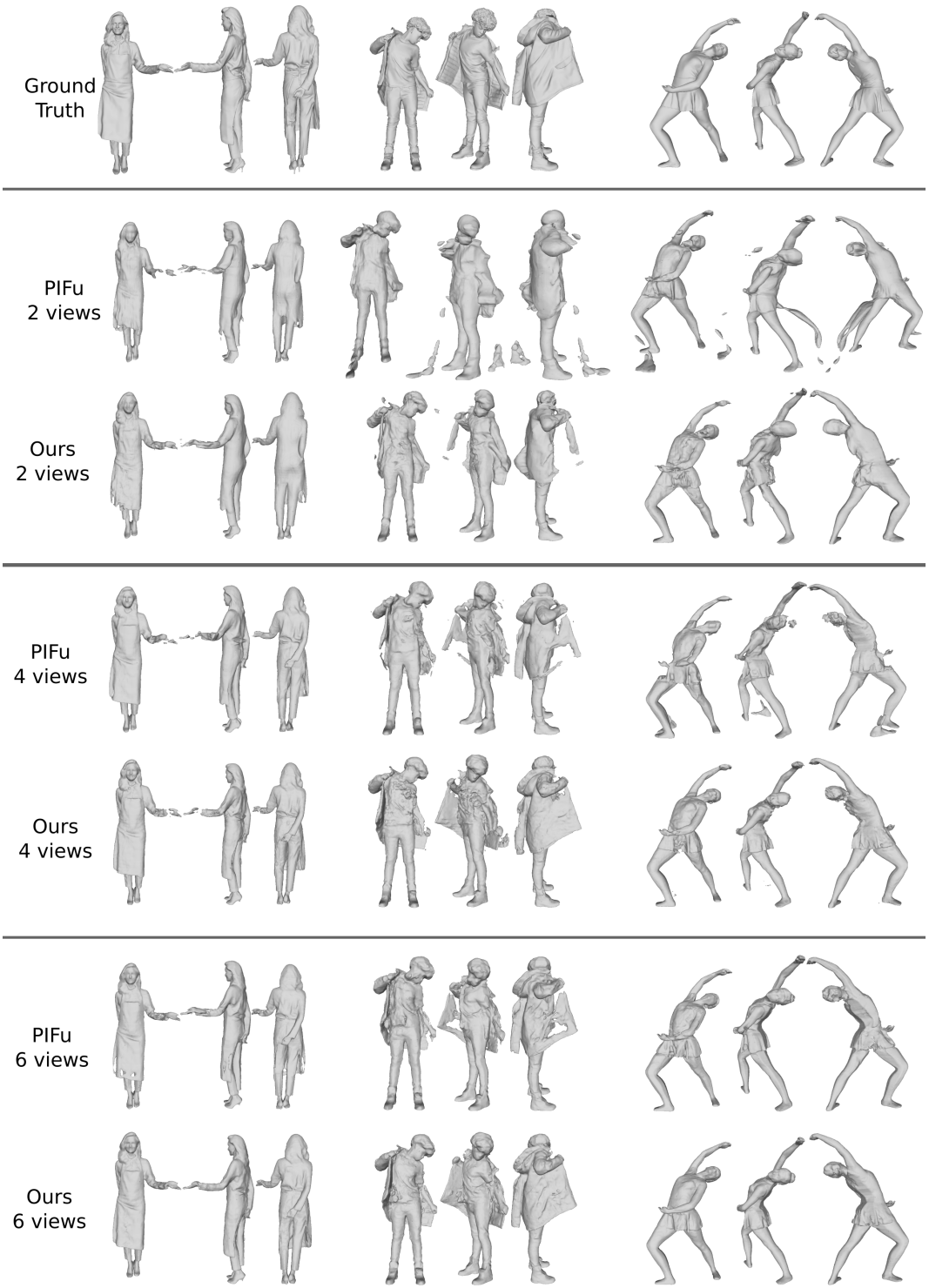


Figure 3.9: Ablation on different number of input views. As more views are added, the reconstruction with our method are improved. We also show the reconstruction results of PIFu [162]

3.4.4 Spatially Consistent Reconstruction

To demonstrate the spatial consistency of the reconstructions we consider two scenarios, using data from Renderpeople [2]. First, we apply our method

to dynamic input, namely to four synchronized video sequences showing a human walking in a scene. We reconstruct the sequence frame-by-frame, and Figure 3.10(a) shows that the reconstructions contain details (ears, clothing wrinkles) and are spatially consistent with the ground truth.

As a second scenario, we consider a static scene containing multiple persons at different positions and render high resolution images with 4 cameras. Note that this evaluation focuses on spatially consistent reconstructions and not occlusions between persons. Hence, we render each person individually while the other persons are hidden. Figure 3.10(b) shows that the reconstructions are spatially consistent with the ground truth and we can also note that the heights of the persons are correctly reconstructed.

3.4.5 Application to Real-world Data

To demonstrate the generalization of our method, we show 3D reconstructions of clothed humans with real images obtained with a 60 camera multi-view capture system. We compare with PIFu and PIFuHD when reconstructing with the front view only, to PIFu when reconstructing with 4 views, and to a multi-view stereo method [103] on the same scenes but with 60 images. For all methods to be applicable, we consider the person centered in the middle of the scene. It is important to note that the networks were trained purely on synthetic data while tested on images from a real acquisition scenario. Figure 3.11 shows that single view reconstructions suffer from an inherent depth ambiguity: some parts are missing (hair and backpack) and the pose is incorrect. Our method performs better than PIFu when 4 views are available, with more realistic global shapes and more detailed local geometries. More importantly, the comparisons with the multi-view stereo method applied to 60 images demonstrate the potential of data-driven strategies in the multi-view reconstruction domain.

3.4.6 Additional Experiments

Human Center Localization

As shown in Table 3.4, the average Euclidean distance between the ground truth and triangulated 3D human center is around 4.4cm. We compute these metrics on test data (360 groups of 4 views for 50 persons) and follows the strategy explained in Section 3.3.3 to select the 4 views. Additionally, we show in Figure 3.12 an example of reconstruction with manually specified errors on the human center location. We see that the reconstruction quality is not affected too much up to 5 cm. Noise starts being visible with an error of 10 cm and the reconstruction fails with larger error like 20 cm.

Attention scores

We provide in Figure 3.13 a visualization of the attention scores of our view fusion module. We use 4 input views, evaluate our deep neural network in a 3D grid of resolution 256 and save the attention score of the first self-attention layer. Note that we use a single head for this experiment. Points that are

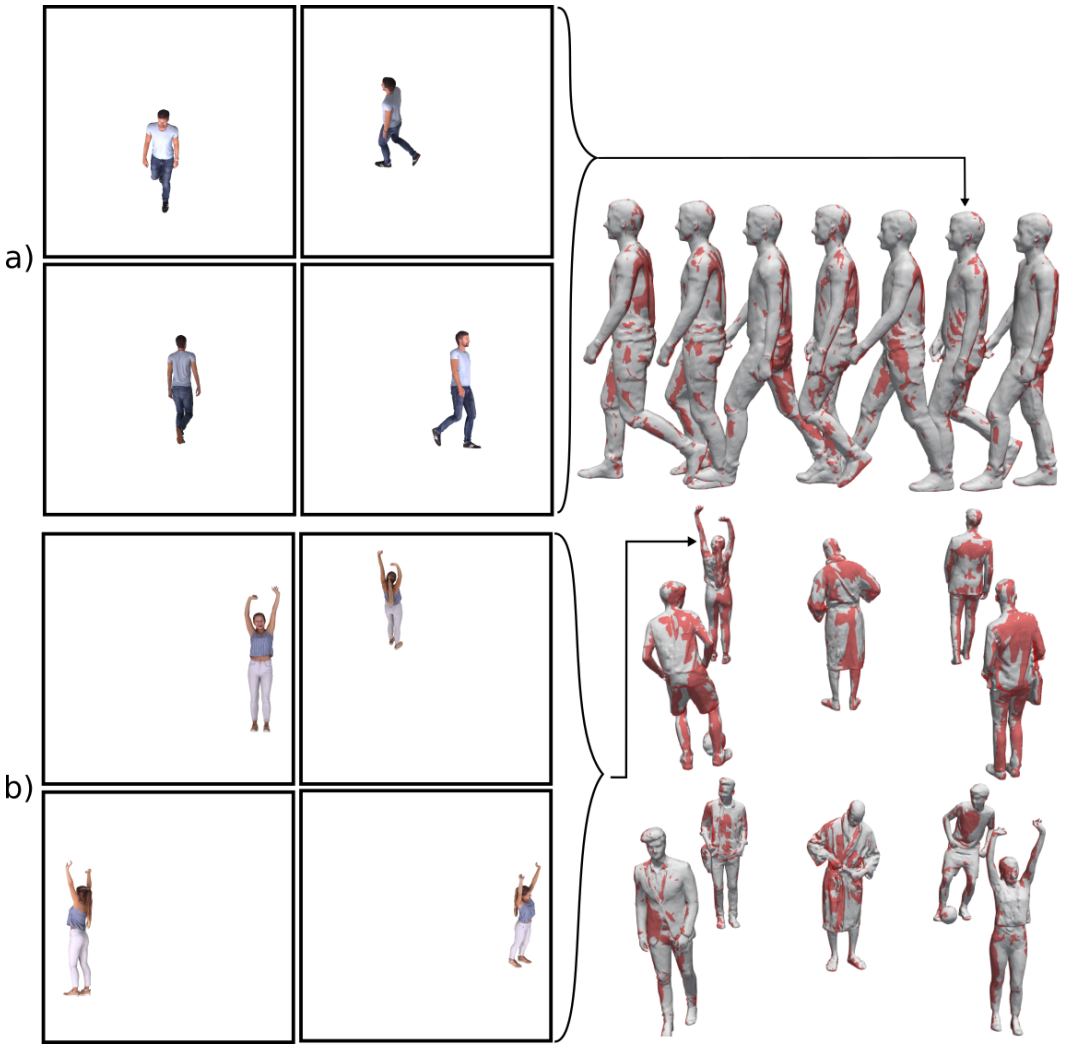


Figure 3.10: Spatially consistent reconstructions. a) Frame-by-frame reconstruction of a sequence from synchronized videos. Left: high resolution images for one example frame. Right: our result with the ground truth superimposed in red. b) Person-by-person reconstruction of a scene with multiple people. Left: high resolution images for one example person. Right: our result with the ground truth superimposed in red. a) and b) For both, the camera rotations around the vertical axis are 10°, 110°, 200°and 300°with a random elevation angle between 0°and 40°.

predicted close to the surface inside or outside are visualized and the intensity of the red channel represents how much the considered view contributed for each point. We clearly see that each point attend more to views in which they are visible.

Encoders

In our work, 2D features are extracted using the Stacked Hourglass encoder [138] that stacks multiple pooling and up-sampling networks. It allows the extraction of information at multiple scales and accounts therefore for both

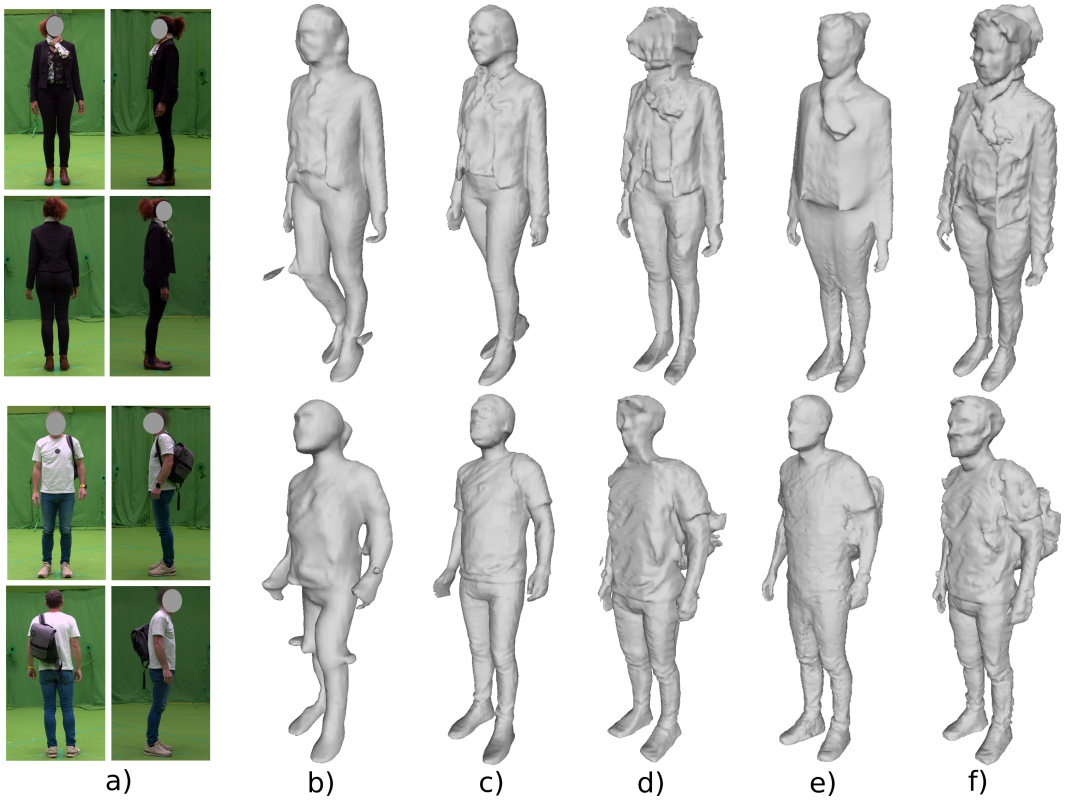


Figure 3.11: a) Real scene cropped images. b) PIFu [162] and c) PIFuHD [163] with a single frontal view. d) PIFu with 4 views. e) Multi-view stereo [103] reconstruction with 60 views. f) Our method with 4 views.

L2 - 2D (pixels)		**L2 - 3D (cm)**	
mean	median	mean	median
9.795	8.944	4.398	4.291

Table 3.4: Evaluation of the human center detection on images and the 3D triangulated position of the center. Both are evaluated on test images.

local and global contexts. Intermediate supervision is also applied to the output of each module while training our network. Of course numerous alternative encoders exist and could be used in our architecture in place of the Stacked Hourglass encoder. We provide in this section a comparison with 2 popular options: U-Net [161] and HRNet [180]. Results are shown in Figure 3.14 and in Table 3.5. The U-Net [161], a fully convolutional network based on a contractive and an expansive part, gives results which are visually close to those obtained with the Stacked Hourglass encoder, with however significantly more noise as confirmed by the metrics in Table 3.5. On the other hand, the more recent work HRNet [180] fails to provide similar results in this context.

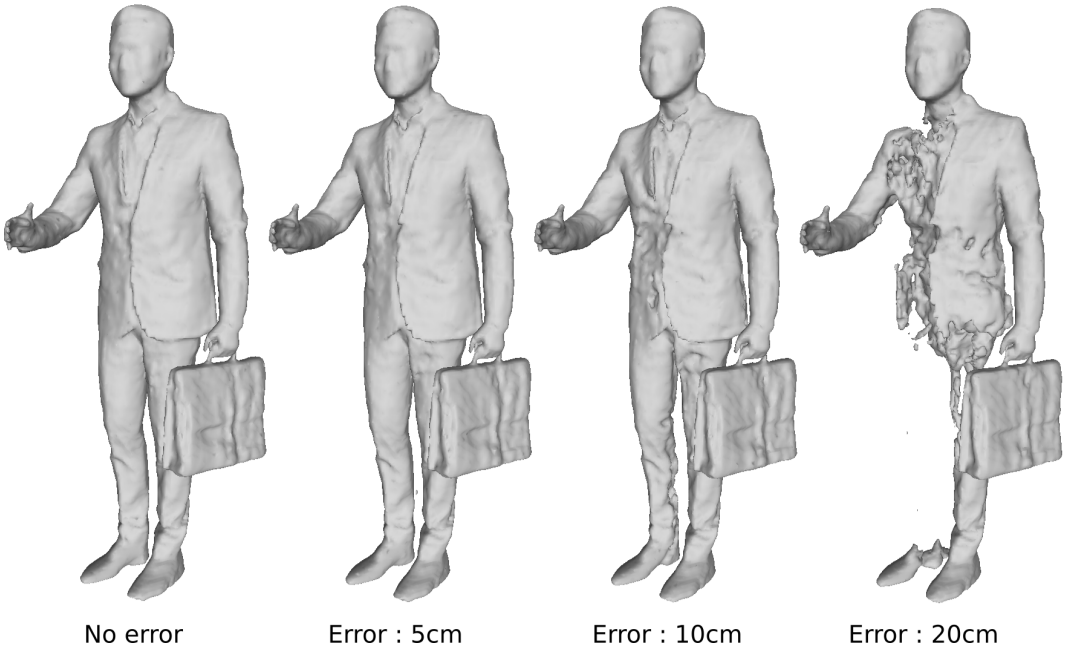


Figure 3.12: Reconstructions from 4 views that show the impact of a 3D human center localization error on the reconstruction.

Encoders	**CD (cm) ↓**		**OCC L1 ↓**		**Norm Cosine ↓**		**Norm L2 ↓**	
	mean	median	mean	median	mean	median	mean	median
SHG	**0.385**	**0.322**	**1.602**	**1.380**	**0.087**	**0.081**	**0.343**	**0.326**
U-Net [161]	0.572	0.482	1.984	1.688	0.108	0.101	0.389	0.369
HRNet [180]	1.092	1.075	3.682	3.547	0.181	0.178	0.565	0.553

Table 3.5: Quantitative results obtained by our approach, on Renderpeople data [2], with 3 different image encoders (see text in Section 3.4.6 for comments). Best scores are in bold.

Local grid size

A key point of our method is the encoding of the local context of each sampled 3D point. To this purpose, we use a local 3D grid around each sampled point and in the pipeline, each original sampled point is associated with the additional points from their 3D local neighbourhood. At each training iteration, the local grids are aligned randomly with one of the camera used and the grid size is constant and defined before training.

Here we provide the results obtained with different grid sizes defined in world coordinates: small (2 cm), medium (10 cm) and large (20 cm) grids.

Table 3.6 shows that the best results were obtained with the medium-sized local grid, which is the one that was used for the other results in this chapter. This result is confirmed visually on Figure 3.15, where the medium grid shows better reconstructions with more details and less noise. This experiment demonstrates that the size of the local grid is important as it defines

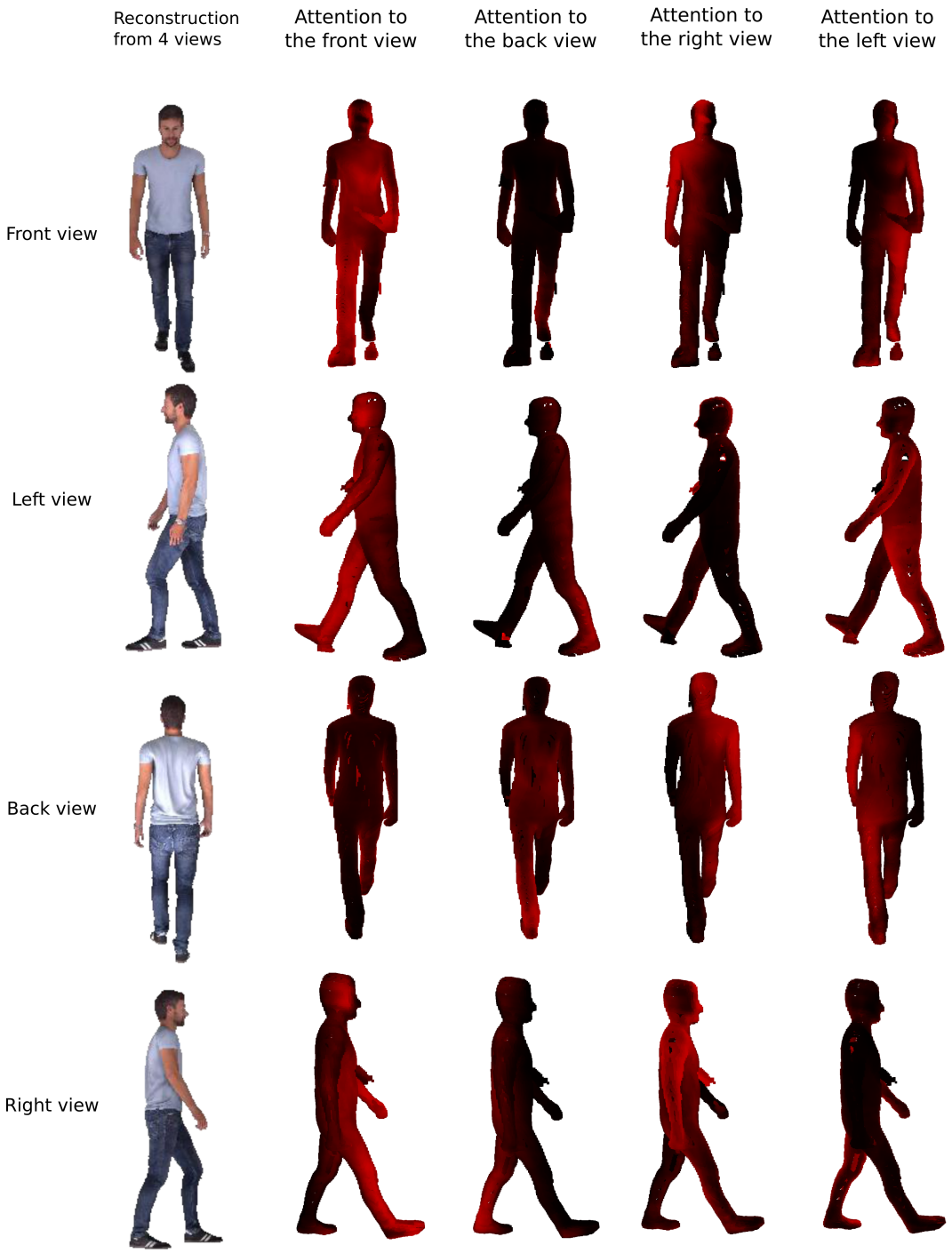


Figure 3.13: Attention scores: points predicted as close to the surface (in or out) are visualized. The intensity of the red channel represents the contribution of the considered view.

the neighbourhood considered to predict the occupancy probability of the grid center. With a small grid, all grid points tend to be projected on the same 2D feature which prevents the 3D context to be encoded. On the other hand, with large grids, points can be far from each other, even on different body parts. In that case, the neighbourhood considered is too large and not informative

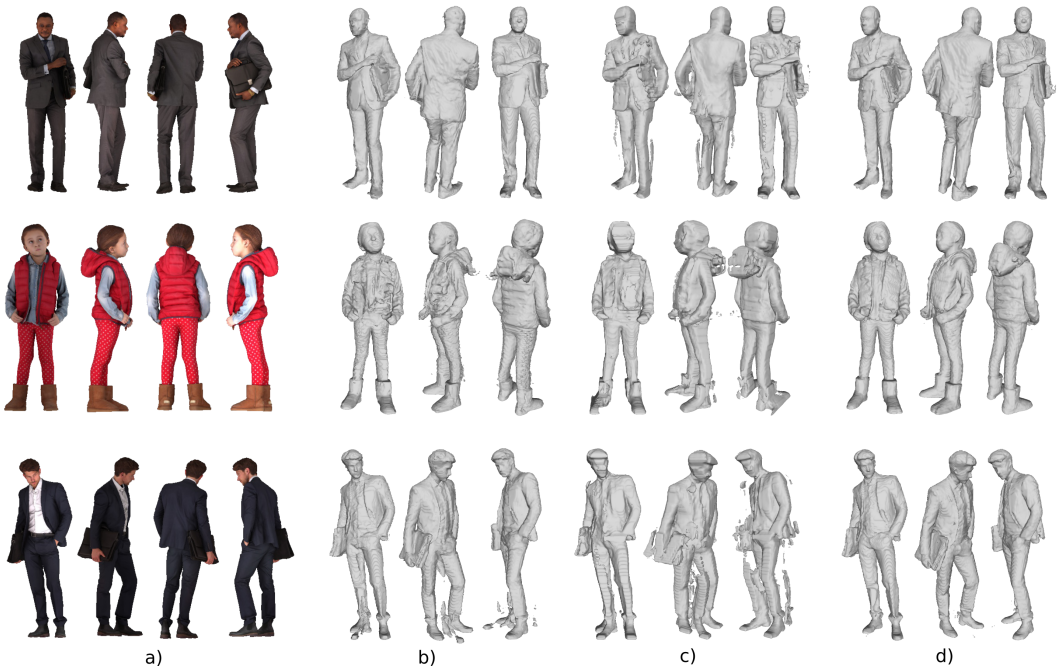


Figure 3.14: Comparative reconstruction results with our approach applied using 3 different image encoders. a) Input RGB images. b) U-Net [161]. c) HRNet [180]. d) Stacked Hourglass [138].

when predicting occupancies.

Grid size	**CD (cm) ↓**		**OCC L1 ↓**		**Norm Cosine ↓**		**Norm L2 ↓**	
	mean	median	mean	median	mean	median	mean	median
small (2 cm)	0.422	0.413	1.668	1.566	0.089	0.087	**0.342**	0.336
medium (10 cm)	**0.385**	**0.322**	**1.602**	**1.380**	**0.087**	**0.081**	0.343	**0.326**
large (20 cm)	0.441	0.421	1.677	1.592	0.091	0.089	0.350	0.341

Table 3.6: Quantitative results and comparisons with 3 local grid sizes on Renderpeople data [2]. Best scores are in bold.

3.5 Conclusion

In this chapter, we build on recent progress on implicit representations of 3D data and propose a method for 3D reconstruction of clothed humans from a few sparse views. We introduce three key components: 1) a spatially consistent reconstruction that allows for arbitrary placement of the person in the input views using a perspective camera mode; 2) a fusion layer based on an attention mechanism that learns to efficiently combine the information from all available views; 3) a mechanism that encodes local 3D patterns in the multi-view context. Our experiments show that the proposed method outperforms the state of the art in terms of details and global quality of the reconstructions on

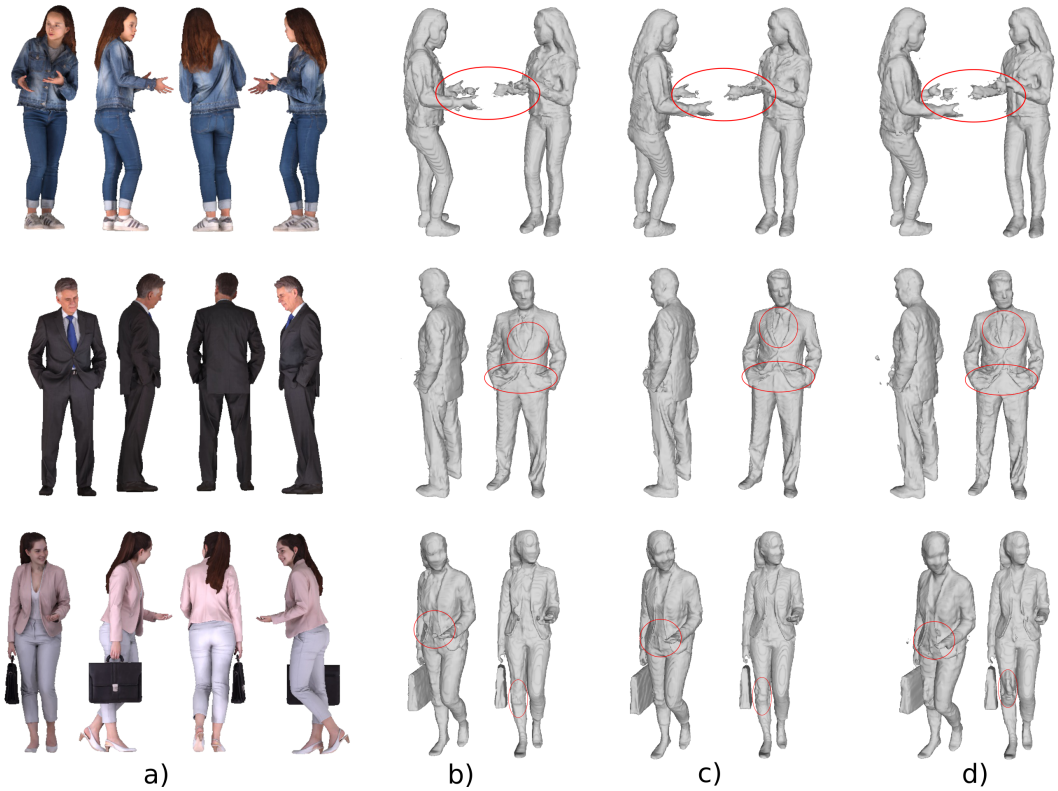


Figure 3.15: Qualitative comparisons of the reconstructions using our method with 3 different local grid sizes. a) Input RGB images. b) Small grid (2 cm). c) Medium grid (10 cm). d) Large grid (20cm).

synthetic data. We also demonstrate a better generalization of our method on real data acquired with a multi-view platform. Additionally, we show that our approach can even approximate multi-view stereo results with dramatically fewer views.

4

Multi-View Reconstruction Using Signed Ray Distance Functions (SRDF)

4.1 Introduction

In this chapter we consider the 3D reconstruction problem in a more traditional scenario with dense input viewpoints. This problem of reconstructing 3D shape geometries from 2D image observations has been a core issue in computer vision for decades. Applications are numerous and range from robotics to augmented reality and human digitization, among others. The problem can be decomposed into two parts, the 3D shape geometry and the shape appearance estimations. While the second part is necessary to generate images from novel viewpoints, the first appears more crucial in most applications and is often a preliminary step to the appearance estimation. We focus on the shape geometry estimation in this chapter. When images are available in sufficient numbers, multi-view stereo (MVS) is a powerful strategy that has emerged in the late 90s (see [170]). In this strategy, 3D geometric models are built by searching for surface locations in 3D where 2D image observations concur, a property called photo-consistency. This observation consistency strategy has been later challenged by approaches in the field that seek instead for observation fidelity using differentiable rendering. Given a shape model that includes appearance information, rendered images can be compared to observed images and the model can thus be optimized. Differentiable rendering adapts to several shape representations including point clouds, meshes and, more recently, implicit shape representations. The latter can account for occupancy, distance functions or densities, which are estimated either directly over discrete grids or through continuous MLP network functions. Associated to differentiable rendering these implicit representations have provided state-of-the-art approaches to recover both the geometry and the appearance of 3D shapes from 2D images.

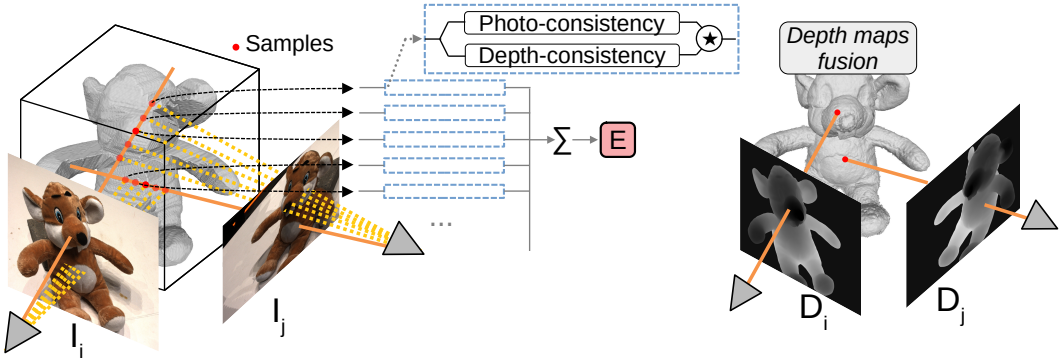


Figure 4.1: Our method overview. Left: Given multiple RGB images and an initial coarse reconstruction, our method optimizes depth maps using a volumetric shape energy E that is evaluated at samples along viewing lines. Right: The optimized depth maps are further fused into a surface model.

With the objective to improve the precision of the reconstructed geometric models and their computational costs, we investigate an approach that takes inspiration from differentiable rendering methods while retaining beneficial aspects of MVS strategies. Following volumetric methods we use a volumetric signed ray distance representation which we parameterize with depths along viewing rays, a representation we call the Signed Ray Distance Function or SRDF. This representation makes the shape surface explicit with depths while keeping the benefit of better distributed gradients with a volumetric discretization. To optimize this shape representation we introduce an unsupervised differentiable volumetric criterion that, in contrast to differentiable rendering approaches, does not require color estimation. Instead, the criterion considers volumetric 3D samples and evaluates whether the signed distances along rays agree at a sample when it is photo-consistent and disagree otherwise. While being volumetric our proposed approach shares the following MVS benefits:

i) No expensive ray tracing in addition to color decisions is required;
ii) The proposed approach is pixel-wise accurate by construction;
iii) The optimization can be performed over groups of cameras defined with visibility considerations. The latter enables parallelism between groups while still enforcing consistency over depth maps.

To evaluate the approach, we conducted experiments on real data from DTU Robot Image Data Sets [82], BlendedMVS [216] and on synthetic data from Renderpeople [2] as well as on real human capture data. Ablation tests demonstrate the respective contributions of the SRDF parametrization and the volumetric integration in the shape reconstruction process. Comparisons with both MVS and Differential Rendering methods also show that our method consistently outperforms state of the art both quantitatively and qualitatively with better geometric details.

4.2 Method

Our method takes as input N calibrated color images $\mathcal{I} = I_{j\in[1,N]}$ and assumes N initial associated depth maps $\mathcal{D} = D_{j\in[1,N]}$ that can be obtained using an initial coarse reconstruction, with for instance pre-segmented image silhouettes as in Figure 4.1. It optimizes depth values along pixel viewing lines by considering a photo-consistency criterion that is evaluated in 3D over an implicit volumetric shape representation. Final shape surfaces are thus obtained by fusing depth maps, as in *e.g.* [33, 54]. The main features of the method are:

- Shape representation (Sec 4.2.1): depth maps determine the signed distances, along pixel viewing rays, that define our volumetric shape representation with the SRDF. Parameterizing with depths offers several advantages: it better accounts for the geometric context by materializing the shape surface; it enables pixel accuracy regardless of the image resolution; it allows for coarse to fine strategies as well as parallelization with groups of views.

- Energy function (Sec 4.2.2): our shape energy function is evaluated at sample locations along viewing lines and involves multiple depth maps simultaneously, therefore enforcing spatial consistency. It focuses on the geometry and avoids potential ambiguous estimation of the appearance.

- Photometric prior (Sec 4.2.3): The photo-consistency hypothesis evaluated by the energy function along a viewing line can be diverse. We propose a criterion that is learned over ground truth 3D data such as DTU [82]. We also experiment a baseline unsupervised criterion that builds on the median color.

4.2.1 Signed Ray Distance Function

Our shape representation is a volumetric signed distance function parameterized by depths along viewing rays. This is inspired by signed distance functions (SDF) and shares some similarities with more recent works on signed directional distance functions (SDDF) [239]. Unlike traditional surface-based representations such a function is differentiable at any point in the 3D observation volume.

Instead of considering the shortest distances along any direction as in standard SDF, or in a fixed direction as in SDDF [239], we define, for a given 3D point X, its N signed distances with respect to cameras $j \in [1, N]$ as the signed distances of X to its nearest neighbour on the surface as predicted by camera j along the viewing ray passing through X. We denote the distance for X and camera j by the *Signed Ray Distance Function (SRDF)*, as illustrated in Figure 4.2:

$$SRDF(X, D_j) = SRDF_j(X) = D_j(X) - Z_j(X), \tag{4.1}$$

where $D_j(X)$ is the depth in depth map D_j at the projection of X and $Z_j(X)$ the distance from X to camera j.

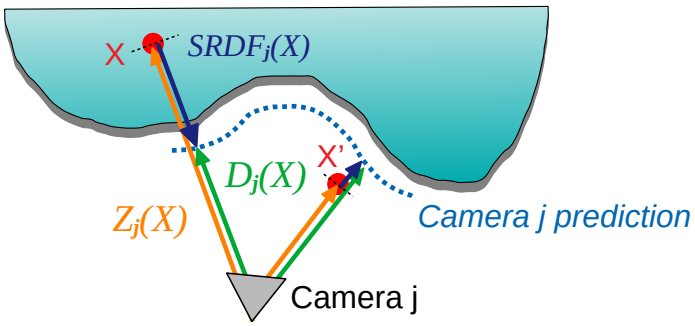


Figure 4.2: For any 3D point X, its ray signed distance $SRDF_j(X) = D_j(X) - Z_j(X)$ with respect to camera j is the signed shortest distance from X to the surface, as predicted by camera j along the corresponding viewing line.

4.2.2 Volumetric Shape Energy

The intuition behind our volumetric energy function is that photometric observations across different views should be consistent on the surface and not elsewhere. Importantly such a behavior should be shared by the SRDF predictions across views that should also consistently identify zero distances for points on the surface and non-consistent distances elsewhere. Given this principle, illustrated in Figure 4.3, a computational strategy is to look at the correlation between these 2 signals, the observed photo-consistency and the predicted SRDF consistencies, and to try to maximise it at 3D sample locations $\{X\}$ in the observation space (see Figure 4.4). To this purpose we introduce the following consistency energy function:

$$E(\{X\},\mathcal{D},\mathcal{I}) = \sum_X C_{SRDF}(X,\mathcal{D}) \;\; C_\Phi(X,\mathcal{I}), \tag{4.2}$$

where $\{X\}$ are the 3D sample locations, $C_{SRDF}(X,\mathcal{D})$ and $C_\Phi(X,\mathcal{I})$ represent measurements of consistency among the predicted SRDFs values $SRDF_{j\in[1,N]}(X)$ and among the observed photometric observations $\Phi_{j\in[1,N]}(X)$, respectively, at location X. Both are functions that return values between 0 and 1 that characterize consistency at X. We detail below the SRDF consistency measure $C_{SRDF}(X)$. The photo-consistency measure $C_\Phi(X)$ is discussed in Section 4.2.3. The above energy E is differentiable with respect to the predicted depths values $\mathcal{D}$ and computed in practice at several sample locations along each viewing ray of each camera, which enforces SRDFs to consistently predict surface points over all cameras.

SRDF consistency From the observation that SRDF consistency is only achieved when X is on the surface, *i.e.*when $SRDF_j(X) = 0$ for all non occluded cameras j, we define:

$$C_{SRDF}(X) = \prod_{j=1}^{N} \Big(exp\Big(- \frac{SRDF_j(X)^2}{\sigma_d} \Big) + \Gamma_{SRDF} \Big), \tag{4.3}$$

where the ray signed distances are transformed into probabilities using an exponential which is maximal when $SRDF_j(X) = 0$. Γ_{SRDF} is a constant that

prevents the product over all cameras to cancel out in case of inconsistencies caused by camera occlusions. It can be interpreted as the probability of the $SRDF_j$ value at X knowing camera j is occluded, which can be set as constant for all values. σ_d is a hyperparameter that controls how fast probabilities decrease with distances to the surface. It should be noted here that the above energy term C_{SDRF} is a product over views at a 3D point X and not a sum, hence gradients w.r.t. depth values are not independent at X, which forces distances to become consistent across views.

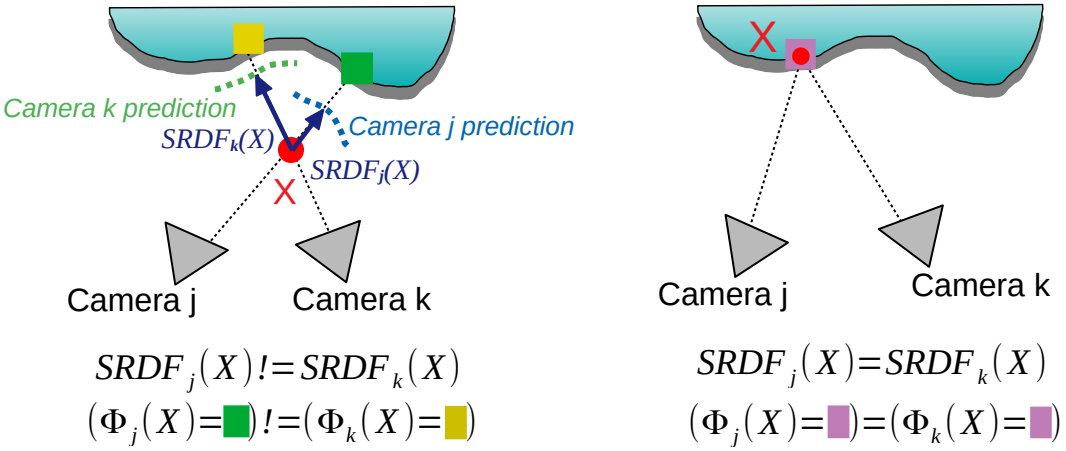


Figure 4.3: Inconsistency (left) and consistency (right) of the ray signed distances $SRDF_{j,k}(X)$ and of the photometric information $\Phi_{j,k}(X)$ at X with respect to cameras j and k.

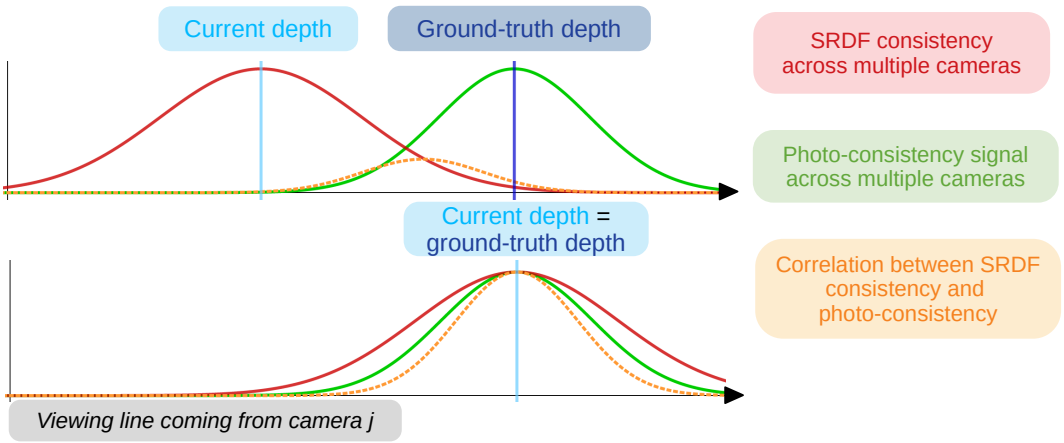


Figure 4.4: The SRDF consistency (red) and photo-consistency signals (green) along a viewing line. Their cross correlation will be maximal when the current predicted depth aligns with the ground truth depth.

4.2.3 Photometric Consistency

Our model is agnostic to the photo-consistency measure $C_\Phi(X)$ that is chosen. In practice, we have considered 2 instances of $C_\Phi(X)$: A baseline version that

relies on the traditional Lambertian prior assumption for the observed surface and a learned version that can be trained with ground truth 3D data.

Baseline Prior Our baseline prior assumes a Lambertian surface and therefore similar photometric observations for points on the observed surface for all non-occluded viewpoints. While ignoring non-diffuse surface reflections the assumption has been widely used in image based 3D modelling, especially by MVS strategies. The associated consistency measure we propose accounts for the distance to the median observed value. Under the Lambertian assumption all observed appearances from non-occluded viewpoints should be equal. Assuming further that occluded viewpoints are fewer we define the photo-consistency as:

$$C_{\Phi}(X) = \prod_{j=1}^{N} \left(exp\left(- \frac{\|\Phi_j(X) - \widetilde{\Phi}(X,\mathcal{I}))\|^2}{\sigma_c} \right) + \Gamma_{\Phi} \right), \tag{4.4}$$

where $\Phi_j(X)$ is photometric observation of X in image j, typically an RGB color, and $\widetilde{\Phi}(X,\mathcal{I}))$ is the median value of the observations at X over all images. Similarly to equation 4.3, Γ_{Φ} is a constant that prevents the product over all cameras to cancel out in case of occlusion and σ_c is a hyperparameter. As shown in Section 4.4 this baseline photometric prior yields state-of-the-art results on synthetic 3D data for which the Lambertian assumption holds but is less successful on real data.

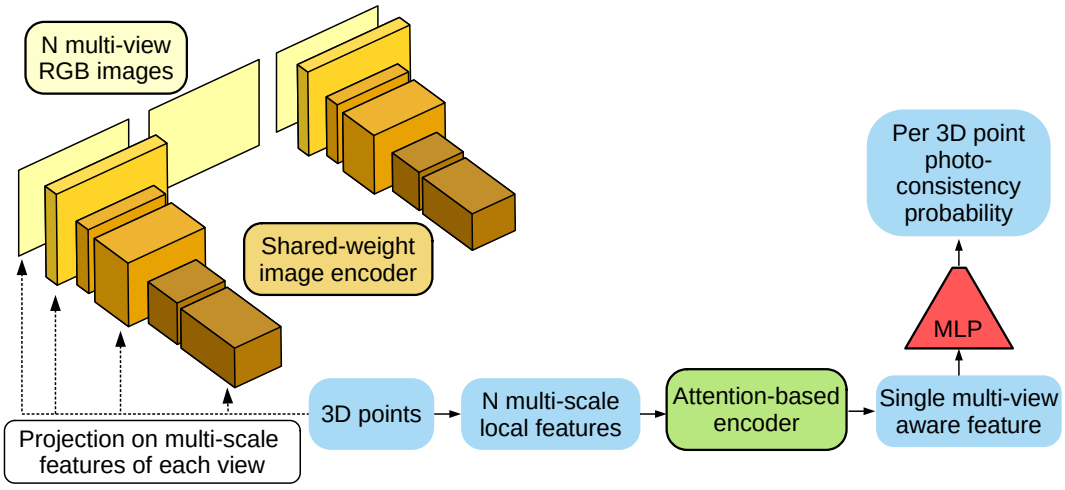


Figure 4.5: Proposed architecture to learn the photo-consistency.

Learned Prior In order to better handle real images that are noisy and for which the Lambertian assumption is partially or not satisfied, we have experimented a more elaborated photo-consistency measure with a data driven approach. Inspired by previous works [66, 104], we cast the problem as a classification task between points that are photo-consistent across multiple views and points that are not and train a network for that purpose.

As described in Figure 4.5, the network architecture tries to match the local appearance of a 3D point in different views and outputs a photo-consistency score between 0 and 1. This module is independent of the number of cameras and provides very good results on real data and generalization abilities as demonstrated in Section 4.4.

4.3 Implementation

In the following, we first provide more details about our optimization pipeline. Then we describe the architecture of our photo-consistency network and finally specify the hyperparameters we use in the experiments.

4.3.1 Optimization Pipeline

To allow for efficient processing, we define G groups of cameras, which can be optimized in parallel. Since our approach optimizes geometry based on appearance matching, it is preferable to minimize occlusions. For this reason, we heuristically choose to gather cameras that are close to each other.

For the depth maps associated to a camera group, at each epoch, we iterate over all rays r_j^i corresponding to foreground pixels i of cameras j, as defined by pre-segmented silhouettes, and sample points along r_j^i around the current depth estimation d_j^i. This sampling is parameterized by two parameters: an offset o that defines an interval for the sampling around the current depth $[d_j^i - o; d_j^i + o]$, and the density of the sampling which represents the number of points that we sample uniformly in that interval. Ideally, the real depth $\hat{d}_j^i$ should be contained inside the interval $[d_j^i - o; d_j^i + o]$, otherwise it is difficult for the appearance to guide the geometry optimization. From that observation, we define a coarse-to-fine strategy for the sampling. With the aim to capture the ground truth depth in the interval, we start with a large interval that is gradually reduced. The sampling density can be adjusted in the same way but decreasing the size of the sampling interval already indirectly increases its density, so in practice we keep the sampling density constant.

Our shape energy, described in subsection 4.2.2, is computed over all the samples from all the rays of each camera. The gradients are computed using Pytorch autodiff [147] and back-propagated to update depth maps.

4.3.2 Photo-consistency Network

Architecture

As explained in section 4.2.3, we propose a data-driven photo-consistency measure to better handle real images that are noisy and for which the Lambertian assumption is partially or not satisfied. This network is composed of 3 main parts. First, features are extracted from the input images by an image encoder composed of convolutional layers, batch normalizations, ReLu activations and max-pooling operations, as shown in Figure 4.6. Given an input 3D point, its per view multi-scale features are obtained by projecting it in the

multi-scale feature maps extracted with the image encoder and by concatenating over scales. Next, we use a self-attention module [191] to combine the multi-scale features from all views and obtain therefore a multi-scale/multi-view (MSV) feature. This Pytorch [148] module is parameterized as follows, $d\_model = 115$, $nhead = 1$, $dim\_feedforward = 256$, $num\_layers = 6$. Note that we also apply a Mean operation on the output of this self-attention module. Finally, a fully connected network decodes the MSV feature and outputs a photo-consistency score between 0 and 1, as shown in Figure 4.7. To train the network, we use an MSE loss between the ground truth and predicted photo-consistency scores and the Adam optimizer with a learning rate of $1e^{-4}$.

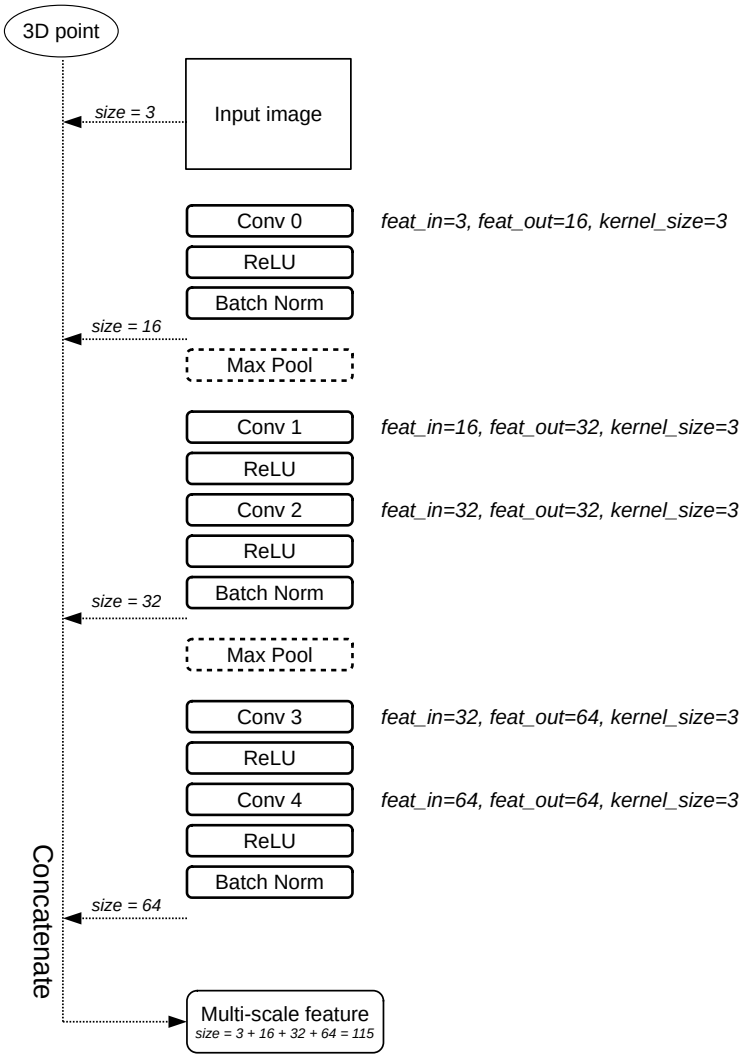


Figure 4.6: Architecture of the image encoder.

Training Strategy

To train the photo-consistency network, we use the DTU Robot Image Data Sets [82] composed of 124 scans of objects. For each scan, there are 49 or 64 images under 8 different illuminations settings, camera calibration and ground truth point cloud obtained from structured light. We select 15 test objects and

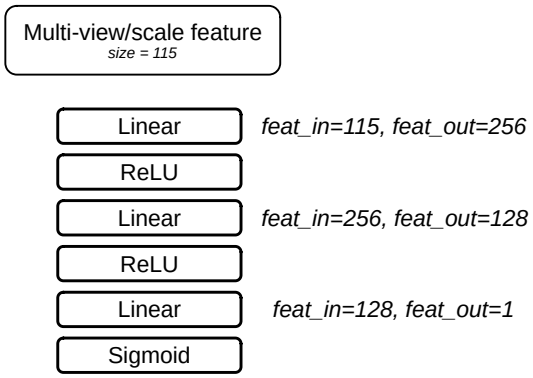


Figure 4.7: Fully connected decoder.

remove all the scans that contain these objects from the training set which results in 79 training scans. Next, we reconstruct a surface from the ground truth point cloud using the Screened Poisson algorithm [93] and surface trimming of 9.5. From the reconstructed meshes we render ground truth depth maps and use them to sample points on the surface (positive samples) and points that are either in front or behind the surface (negative samples). We make sure to keep a balanced sampling strategy with an equal number of positive and negative samples.

To encourage the network to remain invariant to the number of cameras, at each training iteration, we randomly select a subset of K cameras from the total N cameras. Matching appearances between cameras too far from each other leads to inconsistencies as a result of the potentially high number of occlusions. To remedy this, we create the camera groups using a soft nearest neighbour approach. We randomly select a first camera, compute its K' nearest neighbours cameras with $K < K' < N$, and randomly select $K-1$ cameras from them. In practice, $N = 49$ or 64 and we choose $K \in [4, 10]$ and $K' = min(2K, 15)'$.

4.3.3 Hyperparameters

In Table 4.1 we specify the hyperparameters we used in our experiments. They mostly depend on the unit of the dataset and the prior of photo-consistency that is used. The definition of each parameter is available in sections 4.2.2, 4.2.3 and 4.3.1.

4.4 Experimental Results

To assess our method we conduct an evaluation on multi-view 3D shape reconstruction. First, we introduce the existing methods that we consider as our baseline. Then, we present the datasets as well as the evaluation metrics. We provide quantitative and qualitative comparisons against the current state of the art on real images using our learned prior for photo-consistency. Then, we also show that our method combined with a baseline prior for photo-consistency provides good reconstruction results under the Lambertian sur-

	DTU	Renderpeople	Real human capture data
Unit	mm	cm	m
Photo-consistency prior	learned	baseline	learned
σ_c	0.1	0.05	0.1
σ_d	25	2.5	0.025
o	100	10	0.05
Sampling density	51	51	51
lr	1	0.1	0.001

Table 4.1: Hyperparameters used in our optimization for the different experiments.

face assumption. Finally, we demonstrate better generalization abilities of our method compared to deep MVS inference-based methods and that the latter can serve as an initialization.

4.4.1 Datasets and Metrics

To evaluate our method on real multi-view images with complex lighting, we use the 15 test objects from the DTU Data Sets [82] and BlendedMVS [216]. Note again that BlendedMVS is not used to train our learned photo-consistency prior. For DTU, the corresponding background masks are provided by Yariv et al. [217]. To test our method with the baseline prior for photo-consistency, we render multi-view images from Renderpeople [2] meshes. This dataset provides highly detailed meshes obtained from 3D scans of dressed humans and corrected by artists. We render 19 high-resolution images (2048x2048) that mostly show the frontal part of the human. For the quantitative evaluation with DTU we use a Python implementation [1] of the official evaluation procedure of DTU. The accuracy and completeness metrics, with the Chamfer distances in mm, are computed w.r.t. ground truth point clouds obtained from structured light. Finally, to evaluate generalization to novel data, we also experiment with images from a large scale hemispherical multi-view setup with 65 cameras of various focal lengths.

4.4.2 Baseline Methods

To assess our approach, we evaluate the geometry against state-of-the-art methods among 3 categories: classic MVS, deep MVS and differential rendering-based methods. First, COLMAP [167] and ACMMP [204] are classic MVS methods that have been widely used and demonstrate strong performances for MVS reconstruction. Among all the deep MVS methods, we consider two of the most efficient methods PatchmatchNet [195] and CasMVS-Net [60] for which the code is available and easy to use. Finally, for the

differentiable rendering-based methods we consider IDR [217] which was one of the first works that combines a differentiable surface renderer with a neural implicit representation. It requires accurate masks but handles specular surfaces and has shown impressive reconstruction results. We also compare with two more recent works that use volumetric rendering and provide impressive reconstruction results: NeuS [197] and NeuralWarp [34].

For the evaluation on DTU using all the available views (49 or 64 depending on the scan) we retrain PatchmatchNet and CasMVSNet as their pre-trained models use a different train/test split. We use the pre-trained models for IDR, NeuS (with the mask loss) and NeuralWarp.

To recover meshes with our method, we use a post-processing step with a bilateral filter on the optimized depth maps, a TSDF Fusion [33] method and a mesh cleaning based on the input masks. For COLMAP, ACMMP, PatchmatchNet and CasMVSNet we try to use the same TSDF Fusion method [33] as much as possible. For differentiable rendering-based methods (IDR, NeuS and NeuralWarp), the implicit representation is simply evaluated in a 3D grid of size 512^3 and then Marching Cubes [117] is applied.

4.4.3 Multi-View Reconstruction From Real Data

Qualitative results In Figure 4.8, we show comparisons between our method and the considered baselines. While IDR, NeuS and NeuralWarp produce high quality details, they show some artifacts on some misleading parts: some regions of the fruits (1st row), near the right arm of the figurine (2nd row) or at the separation between the belly and the legs of the statue (3rd row). In contrast, our method provides high level of details without failing on these difficult parts. For IDR and NeuS, the appearance prediction probably compensates for the wrong geometry during the optimization, however our approach exhibits more robustness by focusing on the geometry. Our method produces visual results comparable to COLMAP, ACMMP, PatchmatchNet and CasMVSNet with high fidelity and reduced noise. As shown in Figure 4.9 our method provides similar results with high quality details on BlendedMVS, using the same photo-consistency prior trained on DTU. Note that in the bottom line of Figure 4.9, we use a coarse initial reconstruction obtained with [195].

Quantitative results In our quantitative evaluation, all the results are computed on the meshes obtained with the differentiable rendering-based methods and directly on the point clouds fused from the depthmaps for COLMAP, ACMMP, PatchmatchNet, CasMVSNet and our method. On average, our method clearly outperforms methods based on differentiable rendering (IDR, NeuS and NeuralWarp) in terms of accuracy and completeness. Our method also demonstrates an improvement over classic MVS methods COLMAP and ACMMP. Compared to deep MVS methods that are trained end-to-end on DTU, the approach is on par, though better on combined accuracy and completeness, while training only a small neural network for photo-consistency that is used as a prior in the optimization. Note that quantitative results for PatchmatchNet and CasMVSNet are not as high as in their paper since the

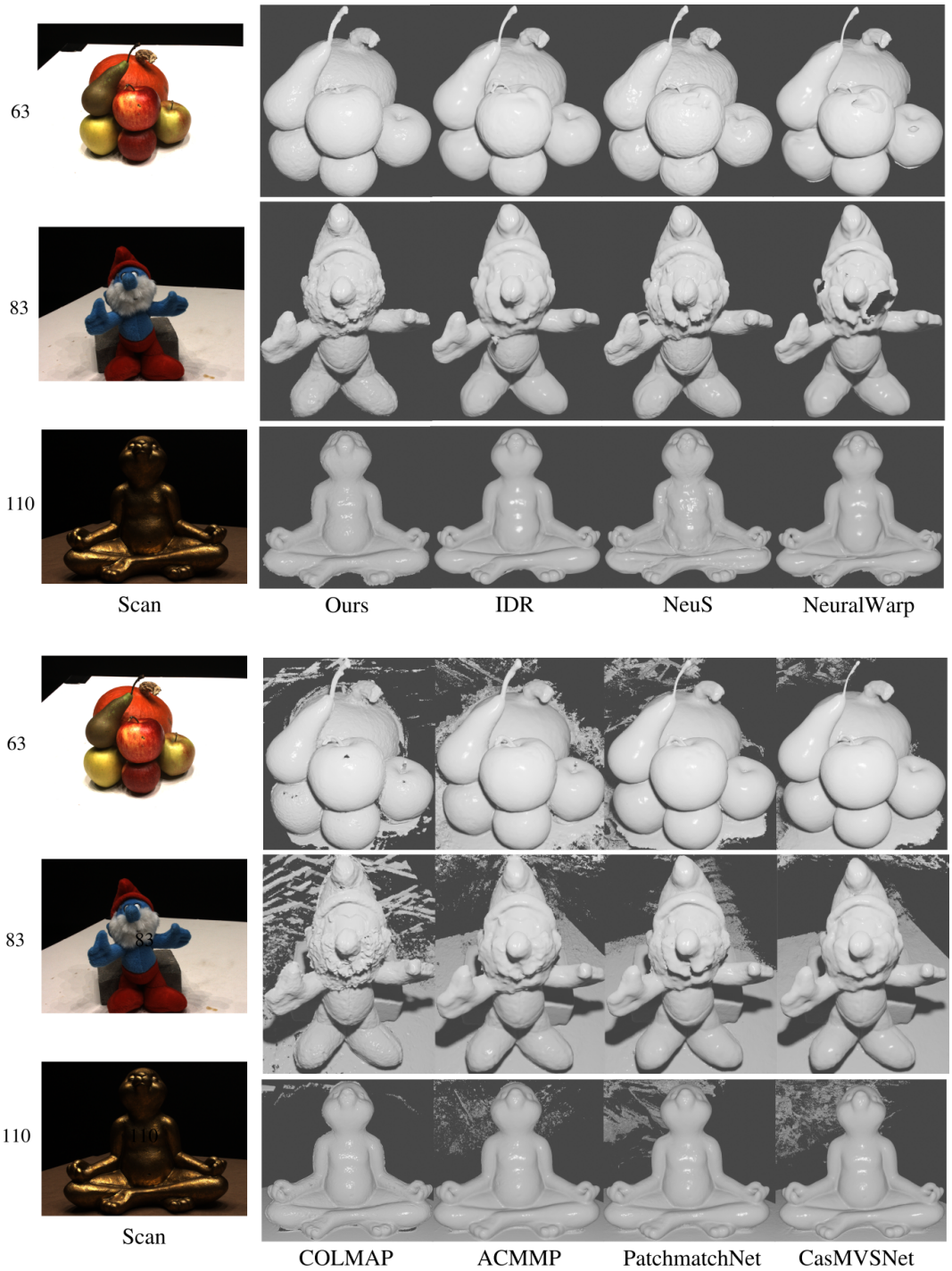


Figure 4.8: Qualitative comparisons with state-of-the-art methods.

training set is not the same and in contrast to their train/test split, we remove all the scans from the training set in which a test object is seen.

Runtime analysis In terms of runtime, our method (single machine and GPU) is competitive with the MVS methods COLMAP and ACMMP and takes around 45 minutes to optimize with 49 images from DTU. We note that it could also benefit from an easy parallelization by optimizing different

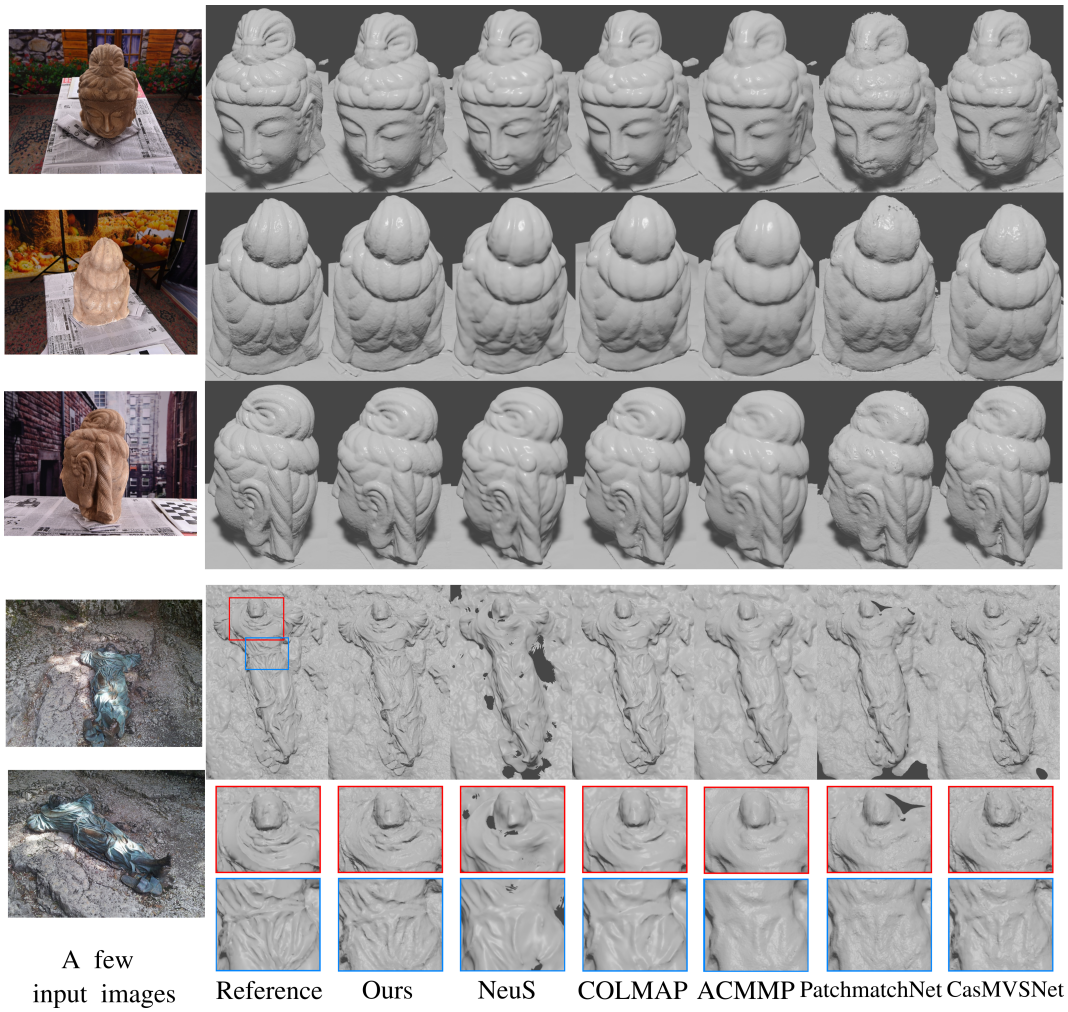


Figure 4.9: Qualitative comparisons using 36 *and* 14 *images of a model from BlendedMVS [216].*

groups of depthmaps on different GPUs or even machines which can significantly reduce the computation time. COLMAP, ACMMP and our method have a strong computation time advantage compared to methods based on differentiable rendering and neural implicit representations (IDR, NeuS and NeuralWarp) that require several hours (between 10h and 25h) before converging to an accurate 3D reconstruction. Of course, deep MVS methods like PatchmatchNet and CasMVSNet that perform only inference are inherently much faster (a few seconds or minutes) than optimization-based methods.

4.4.4 Reconstruction From Synthetic Data

To reinforce the validity of our volumetric shape energy, we experiment with the proposed baseline photo-consistency prior defined in Section 4.2.3. We use 19 synthetic images from Renderpeople [2] and compare qualitatively with classic MVS methods (COLMAP, ACMMP), differentiable rendering-based methods (IDR, NeuS) and deep MVS methods (PatchmatchNet, CasMVSNet).

As shown in Figure 4.10, our method is able to reconstruct very accurate

Methods	Chamfer Distance ↓	Accuracy ↓	Completeness ↓
IDR [217]	0.89	1.02	0.79
NeuS [197]	0.77	0.85	0.68
NeuralWarp [34]	0.69	0.68	0.69
COLMAP [167]	0.49	0.40	0.58
ACMMP [204]	0.42	0.46	0.39
PatchmatchNet [195]	0.40	0.44	**0.35**
CasMVSNet [60]	<u>0.38</u>	**0.34**	0.43
Ours	**0.36**	<u>0.37</u>	<u>0.36</u>

*Table 4.2: Quantitative evaluation on DTU [82] (49 or 64 images per model). Best scores are in **bold** and second best are <u>underlined</u>.*

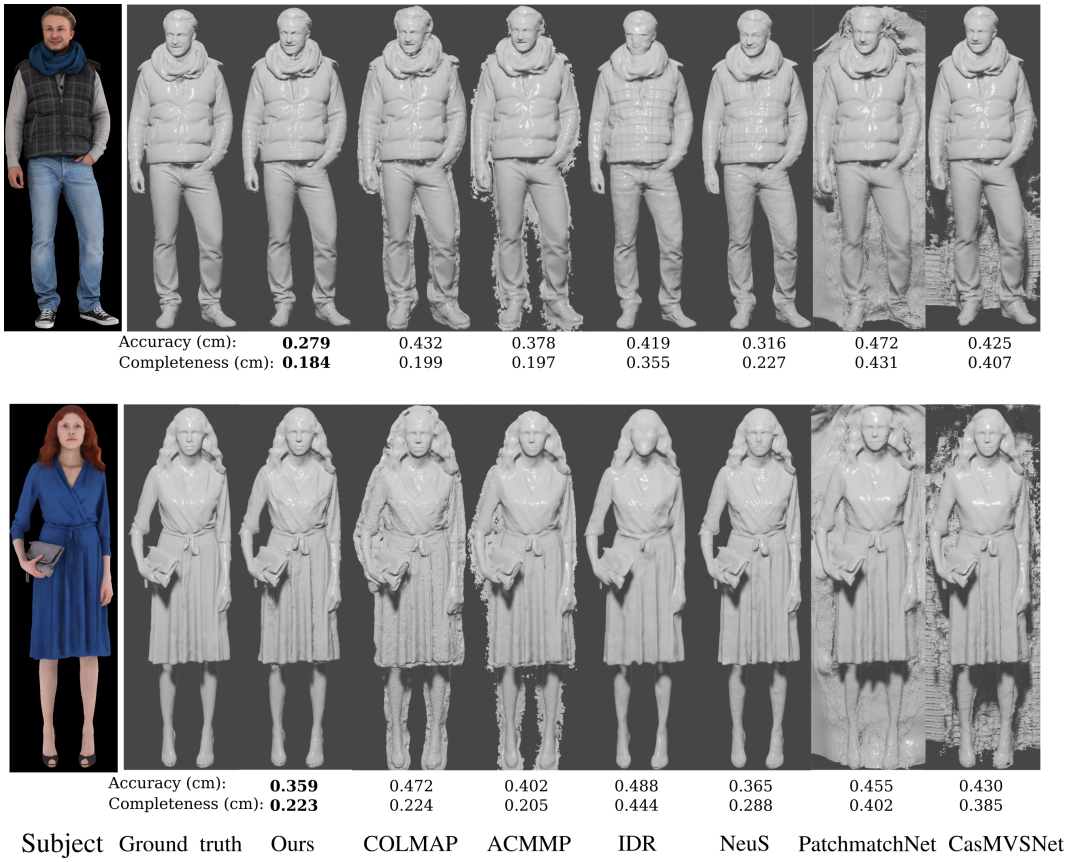


Figure 4.10: Qualitative and quantitative results with 19 images (Renderpeople [2]) and using the baseline photo-consistency prior defined in 4.2.3.

and detailed meshes. COLMAP and ACMMP's are less detailed and more noisy (*e.g.* COLMAP's bottom row). IDR and NeuS also lack details and even fail to reconstruct correctly the geometry of the jacket on the first row because of the checkered texture. In that case, optimizing both the geometry and the

color clearly leads to the wrong geometry. PatchmatchNet and CasMVSNet also work well with very little noise (*e.g.* feet on the first row) and slightly less pronounced details compared to our method (*e.g.* wrinkles on the scarf and sweater on the first row and on the upper part of the dress on the bottom row). The qualitative results are confirmed by the accuracy and completeness metrics computed between each reconstruction and the ground truth mesh.

4.4.5 Reconstruction From Real Captured Data

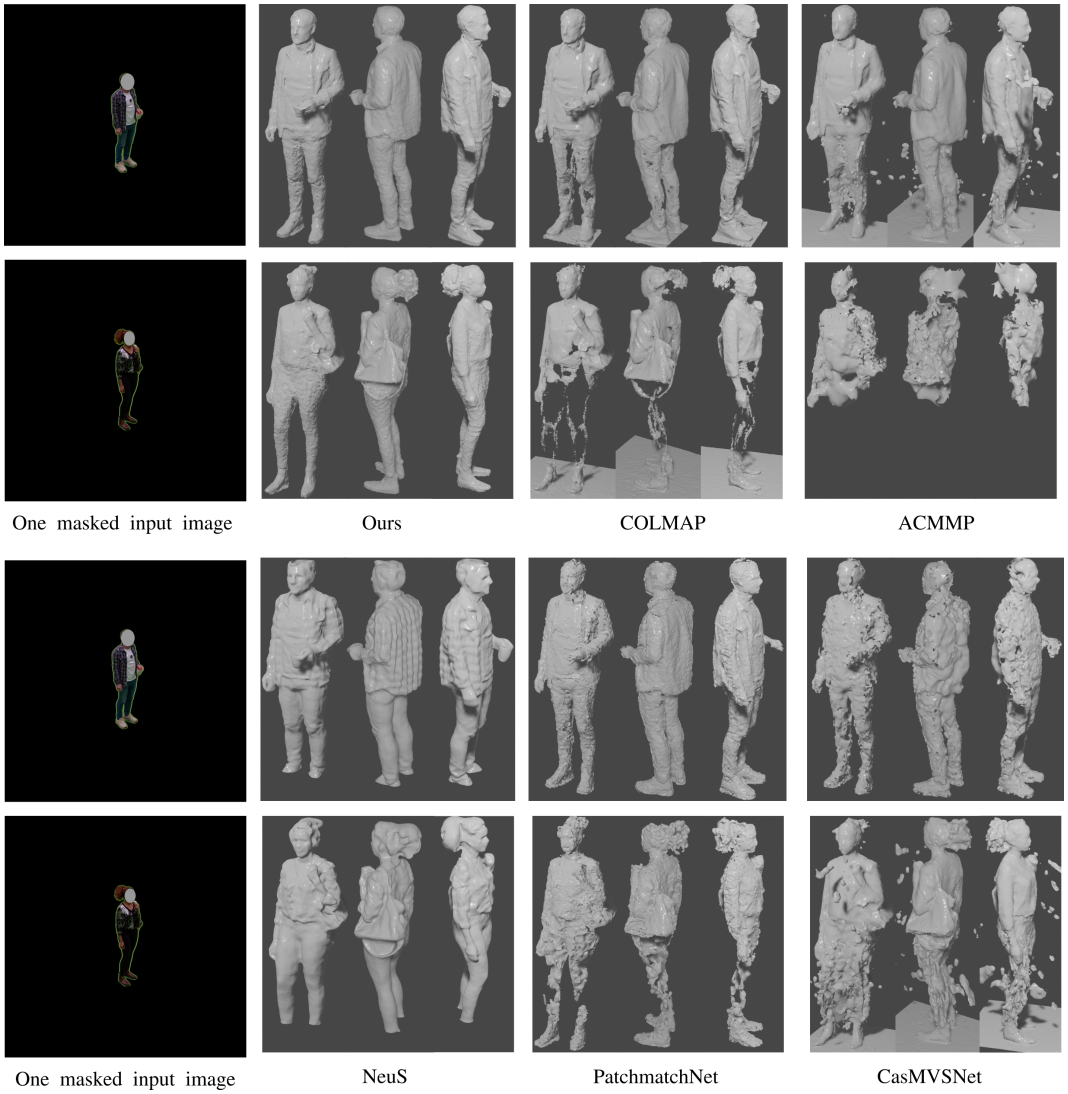


Figure 4.11: Left: One example image. Right: Qualitative comparison using 65 *images from a multi-camera platform.*

To further evaluate the generalization ability of our method, we apply it on human capture data. We use images from a hemispherical multi-camera platform composed of 65 cameras of various focal lengths. This setup is designed to capture humans moving in a large scene so the setting is significantly different from DTU with more distant cameras and significantly wider baselines.

Similarly to the previous experiments we compare with different methods: COLMAP, ACMMP, PatchmatchNet and CasMVSNet. For time reason, we only compare with one optimization method based on differentiable rendering. We choose Neus as it performs better than IDR on DTU and is much faster than NeuralWarp which requires an expensive two stage optimization. Note that PatchmatchNet, CasMVSNet and our learned photo-consistency prior are all trained on the same training set of DTU.

As shown in Figure 4.11, COLMAP performs well with the top row model, despite some holes in the legs. However, it has difficulties with the black pants and the hair with the bottom row model. ACMMP is less precise but we mention that a single optimization iteration was used due to RAM's limitation, even with 64Gb. NeuS reconstructs a nice watertight surface but lacks high-frequency details (*e.g.* faces on both rows) and exhibits poor geometries at different locations due to appearance ambiguities. The deep MVS methods PatchmatchNet and CasMVSNet partially succeed with the top example but fail with the bottom one. This illustrates the generalization issue with the full end-to-end learning-based methods when the inference scenario is substantially different from the training one (*i.e.*DTU). On the other hand, our method shows detailed surfaces with limited noise even on some difficult parts as the black pants on the bottom row. It demonstrates the benefit of a weaker prior with local photo-consistency, that is anyway embedded in a global optimization framework.

4.4.6 Finetuning Inference-based Results

Deep MVS methods like PatchmatchNet and CasMVSNet present the advantage of very fast inference but, as shown in the previous experiment 4.4.5, tend to poorly generalize. From this observation, we experiment in this section the combination of an inference-based method with our optimization-based method. As shown in Figure 4.12, the result of PatchmatchNet can be used as the initialization for the optimization rather than a coarse Visual Hull. Finetuning the results from PatchmatchNet exhibits more details and less noise but it fails to recover from large errors as with the top of the head, the top of the back and the left hip.

4.4.7 Ablation Study

In order to provide more in depth insights into our approach behavior we provide a comparison with two alternative strategies within our framework. First, we mention in Section 4.2.2 that the product over cameras in Equation 4.3 enforces depths to become consistent across views. To evaluate this aspect we show, in Figure 4.13, results with an optimization of depths individually per camera, without camera product. Second, to demonstrate the benefit of the volumetric optimization we also show results with a direct search and selection of the photo-consistency maximum along rays without optimization.

In Figure 4.13, it can be observed that optimizing depth per-camera, in the first alternative, is prone to local minima and that the reconstructed surfaces are quite noisy even with considering synthetic images from Renderpeople.

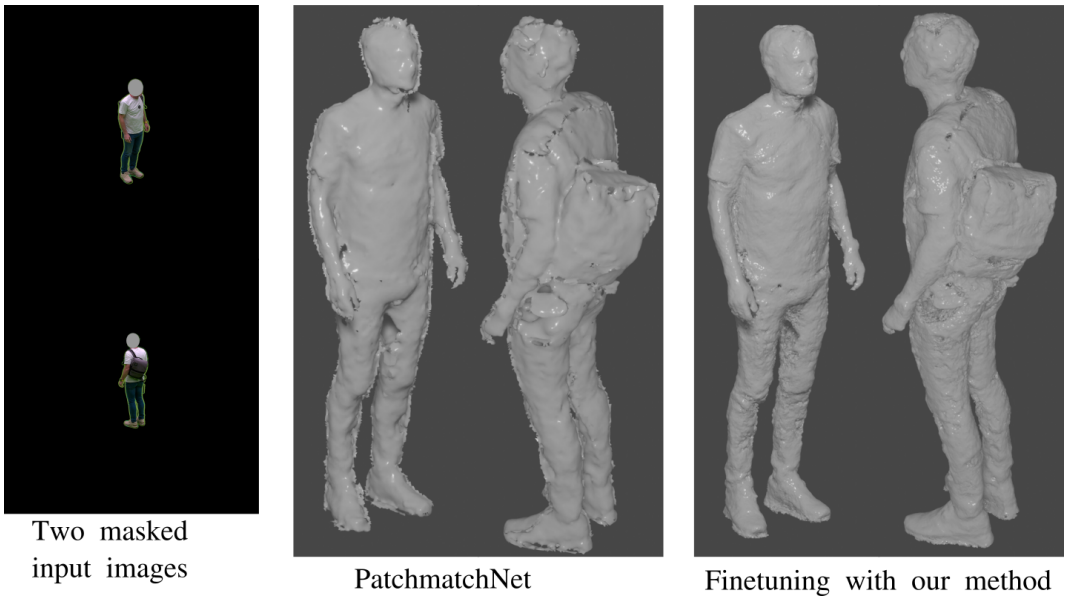


Figure 4.12: Finetuning the reconstruction result of PatchmatchNet with our optimization method.

Moreover, a global search for the maximum of the photo-consistency along each camera ray, in the second alternative, yields somewhat good results with Renderpeople data despite some noise. On the other hand, results are very noisy with real data from DTU. For both real and synthetic data, our proposed strategy that optimizes depth based on a volumetric representation clearly outperforms the two alternatives considered here.

4.5 Conclusion

We have presented a strategy that combines depth optimization, as performed in the latest MVS strategies, with volumetric representations as used in more recent methods based on differentiable rendering. Building on signed distances our SRDF representation allows to optimize multi-view depthmaps in a consistent way by correlating depth prediction with photometric observations along viewing rays. Experiments on real and synthetic data demonstrate the efficiency of our method compared to classic MVS, deep MVS and differentiable rendering-based methods. We also demonstrate the good applicability of our method with a learned photo-consistency prior that generalize well on data completely different from the training set.

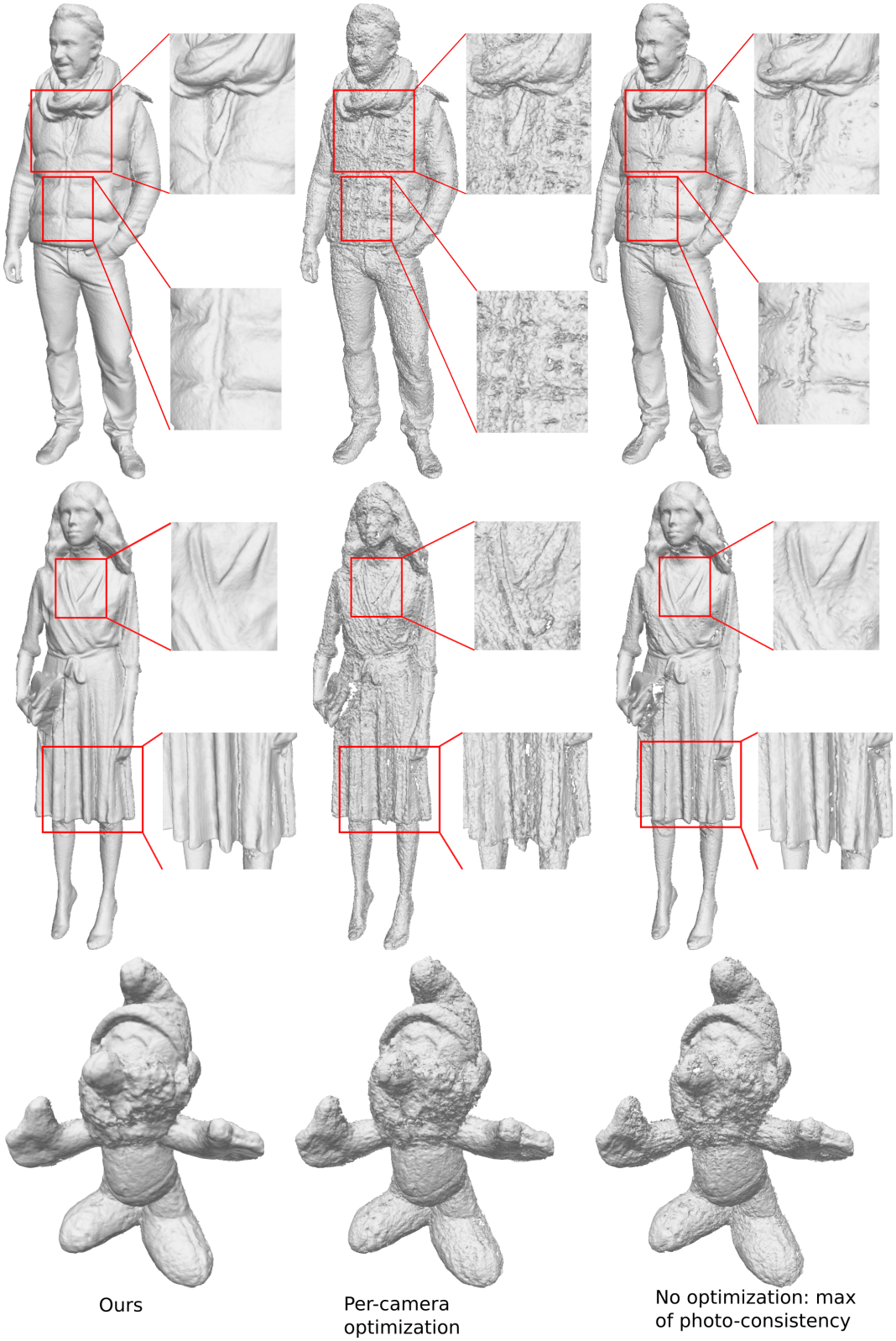


Figure 4.13: Ablation study with two alternative strategies. Reconstructions with data from Renderpeople [2] (two top rows) and DTU [82] (bottom row).

5

Improved Implicit Shape Modeling Using Multi-View Constraints

5.1 Introduction

In this chapter, we combine the insights coming from the two previous ones. On the one hand in Chapter 3, we propose a novel method, called Implicit Shape Modeling here, to reconstruct dressed humans from a limited number of input views. It relies on an implicit shape representation that is trained on a large synthetic dataset of dressed humans and, despite good results, the generalization to unseen data is limited. On the other hand, in Chapter 4, we explore and contribute to the problem of reconstructing 3D shapes from dense viewpoints. In that case, the appearance of a shape is matched within multiple views to enforce multi-view constraints and estimate the surface. By using these local cues, the reconstruction methods generalize much better but reconstruct only the visible surface and require dense viewpoints all around the shape to obtain a complete model.

The motivation of this chapter is to combine the advantages of both approaches by considering a configuration with only a few input views that still include redundant information. In particular, we leverage the ability of the Implicit Shape Modeling method to reconstruct both the visible and hidden parts of a dressed human from only a few views and incorporate multi-view constraints, as used in MVS methods, to offer a better generalization ability and improved surface reconstruction. More specifically, we explore two techniques to combine Implicit Shape Modeling and multi-view constraints.

First, we investigate how to use multi-view constraints as additional information directly during the training step of Implicit Shape Modeling. In this way, we consider that MVS cues are obtained as depth estimations in a pre-processing step. The depth representation is motivated by two elements. A large majority of MVS methods consider depthmaps to represent 3D as it

is relatively light in memory and allows to reconstruct at high resolutions. As shown in StereoPIFu [73], depth estimates from a camera can be efficiently combined with an implicit coordinate-based representation during training.

Second, we consider the reconstruction obtained with Implicit Shape Modeling as a prior and use differentiable rendering to improve and finetune it. In that case, the multi-view constraints do not occur as an explicit photometric matching within the input images. Instead, they are implicitly verified when the method seeks for observation fidelity by rendering multi-view images from its internal geometry and appearance representations.

The main difference between the two strategies is the time at which the multi-view constraints are incorporated into the shape reconstruction of Implicit Shape Modeling. The first strategy offers an *early integration* that takes place before the shape reconstruction while the second strategy consists in a *late integration* when the multi-constraints are used after the shape reconstruction.

In the experiments, we evaluate and compare these two strategies and demonstrate several advantages such as an improved generalization capability, more accurate surface reconstruction of details such as facial expressions or clothing wrinkles and the support of higher resolution images as input.

5.2 Method

In this section, we present the two strategies explored to improve Implicit Shape Modeling such that it leverages redundant information within the input images. An overview of the two approaches is shown in Figure 5.1.

5.2.1 Early Integration

Multi-View Implicit Surface Representation

In this section, we describe a strategy to directly incorporate MVS cues in the architecture of Implicit Shape Modeling presented in Chapter 3. Implicit Shape Modeling uses an implicit function that predicts an occupancy probability for any 3D point in space. For a 3D point X, this function takes as input, for each view, the combination of pixel-aligned features with the z-coordinate of the point X in the local coordinate system aligned with the view. This representation allows to train the deep neural network to solve a classification problem by predicting if a 3D point is inside or outside a closed shape. The surface is implicitly represented as the prediction boundary (0.5-level set) of the occupancy function. In contrast, the process of matching the appearance of a 3D point into different images provides direct information about the surface presence. For example, a large majority of MVS methods directly predict depths which represent the distance between the camera and the surface position. Similarly to [73], we propose to solve this apparent incompatibility through the z-coordinate of the 3D point X.

To facilitate the understanding, we consider that the z-coordinates are defined in the local coordinate system aligned with the i-th camera and leave out the transformations to that local coordinate system as defined in Chapter 3.

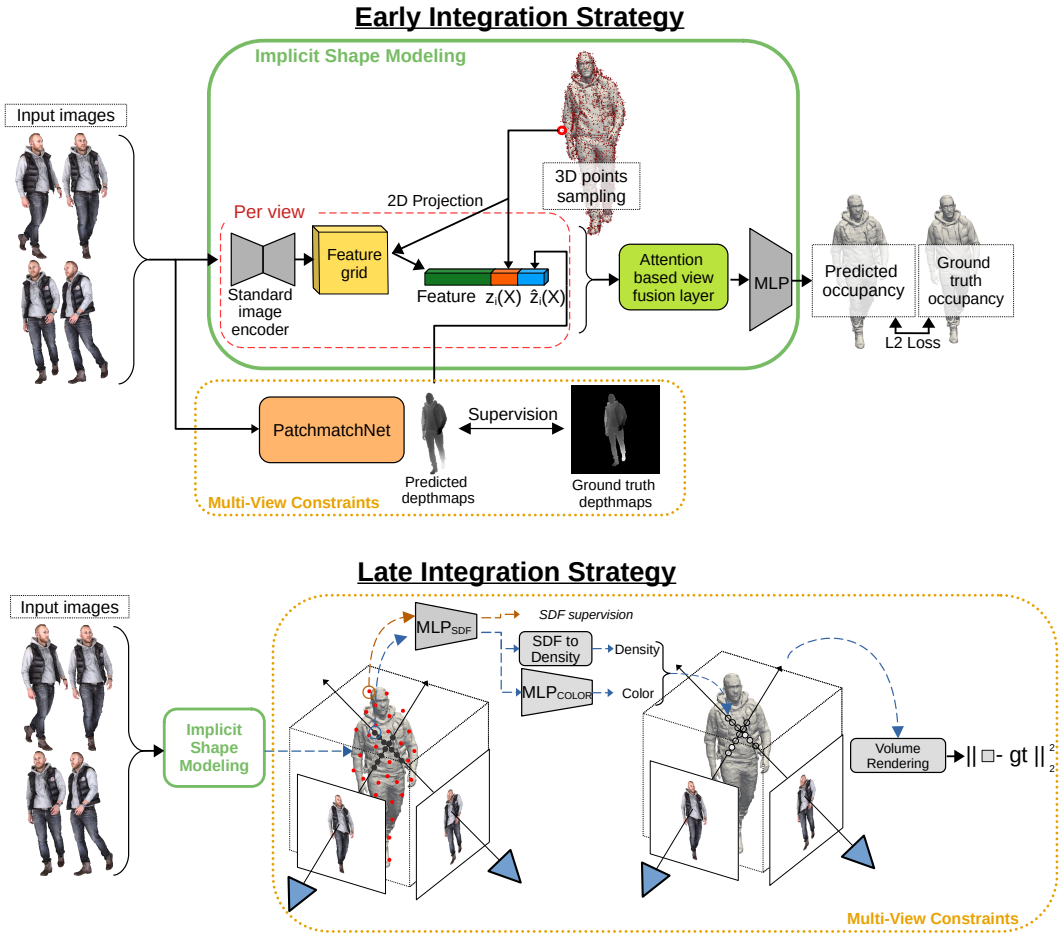


Figure 5.1: Overview of the early and late strategies to incorporate multi-view constraints in the Implicit Shape Modeling method.

In particular, we define $\hat{z}_i(X)$ which is the z-coordinate of the surface estimated by the camera i using MVS cues along the ray cast from C_i to X,

$$\hat{z}_i(X) = z(C_i) - D_i(X), \tag{5.1}$$

where X represents a 3D query point, C_i is the center of camera i and $D_i(X)$ the depth predicted by camera i using MVS cues along the considered ray.

Then, the implicit function f takes the form:

$$f(E_I(\pi_i(X)), z_i(X), \hat{z}_i(X)) = o, \tag{5.2}$$

where π_i is the projection matrix of the i-th camera and $E_I(...)$ returns the 2D image features defined at any location in the image using bilinear interpolation of the values of E_I at pixel locations. $z_i(X)$ is the z-coordinate of the point X, $\hat{z}_i(X)$, as defined previously, the z-coordinate of the surface estimated by the camera i using MVS cues and o the occupancy probability at X. Note also that the inputs of the implicit function f are considered for each input view i.

With this additional input, the deep neural network can deduce information about the difference between $z_i(X)$, the z-coordinate of the 3D query point X, and $\hat{z}_i(X)$, the z-coordinate of the predicted surface position. The sign

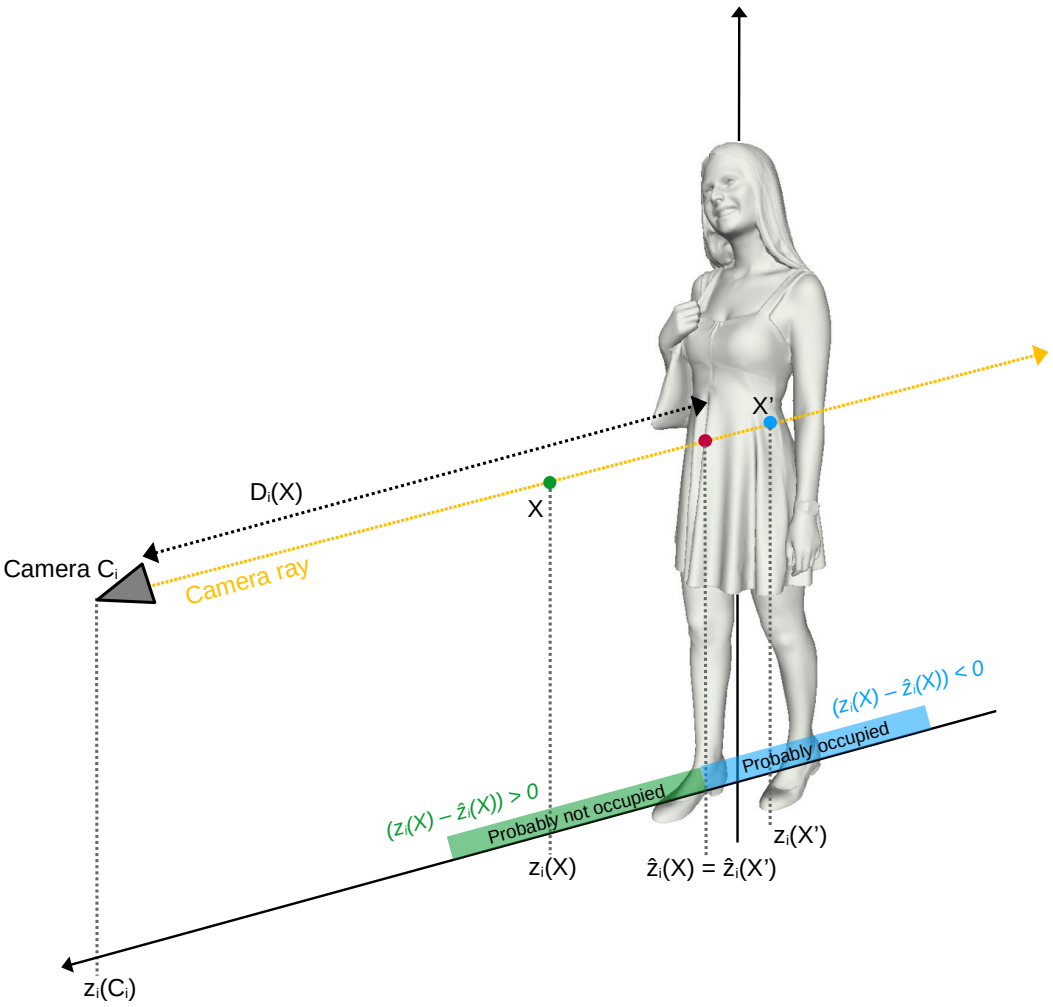


Figure 5.2: The difference between $z_i(X)$, the z-coordinate of the query point, and $\hat{z}_i(X)$, the z-coordinate of the surface estimated with MVS cues, provides information about the occupancy of the point X.

of the difference directly provides helpful information about the occupancy of the query point. As shown in Figure 5.2, a positive value indicates that the query point is in front of the surface and, as a result, is probably not occupied. On the contrary, a negative value indicates that the query point is behind the surface estimation and is more likely to be occupied. Of course, if the query point is behind the human, the value is still negative and the query point is not occupied. As a result, this additional input is only relevant close to the surface estimation.

Depth Estimation

Although the method used to obtain the depth estimation $D_i(X)$ from the multi-view input images is arbitrary, an important feature to consider is the runtime. Indeed, the final pipeline is intended to be trained on a large dataset which involves a very large number of depthmap estimations. For that reason, we leave out all the time-consuming optimization-based methods and take

benefit of end-to-end deep learning-based MVS methods. Once trained, these methods offer fast inference which allows to predict a depthmap at most a few seconds depending on the desired resolution. In our implementation, we use PatchmatchNet [195] which offers very good performances in terms of accuracy, runtime and GPU memory usage.

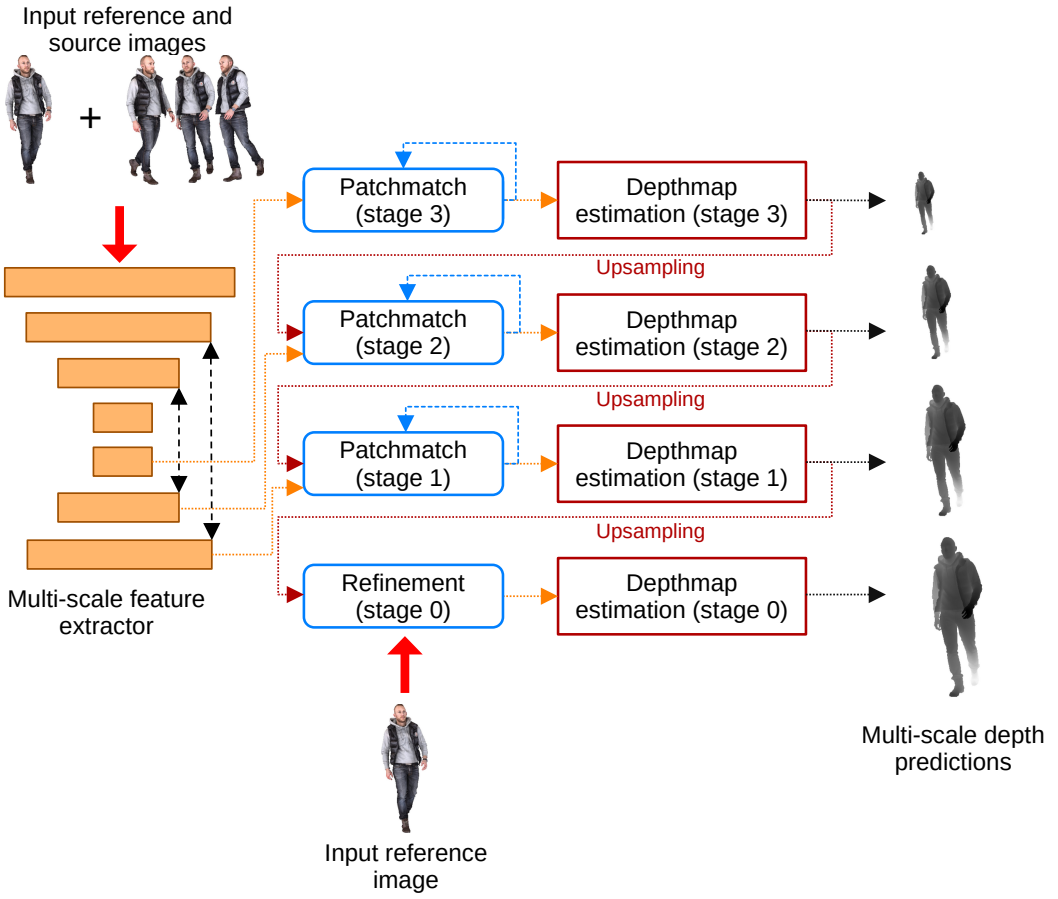


Figure 5.3: Architecture of PatchmatchNet.

PatchmatchNet is an an end-to-end trainable architecture inspired by the Patchmatch [8] algorithm which predicts depthmaps for a reference view by using different source views. The global structure of the pipeline is shown in Figure 5.3. Given N input images, multi-scale features are first extracted using a Feature Pyramid Network (FPN) [109]. Then, three stages are used in a coarse-to-fine manner to run Patchmatch and depthmap prediction. The depthmap predicted at each stage is upsampled and used as input to the next finer stage and on each stage, the Patchmatch module is run multiple times. Finally, a fourth stage of refinement takes as input the upsampled depthmap from the previous stage and the reference image to predict the depthmap at the original resolution.

The main component of PatchmatchNet is the learnable Patchmatch module that performs the following three main steps:

- Initialization and local perturbations: random depth hypotheses are generated in a pre-defined range $[d_{min}, d_{max}]$ at the very first iteration or

around the depth prediction from the previous iteration for the others.

- Adaptive propagation: depth hypotheses are propagated to the neighbouring pixels that are likely to correspond to the same surface. This step augments the depth hypotheses obtained in the initialization and local perturbations step.
- Adaptive evaluation: by using differentiable warping, a matching cost is computed for all depth hypotheses for each pixel and a final depth value is predicted for each pixel.

Losses

To train our global pipeline built with Implicit Shape Modeling and PatchmatchNet, we use two losses and ground truth supervision.

First, for PatchmatchNet we adopt the same loss function as in the original method. The output of the Patchmatch modules of each stage for all the iterations is supervised with an L1 loss between the predicted depthmap and the ground truth depthmap at the same resolution. Similarly, the last refinement stage is supervised with the same metric which gives the loss:

$$L_{\text{patchmatchnet}} = \sum_{k=1}^{3} \sum_{i=1}^{n_k} |D_i^k - \hat{D}_i^k| + |D_{\text{refinement}} - \hat{D}_{\text{refinement}}| \tag{5.3}$$

where k represents the stage, n_k the maximum number of iterations of the Patchmatch module at stage k and D_i^k and $\hat{D}_i^k$ respectively the depthmap predicted by PatchmatchNet and the ground truth depthmap at iteration i of stage k. Additionally, $D_{\text{refinement}}$ and $\hat{D}_{\text{refinement}}$ are respectively the predicted and ground depthmaps after the refinement module.

Second, to train Implicit Shape Modeling we adopt the same L2 loss as in Chapter 3 applied on the predicted occupancy of each sample point.

$$L_{\text{MVPIFu}} = \frac{1}{N} \sum_{i=1}^{N} (o^i_{\text{predicted}} - o^i_{gt})^2 \tag{5.4}$$

where N is the number of 3D sampled points and $o^i_{predicted}$ and o^i_{gt} respectively the predicted and ground truth occupancy values for the i-th sample.

5.2.2 Late Integration

Neural Implicit Representation

As an alternative, we explore a late integration strategy of multi-view constraints that can be seen as a post-processing step applied on top of an initial reconstruction. A key element to consider here is the compatibility between the two representations used to obtain the complete initial reconstruction and to incorporate the multi-view constraints. In Implicit Shape Modeling, the representation is a continuous implicit function that can be discretized and transformed into a mesh. In contrast, the state-of-the-art MVS methods use multi-view depthmaps as 3D representation for the visible parts which cannot

be easily combined with the implicit function or a mesh in a post-processing step. As a solution, we consider recent differentiable rendering-based methods that employ an internal implicit representation and use the initial shape reconstructed with Implicit Shape Modeling as a prior during the optimization. In that case, the multi-view constraints are implicitly verified by the rendering process that targets fidelity with multi-view observations.

Differentiable Rendering-Based Reconstruction

In practice, we use NeuS [197] which achieves very good results for surface reconstruction by using volumetric rendering. Inspired by Nerf [129], NeuS uses two functions to encode both the geometry and the appearance of a scene. The color function $c : \mathbb{R}^3 \times \mathbb{S}^2 \mapsto \mathbb{R}^3$ is standard and maps an input 3D point $X \in \mathbb{R}^3$ and a viewing direction $v \in \mathbb{S}^2$ to a color $C \in \mathbb{R}^3$. The second function, that relates to the geometry of the scene, $f : \mathbb{R}^3 \mapsto \mathbb{R}$ maps a 3D point $X \in \mathbb{R}^3$ to its signed distance from the object. This definition differs from the original one in Nerf, in which the geometry function maps X to a density value. In NeuS, the surface S of an object is therefore represented as the 0-level set of the SDF: $S = \{X \in \mathbb{R}^3 \mid f(X) = 0\}$.

To render an image from its internal representation, NeuS uses the standard volume rendering technique. For that, a ray r cast from the center $o \in \mathbb{R}^3$ of a camera in direction $v \in \mathbb{R}^3$ (with $||v|| = 1$) is parameterized as $r(t) = o + td$ with $t >= 0$. The color C of the pixel corresponding to the ray r, is computed by accumulating the color over the ray r:

$$C(r) = \int_{t_n}^{t_f} T(t)\sigma(r(t))c(r(t), d)dt, \tag{5.5}$$

where t_n and t_f represent the near and far bounds of the scene and $\sigma(r(t)) \in \mathbb{R}^+$ is the density value at t. $T(t)$ represents the accumulated transmittance, the probability that the ray traverses from t_n to t without hitting other particles and is defined as:

$$T(t) = exp(-\int_{t_n}^{t} \sigma(r(s))ds. \tag{5.6}$$

To incorporate the SDF function in the volume rendering, NeuS derives a mapping function Ω based on the Sigmoid function that transforms the SDF into a density function $\sigma(t) = \Omega(f(r(t)))$.

In practice, NeuS uses the same trapezoid quadrature approximation as in Nerf to obtain a discrete version of equation 5.5 and evaluates it on n points $\{p_i = o + t_i v;\ i = 1, ..., n;\ t_i < t_{i+1}\}$ sampled along the ray. The color of the corresponding pixel is therefore estimated as:

$$\begin{aligned} \hat{C}(r) &= \sum_{i=1}^{N} T_i \alpha_i c_i, \text{ where} \\ T_i &= exp(-\sum_{j=1}^{i-1} (\sigma_j \delta_j). \end{aligned} \tag{5.7}$$

T_i denotes the discrete accumulated transmittance, δ_j is the distance from sample $j+1$ to sample j and α_i is the discrete opacity value from alpha com-

positing at sample point p_i which is given by

$$\alpha_i = 1 - exp(-\sigma_i \delta_i). \tag{5.8}$$

Integration of Multi-View Constraints

Alone, NeuS is naturally insufficient to reconstruct a complete 3D avatar from only a limited input number of views but the intuition is that it can help to improve an initial reconstruction obtained with Implicit Shape Modeling.

Ideally, the optimization procedure of NeuS could be directly integrated into Implicit Shape Modeling by considering the MLP decoder of the latter as the geometry network of NeuS and replacing the occupancy-based representation with a signed distance function. However, in order to achieve a good generalization ability, Implicit Shape Modeling heavily relies on multi-scale pixel-aligned features that are extracted from the input images, combined in an attention-based fusion module and given as input to the MLP decoder. This additional processing should be performed for each sample point and adds a significant runtime overhead compared to the light MLP of NeuS which directly takes a coordinate and a direction as input. Since optimizing NeuS involves a very large number of forward passes of the MLP, this solution becomes infeasible.

As a solution, we propose to use the mesh representation as proxy and introduce it in NeuS in the form of an additional loss. By doing that, we obtain a strong prior for the geometry optimization in NeuS. In particular, the geometry of the scene, encoded as the function $f(x)$, that predicts the SDF for any 3D point, is additionally supervised with a pseudo ground truth SDF computed from the mesh reconstructed with Implicit Shape Modeling.

The original sampling strategy of NeuS is not sufficient to cover the entire scene as 3D sample points are only selected on the ray sampled at each iteration and mostly located around the visible surface. To efficiently learn the geometry of the entire scene, we propose to use a similar spatial sampling strategy as in Implicit Shape Modeling that allows to capture the geometry of the complete scene with fine details. The sampling strategy combines a uniform sampling in a bounding box of the scene and an importance sampling closer to the surface of the mesh. The important samples are obtained by sampling the surface of the mesh and adding offsets that follow a normal distribution $N(0, \sigma)$. The samples obtained with this strategy are processed by the geometry network and the following reconstruction loss is used to supervise its output:

$$L_{\text{reconstruction}} = \frac{1}{N} \sum_{i=1}^{N} (f(X_i) - \hat{f}(X_i))^2 \tag{5.9}$$

where N is the total number of samples obtained with the uniform and importance sampling, $f(X_i)$ is the SDF of the point X_i predicted by the geometry network from NeuS and $\hat{f}(X_i)$ is the pseudo ground truth SDF of the point X_i computed from the mesh predicted with Implicit Shape Modeling. This loss is combined with the original color, Eikonal [59], and mask losses used by NeuS, with a weight γ set to a high value in our experiments. By using this reconstruction loss and the associated sampling strategy, NeuS is optimized with a

strong prior on the geometry of the scene which is essential to reconstruct a complete 3D avatar from only a few observations.

More Efficient Ray Sampling

Moreover, the initial reconstruction can be used to additionally guide the sampling procedure of Neus. We introduce two new schemes. First, as we have an initial reconstruction we can easily obtain silhouette information for each of the input views. Based on that, we limit the selection of the pixels during the optimization to the foreground pixels. This ensures that the rays corresponding to the selected pixels actually hit the surface and contribute to the rendering of the image. This dramatically reduces the number of possible rays which speeds up the optimization. To account for possible errors in the initial reconstruction we dilate the silhouettes in order to also consider pixels that are close to the outline of the human. In addition to the sampling of pixels, we also propose to improve the sampling of the points on each ray based on the current depth estimation. For that, we render a depthmap for each of the input views by using our initial mesh reconstruction and then limit the sampling range of NeuS to an interval centered around the corresponding depth.

5.3 Implementation Details and Training Strategy

In this section we specify some implementation details about the different methods as well as the training procedure for PatchmatchNet and Implicit Shape Modeling.

5.3.1 Implementation Details

First, Implicit Shape Modeling exactly follows the description from Section 3.3.1 except the 3D local context encoding which is not used here to reduce the training time.

For PatchmatchNet, we adopt the same architecture as the original paper. In particular, we set the number of iterations of Patchmatch to respectively 1, 2 and 2 for the stage 1, 2 and 3. 48 depth hypotheses are used for each pixel in the very first iteration of Patchmatch. For the local perturbations of the other iterations, we generate 8, 8 and 16 additional depth hypotheses respectively on stages 1, 2, 3 with normalized inverse ranges of sizes 0.04, 0.09 and 0.38. For the adaptive propagation of depth predictions, we use 8 and 16 neighbours respectively on the stages 2 and 3 and no propagation on the stage 1. Finally, 9 neighbours are used for the cost aggregation of the adaptive evaluation on all the stages.

For NeuS, we follow the architecture of the original paper with an MLP for the SDF that contains 8 hidden layers of size 256 and an MLP for the color that is built with 4 hidden layers of size 256. The feature vector given to the color MLP also has a size of 256. The positional encodings contain respectively 6 frequencies for the position of the 3D point and 4 for the direction of the ray.

At each iteration, 512 rays are randomly selected from one image and 64 points are sampled on each of them for the coarse sampling and 64 additional points for the fine sampling. We also sample 10000 points per iteration following the uniform and importance samplings. To guide the sampling of NeuS with the prior, we use a dilation kernel of size 15 and a range of size 0.2 (in the unit sphere) along the ray around the current surface. This procedure is enabled after 10k iterations and the final model is optimized for 200k iterations.

5.3.2 Training Strategy

In contrast to NeuS, which is optimized per-scene, Implicit Shape Modeling and PatchmatchNet are both trained on a full synthetic dataset created using Renderpeople [2] meshes. In total, we have 900 meshes for training. As input, we render images of resolution 512×512 all around the meshes and to obtain realistic renderings we use the same strategy as PIFu [162] which combines spherical harmonics and visibility considerations. Both Implicit Shape Modeling and PatchmatchNet are trained with a similar view selection strategy that consists of 4 views with a pseudo random interval of $30° + rand([-5°, -4°, ..., 5°])$ between the views. Implicit Shape Modeling is optimized during 100 epochs using the root mean square propagation algorithm with a learning rate of 1×10^{-4} that is divided by 10 at iterations 60 and 80. For PatchmatchNet, we use the model trained on DTU [82] proposed in the original paper and finetune it for 50 epochs using Adam optimizer with parameters ($\beta_1 = 0.2, \beta_2 = 0.999$) and a learning rate of 0.001.

5.4 Experimental Results

In this section, we introduce the evaluation protocol and evaluate quantitatively and qualitatively the two methods explored in this chapter.

5.4.1 Settings

To evaluate the methods, we use two different datasets of dressed humans. First, we consider the test split of our Renderpeople dataset composed of new meshes that were never seen during training. To obtain more out-of-distribution test data, we create a second test set by rendering images with meshes from the THuman2.0 dataset [223]. The latter contains high-quality human scans, geometry and textures, captured by a dense DLSR rig.

To evaluate quantitatively the reconstructed human meshes, we first compute the Chamfer Distance (**CD**) between the ground truth mesh and the reconstructed mesh. This metric includes the accuracy (**Acc**) and the completeness (**Comp**) which consider average distances between meshes to measure the global quality of the reconstructions. To focus more on local details, we also consider surface normals of the reconstructed and ground truth meshes and compute the $\mathbb{L}_2$ and cosine distances between them (**Norm Cosine** and **Norm L2**, respectively).

The objectives of the experiments is to compare the two proposed strategies with each other and with the vanilla Implicit Shape Modeling method. For

reference, we also show the results of two other baselines PIFuHD and StereoPIFu that respectively use a single input image and two rectified input images. However, this comparison is difficult because we use their pre-trained models since no training code is available to re-train the methods on our dataset. As a result, some of our test subjects may have been used during the training of PIFuHD and StereoPIFu. At test time, we select a frontal view for PIFuHD and use the rendering scripts from StereoPIFu to obtain the two rectified images.

5.4.2 Comparisons on Renderpeople

Methods	CD (cm) ↓		Acc (cm) ↓		Comp (cm) ↓		Norm Cosine ↓		Norm L2 ↓	
	mean	median	mean	median	mean	median	mean	median	mean	median
PIFuHD [163]	2.024	1.806	2.028	1.686	2.019	1.873	0.207	0.218	0.601	0.609
StereoPIFu [73]	0.754	0.709	0.662	0.645	0.847	0.767	0.103	0.098	0.364	0.355
Implicit Shape Modeling [237]	0.491	0.440	0.465	0.410	0.517	0.465	0.096	0.087	0.359	0.343
Early integration	**0.448**	**0.377**	**0.415**	**0.367**	**0.481**	**0.406**	0.091	0.086	0.348	0.333
Late integration	0.485	0.434	0.457	0.406	0.513	0.468	**0.083**	**0.075**	**0.314**	**0.303**

Table 5.1: Quantitative results and comparisons between PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling and our two strategies that integrate multi-view constraints. PIFuHD uses a single input image, StereoPIFu two rectified images and the others four images. The evaluation dataset is built with meshes from Renderpeople. Best scores are in **bold.**

In our first experiment, we compare the reconstructions obtained with PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling and the results obtained with our two strategies that incorporate multi-view constraints. Our test dataset contains 35 subjects from Renderpeople and we use 4 input views with an interval of 30° between them which is close to the setting used during training.

Figure 5.4 presents the qualitative results. The early integration strategy allows a better capture of the global shape with less missing parts and wrong geometries compared to the other methods. It is especially visible with the jacket on the first row, the box on the second row, the shoes on the third row and the robe on the last row. MVS cues also improves the reconstruction of surface details as can be seen on the face of each subject. The late integration strategy mostly improves the reconstruction of surface details as can be seen on the faces or clothing wrinkles. It also achieves better reconstructions of the global shapes compared to Implicit Shape Modeling but the gain seems more subtle than with the early integration strategy. This can be explained by the fact that the MVS cues are directly used for shape estimation in the early integration strategy, while in the second strategy they are used to improve an initial reconstruction. The latter might include significant errors that are difficult to correct in a post-processing step.

The qualitative results are confirmed with the quantitative evaluation shown in Table 5.1. We clearly see that multi-view constraints incorporated early in Implicit Shape Modeling help globally the reconstruction process. This

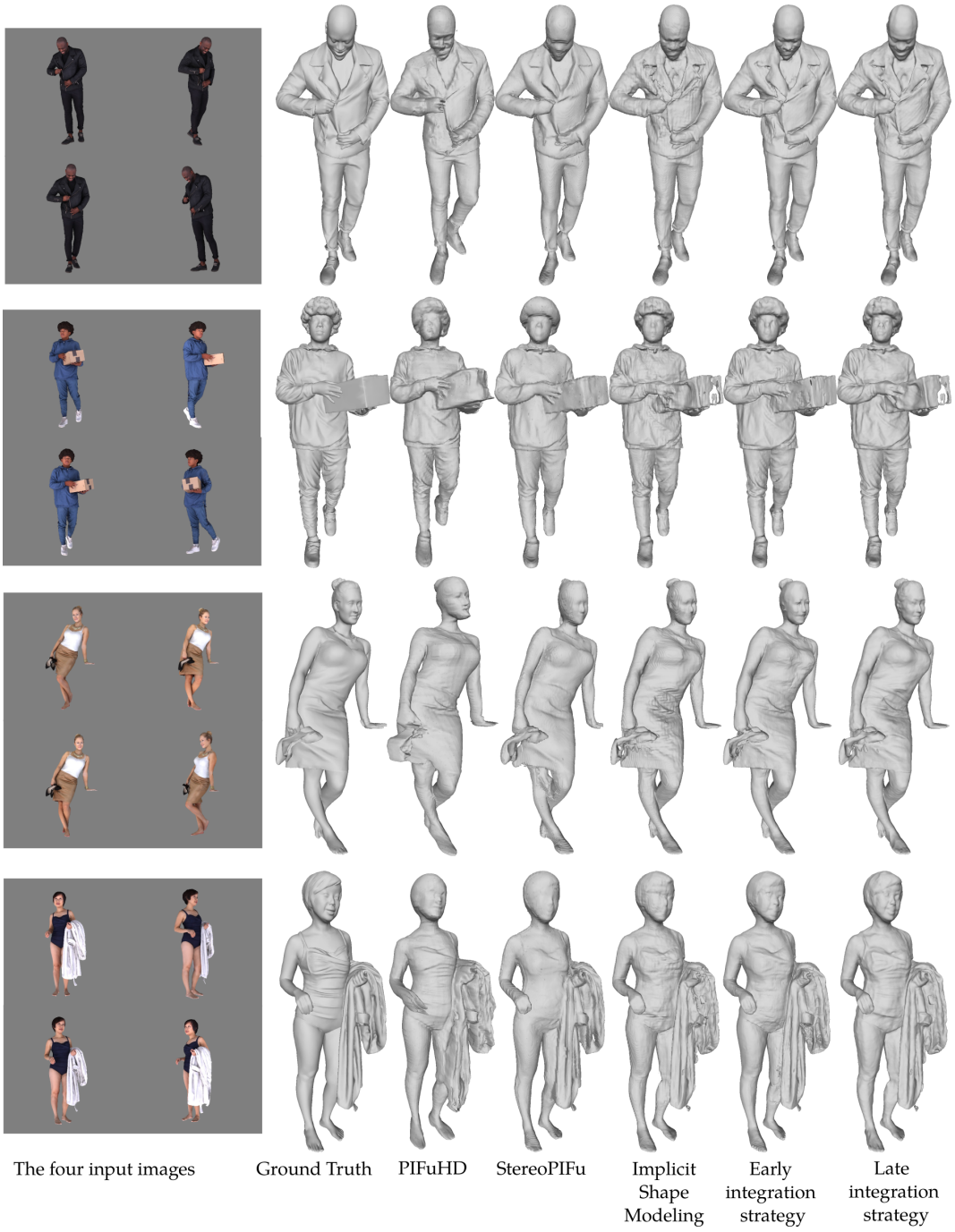


Figure 5.4: Qualitative results and comparisons on Renderpeople between PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling method and the two proposed strategies that integrate multi-view constraints. PIFuHD uses a single input image, StereoPIFu two rectified images and the others four images. The four input images are rendered from Renderpeople meshes with the rotations around the vertical axis : 15°, 45°, 315°, 345°.

strategy obtains the best Chamfer distance among all the considered methods as well as improved metrics on the surface normals compared to the vanilla Implicit Shape Modeling method. For the late integration of multi-view con-

straints with the differentiable rendering-based optimization, we observe a little benefit for the Chamfer distance and a strong improvement for the metrics on normals. This demonstrates that the optimization cannot recover large errors like wrong geometries or missing parts but is efficient at capturing details of the visible surfaces.

5.4.3 Comparisons on THuman2.0

Methods	CD (cm) ↓		Acc (cm) ↓		Comp (cm) ↓		Norm Cosine ↓		Norm L2 ↓	
	mean	median	mean	median	mean	median	mean	median	mean	median
PIFuHD [163]	3.117	2.932	3.196	3.126	3.037	2.943	0.260	0.262	0.690	0.693
StereoPIFu [73]	0.767	0.765	0.742	0.744	0.792	0.797	0.110	0.105	0.380	0.370
Implicit Shape Modeling [237]	0.531	0.518	0.519	0.491	0.544	0.530	0.107	0.106	0.381	0.377
Early integration	**0.471**	**0.453**	**0.456**	**0.430**	**0.486**	**0.470**	0.098	0.091	0.361	0.350
Late integration	0.528	0.513	0.513	0.493	0.543	0.527	**0.092**	**0.089**	**0.342**	**0.339**

Table 5.2: Quantitative results and comparisons between PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling and our two strategies that integrate multi-view constraints. PIFuHD uses a single input image, StereoPIFu two rectified images and the others four images. The evaluation dataset is built with meshes from THuman2.0. Best scores are in **bold.**

To further evaluate the two strategies, we apply the methods on the dataset built with 55 meshes from THuman2.0.

As shown in Figure 5.5, it is obvious that the multi-view constraints bring a significant gain with both integration strategies. The reconstruction of the global shape as well as surface details are better. Similarly to the previous experiment with data from Renderpeople, the early integration is more efficient to recover an accurate global shape with less missing parts or erroneous geometries. On the contrary, the late integration strategy recovers better surface details as can be seen on faces.

Quantitative results are presented in Table 5.2 and confirm the qualitative observations. The improvement obtained with the early integration of multi-view constraints is even more significant than in the previous experiment. This is explained by the fact that data from THuman2.0 is more out-of-distribution compared to the test data from Renderpeople. In particular, this highlights the better generalization of the method thanks to the MVS cues. Metrics on surface normals are also improved which demonstrates also the better capture of details. Similarly to the previous experiment, the late integration strategy slightly improves the reconstruction of the global shape but remains less efficient than the early integration and achieves the best results for the reconstruction of details with the metrics on the surface normals.

5.4.4 Generalization Ability

In this experiment, we assess the generalization ability of the early integration strategy compared to the vanilla Implicit Shape Modeling. For that, we consider test data more different from the training set under two scenarios. First,

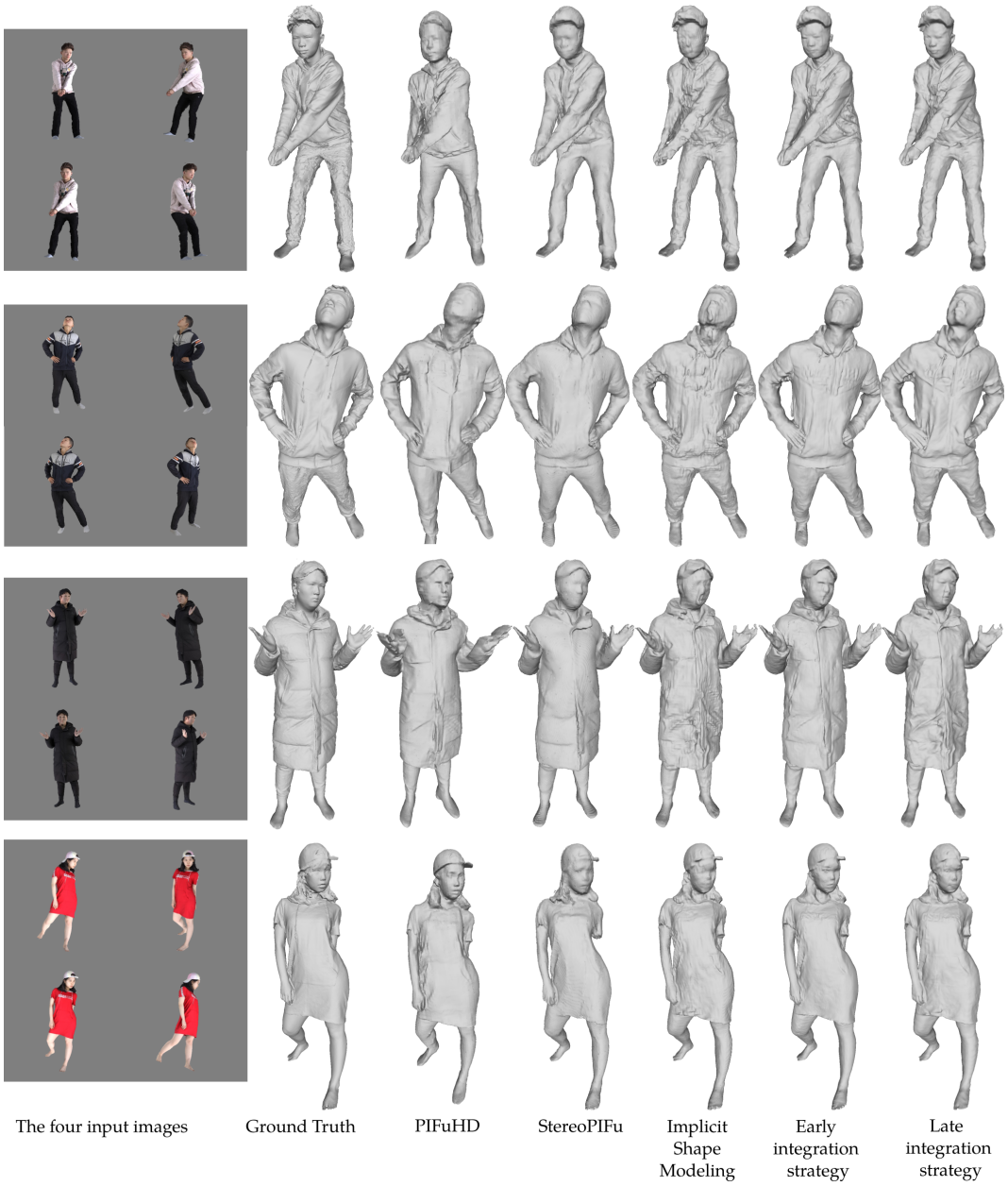


Figure 5.5: Qualitative results and comparisons on THuman2.0 between PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling method and the two proposed strategies that integrate multi-view constraints. PIFuHD uses a single input image, StereoPIFu two rectified images and the others four images. The four input images are rendered from THuman2.0 meshes with the rotations around the vertical axis : 15°, 45°, 315°, 345°.

we consider more difficult poses that are rare or not present in the training dataset. Second, we use as input multi-view images with an interval of 20°, significantly different from the training setting.

Figure 5.6 shows the reconstruction results obtained when the human in the input images performs uncommon poses. As the training dataset, built with Renderpeople, mostly contains meshes in standard poses, the reconstruction

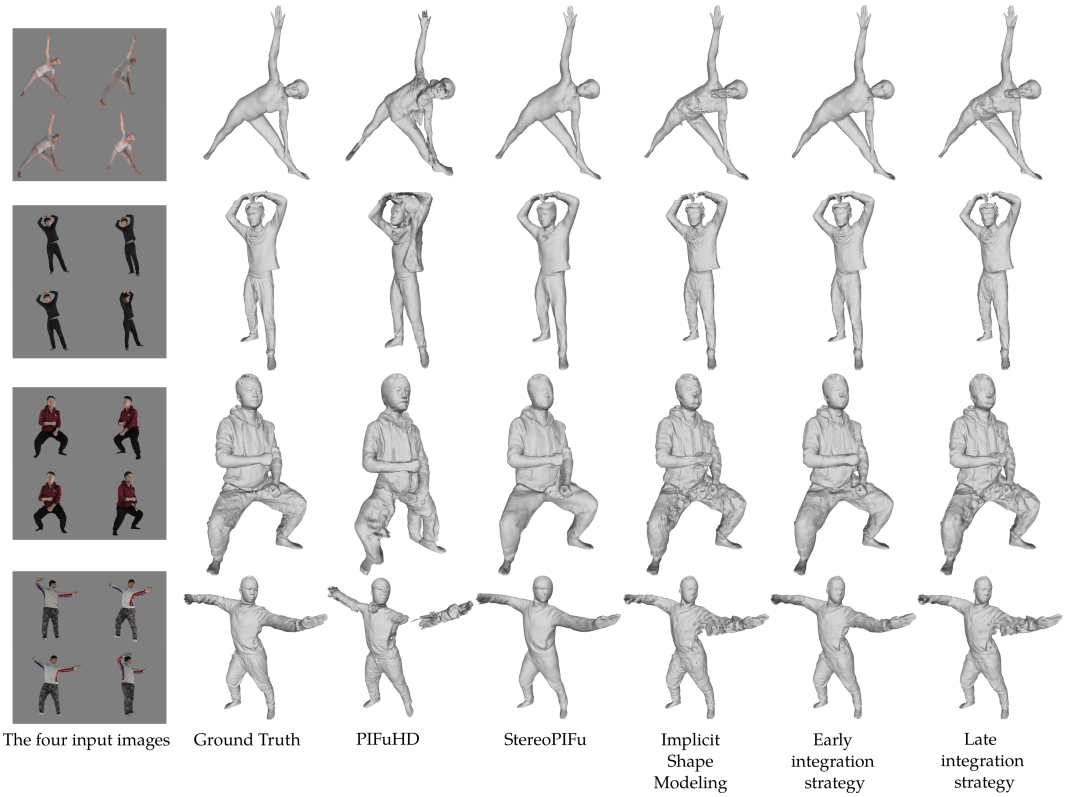


Figure 5.6: Qualitative results and comparisons between PIFuHD, StereoPIFu, the vanilla Implicit Shape Modeling method and the two proposed strategies on more challenging poses. The 4 input images are rendered from Renderpeople (row 1) and THuman2.0 (rows 2,3,4) meshes with the rotations around the vertical axis : 15°, 45°, 315°, 345°.

task in this experiment becomes more challenging. It is especially visible in the results, as Implicit Shape Modeling fails on some parts and obtains noisy reconstructions. In contrast, the early integration strategy achieves much better results with the additional MVS cues and is not affected by the uncommon poses. In particular, this highlights the efficiency of the local MVS cues that generalize better than the multi-scale features used in Implicit Shape Modeling.

This observation is also verified in the second experiment when the interval between the input views is significantly different from the one used during training. This scenario creates multi-view configurations that are not seen during training which also increases the difficulty of the reconstruction task. The results shown in Figure 5.7 demonstrate that the vanilla Implicit Shape Modeling is affected by that whereas the MVS cues used in the early integration strategy bring more robustness.

5.4.5 High Resolution Input Images

In this experiment, we demonstrate an important advantage shared by the two explored strategies which is the ability to leverage high resolution images as input. The vanilla Implicit Shape Modeling uses input images of resolution

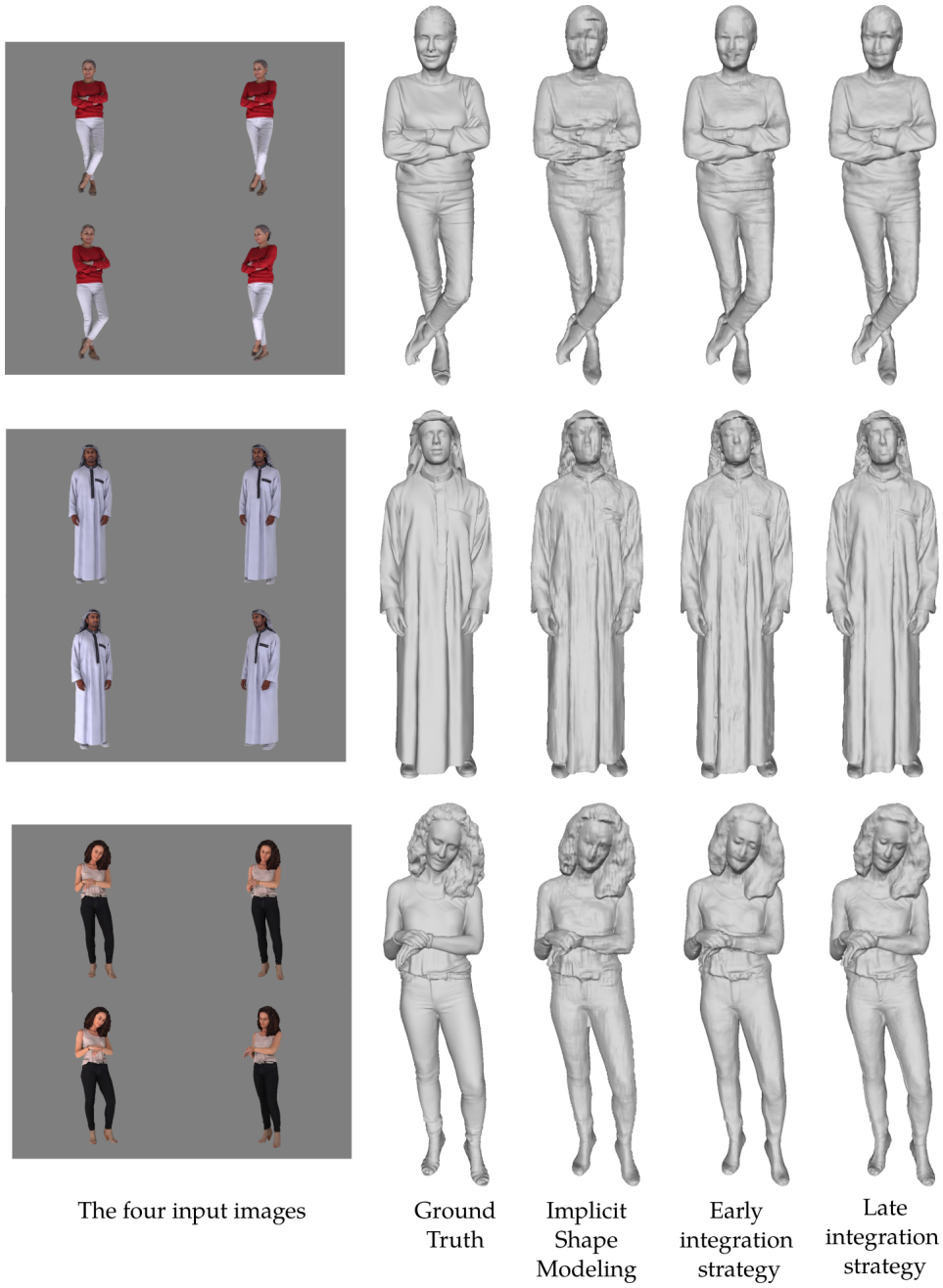


Figure 5.7: Qualitative results and comparisons between the vanilla Implicit Shape Modeling and when multi-view constraints as added following the early integration strategy. The 4 input images are rendered from Renderpeople meshes with the rotations around the vertical axis : 0°, 20°, 40°, 340°.

512×512 and extracts features of size $128 \times 128 \times 256$ for each input view. For memory reason, it is difficult to increase the resolution of the input images and the extracted features. In contrast, PatchmatchNet, and a majority of deep MVS methods, can infer depthmaps at higher resolutions. As a result, the z-coordinate of the surface used as additional input of the implicit function in the early integration strategy, can benefit from higher resolution images. Similarly, in the late integration strategy, the initial reconstruction is obtained

with 512×512 images and then NeuS can optimize its internal representation with higher resolution images. In this experiment, we consider high resolution images of size 2048×2048.

Qualitative results are shown in Figure 5.8 and demonstrate that with higher resolution images as input, both integration strategies allow to obtain significantly improved reconstructions. In particular, we clearly observe that the facial expressions as well as the clothes are more accurately captured.

5.4.6 Computational Efficiency

A key aspect to compare the two strategies relates to the computation time. Despite a long training phase, reconstructing a 3D mesh at test time using Implicit Shape Modeling only involves evaluating the implicit function in a 3D grid and then applying the surface reconstruction algorithm Marching Cubes [117]. The duration of this process depends on the resolution of the 3D grid and takes less than a minute for a grid of size 512^3. The early integration strategy adds an overhead on top of Implicit Shape Modeling as it needs to infer first one depthmap per-image using PatchmatchNet. However, as mentioned in Section 5.2.1, this depthmap inference is very fast and the overhead becomes negligible. PIFuHD and StereoPIFu also build on an implicit representation and have a similar computational efficiency.

In contrast, the late integration strategy does not involve a long training phase but is less efficient at test time. Currently, the optimization with the late integration strategy requires around 15 hours to perform 200k iterations but achieves already good results after only 3 hours of optimization thanks to the improved sampling strategy. In particular, this is shown in Figure 5.10. Recently, different other acceleration techniques such as [24, 131, 178, 187, 219] were also proposed for methods that optimize neural radiance fields using volumetric rendering and adding them in the late integration strategy is conceivable. However, despite a significant possible speed up, the computational efficiency will still remain in favour of the early integration strategy.

5.4.7 Ablation Studies

In this section, we present two experiments to justify the design of the late integration strategy.

First, Figure 5.9 shows the reconstruction result obtained by optimizing NeuS alone with 4 input views. NeuS is not able to reconstruct hidden parts and also fails for the visible surface with this restricted number of input views. In contrast, with the additional reconstruction loss and associated sampling of our late integration strategy NeuS is able to reconstruct a complete and detailed 3D avatar.

Second, Figure 5.10 shows the effect of our improved sampling strategy described in Section 5.2.2. It allows to gather the sample points around the interesting areas while leaving out empty and occluded space. Note that this improved sampling allows for a much faster convergence with more accurate reconstructions even at the beginning of the optimization. With the improved

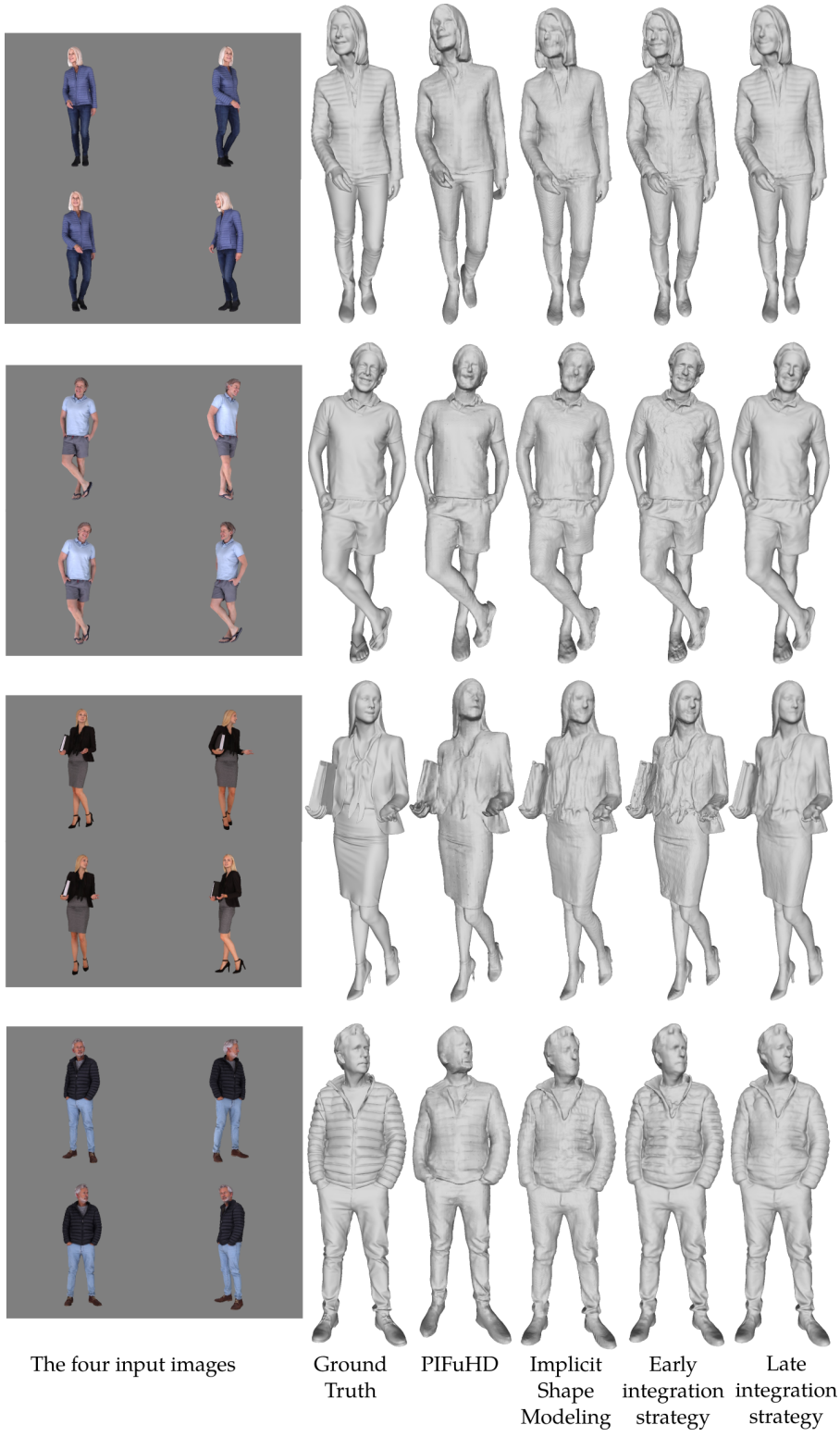


Figure 5.8: Qualitative results and comparisons of the two strategies when higher resolution images are used as input. Implicit Shape Modeling is restricted to use 512×512 *images while PIFuHD and the two explored strategies support* 2048×2048 *images. The 4 input images are rendered from Renderpeople meshes with the rotations around the vertical axis : 15°, 45°, 315°, 345°.*

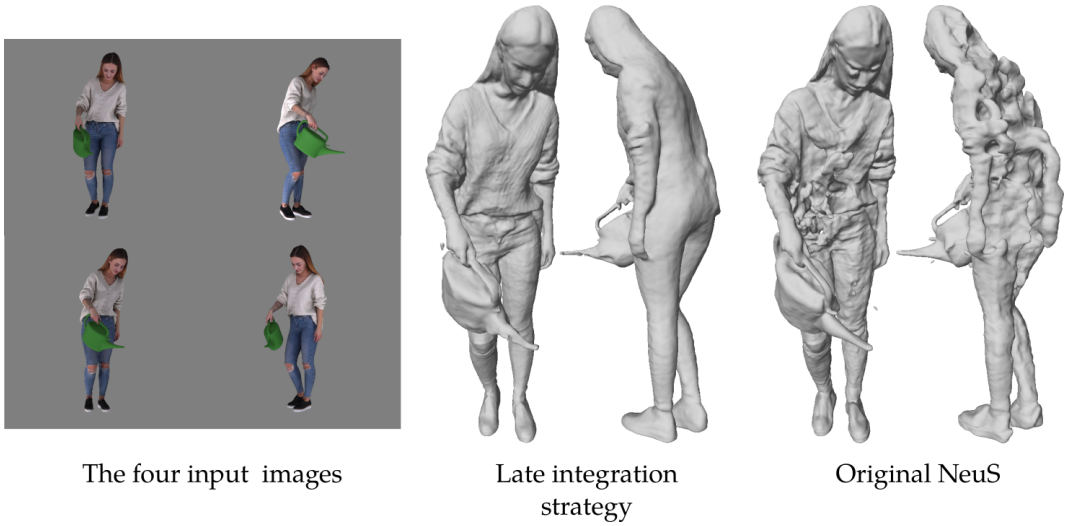


Figure 5.9: NeuS alone is insufficient to obtain a complete and accurate reconstruction of the human with only 4 input views.

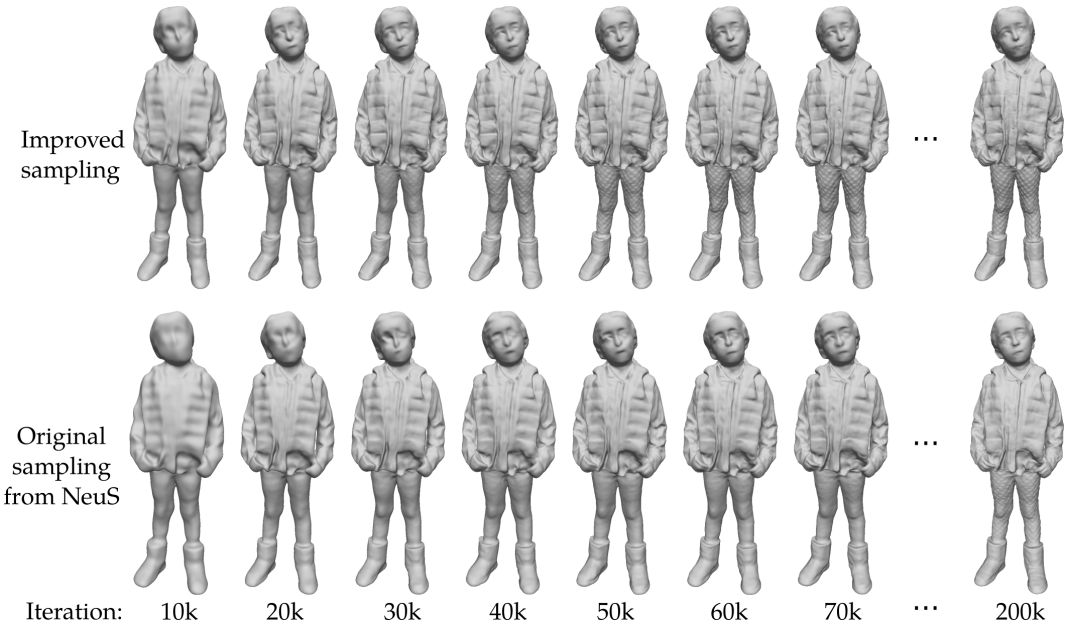


Figure 5.10: Effect of the improved sampling strategy on the optimization.

sampling, only 40k iterations are necessary to obtain a similar result to 200k iterations with the original sampling from NeuS.

5.5 Conclusion

In this chapter, we consider the problem of reconstructing a complete and detailed avatar of a dressed human and potentially accessories from only a few views. In contrast to Chapter 3, we assume that the input images share some redundancy that can be leveraged to integrate multi-view constraints in the Implicit Shape Modeling method. In particular, we explore two differ-

ent strategies, that consider multi-view constraints either before or after the shape reconstruction from Implicit Shape Modeling. We build on top of PatchmatchNet in the first case and NeuS in the second. In several experiments, we demonstrate the superiority of the two methods compared to PIFuHD, StereoPIFu and the vanilla Implicit Shape Modeling method in terms of global shape quality and reconstruction of details. The early integration strategy significantly improves the generalization ability of Implicit Shape Modeling while the late integration is more efficient to reconstruct surface details. Both options also allow to leverage higher resolution images to further improve the reconstruction quality. Finally, the two explored methods are not incompatible and a final strategy that combines both of them is worth considering.

6 Conclusion and Future Work

6.1 Summary

In this thesis, we have presented several contributions to the problem of reconstructing the geometry of a 3D shape from multi-view 2D images. This task becomes more and more important as it meets a growing need of 3D content in many applications from different domains. We have divided our work along three axes.

In Chapter 3, we presented a novel method for 3D reconstruction of dressed humans from a few sparse views. The limited number of input views makes this task very challenging but, at the same time, brings a strong potential to the method as the capture configuration is simplified. Our proposed method builds on top of an implicit representation that is light in memory, able to represent data at high resolutions and easily integrable with deep learning techniques. In particular, we introduced an attention-based view fusion module that efficiently combines the information coming from the different views and a 3D context encoding module that encodes local 3D patterns in the multi-view context and improves the generalization ability of the method. In contrast to existing works, the proposed method also offers spatially consistent reconstructions that allow for arbitrary placement of the person in the input views. We quantitatively and qualitatively evaluated our proposed method on synthetic data to demonstrate superior results compared to the current state of the art. To emphasize the full potential of the method we also applied it on real data from a capture platform and demonstrated much better results than existing methods that use a limited number of views. Additionally, we showed that our approach can even approximate multi-view stereo results with dramatically fewer views. An interesting insight from Chapter 3 is that even if the data-driven strategy allows to infer the 3D geometry of both the visible and the hidden parts, the method remains much more efficient to reconstruct the surface corresponding to the observations. This is particularly visible in the comparisons with monocular strategies and justifies our choice to lift the

single-view input with a few additional views. Moreover, we have shown that our proposed trainable view fusion module follows the same behaviour by assigning strong weights to the views in which a considered 3D point is visible.

In Chapter 4, we considered a more standard reconstruction scenario with dense viewpoints in which the photometric redundancy within the input images is leveraged to estimate the surface position. In that context, we presented a novel strategy that combines the benefit of depth optimization, as performed in the latest MVS strategies, with volumetric integration as used in more recent methods based on differentiable rendering. More specifically, we proposed the SRDF representation, inspired by the SDF, which allows to optimize multi-view depthmaps in a consistent way by correlating depth predictions with photometric observations along viewing rays. We evaluated quantitatively and qualitatively our method on different datasets to demonstrate the efficiency of our method compared to classic MVS, end-to-end deep MVS and differentiable rendering-based methods.

Finally, in Chapter 5, we combine the insights from the two previous chapters. In that sense, we show that the multi-view constraints, as used in Chapter 4, and the data-driven method from Chapter 3, named Implicit Shape Modeling, can complement one another. To allow for this, we used the additional hypothesis which assumes that the input images share some redundancy. We considered Implicit Shape Modeling as the complete shape reconstruction method and explored two ways to integrate the multi-view constraints. On the one hand we proposed an early integration strategy in which the MVS cues are used before the shape reconstruction. On the other hand, we considered the reconstruction obtained with Implicit Shape Modeling as an initialization that is improved by a post-processing optimization which implements multi-view constraints. We quantitatively and qualitatively evaluated the approaches on two datasets and demonstrated that both significantly improve the vanilla Implicit Shape Modeling method from Chapter 3. In particular, the early integration strategy is more inclined to improve the reconstruction of the global shape by expanding its generalization capability while the late integration strategy is most efficient to reconstruct surface details. Additionally, both strategies support higher resolution images in contrast the vanilla Implicit Shape Modeling.

6.2 Limitations and Future Work

In this section, we discuss the limitations of the approaches proposed in this thesis and present some promising future directions. In Chapter 3, we demonstrated that the proposed method has a quite good generalization ability and even show results with real images coming from a capture platform. However, the latter still represents a controlled environment and the performance in an in-the-wild context will remain limited. This is inherent to the learning strategy that is based on multi-scale pixel-aligned features learned on a dataset that contains a finite set of human poses, clothing and accessories. If completely out-of-distribution data is used at test time, the results are likely to degrade. A first possible solution for this limitation would be to leverage additional information such as the multi-view constraints under the hypothe-

sis of redundancy in the few input images, as demonstrated in Chapter 5. A second option would involve the use of much more training data by combining several synthetic datasets such as Renderpeople and THuman2.0 and including much more variety in terms of camera configurations, number of input views and lighting conditions. Methods that can obtain animatable 3D avatars from images [68, 78] or meshes [27, 164, 185] could also be leveraged to dramatically increase the variety in terms of human poses by reposing the original meshes. Unfortunately, for the moment these methods struggle to model complex clothing deformations and are not able to obtain very high quality reposed meshes as required in our data-driven strategy. A third solution to consider would be to remove the need for 3D supervision such that the deep neural network can be trained with 2D images only. This will significantly increase the potential training data which should improve its generalization ability. In practice, appearance prediction combined with differentiable rendering is a promising strategy for that.

A second limitation is the relative high computation time needed by the method. Indeed, at test time, the implicit function is evaluated in a full 3D grid which involves a very large number of forward passes in the deep neural network and is time-consuming. The inference with a high resolution grid $512 \times 512 \times 512$ takes less than a minute but is not sufficient to respect real-time constraints. One option would be to consider a coarse-to-fine approach that identifies areas where the surface is probable and evaluate the implicit function only there.

Other future directions are also conceivable in order to improve the proposed approach. First, the learnable pipeline from Chapter 3 predicts occupancy probabilities for a set of input 3D points and only involves supervision at that occupancy level. The occupancy intermediate representation can then be transformed into a 3D mesh by using a surface reconstruction algorithm such as Marching Cubes [117]. Unfortunately, the latter is not differentiable which prevents the direct supervision with an explicit surface during training. Recently, a few works contributed to that problem such as [108] in which the authors developed a differentiable version of Marching Cubes and [156] in which a new differentiable method to extract 3D surface directly from deep signed distance functions is proposed. Both demonstrated more accurate and complete reconstructions on object datasets and would be worth considering in our context.

Another interesting future direction is the extension to a spatio-temporal representation to better deal with dynamic scenes. On one side, the temporal dimension can be leveraged to reduce the jitter that can appear in the frame-by-frame reconstruction. On the other side, by seeing more parts of the human during the sequence, more information can be efficiently integrated over time to improve the geometry of the reconstruction.

Finally, in the method proposed in Chapter 3, we mostly consider a scenario with a single human in the input images. However, in many situations, the humans are grouped together and interact with each other which brings in strong occlusion issues. To tackle this more challenging task, more work is required, in particular to identify and possibly segment the different humans and efficiently leverage recent multi-human datasets. Some approaches have

already started to explore this promising direction [134, 231].

The method proposed in Chapter 4 also includes some limitations and interesting future directions that are worth considering. The first limitation concerns the generalization ability of the learned photo-consistency measure that we considered. By learning the latter on real data, we expect to gain some robustness with respect to the sensor noise, illumination and camera configurations. Despite good results, this aspect can still be improved, in particular, with data augmentation strategies. Another possible direction is to replace the appearance matching module with a learnable appearance prediction module and optimize the pipeline with differentiable rendering. With this, our proposed method will look more like methods such as [129, 197] while keeping a different parameterization of the 3D space. A second aspect of the method that can be improved is the optimization time. The method optimizes the depth along rays for each pixel of each input image which can become inefficient with high resolution images or for complete scene reconstruction. Two main axes can be explored as a solution. First, we could introduce a coarse-to-fine strategy in terms of image resolution. As the computation time depends on resolution of the images and the number of sampled points along the viewing lines, a coarse reconstruction can be efficiently obtained from low resolution images and then serves as a guide to reduce the number of samples when optimizing at a higher resolution. A second option is to improve the efficiency of the learned photo-consistency network that is evaluated for all the samples on each viewing line of each image. Specific architectures for very fast inference could be considered, maybe by replacing the attention-based encoder which is relatively time-consuming. Finally, following the trend of inference-based reconstruction techniques that allow very fast inference, it would be interesting to see if the geometric consistency aspect of our method could be introduced in an end-to-end learnable pipeline.

The methods explored in Chapter 5, are built on top of the work from Chapter 3 and, in spite of an improved performance, they share similar limitations and possible future directions. In particular, even if the MVS cues that are leveraged in the early integration strategy improve significantly the generalization ability of the method and probably support an in-the-wild setting, the multi-scale features from Multi-View PIFu are still at the core of the pipeline and can hinder the performance. As discussed previously, building a much larger training dataset or alleviating the need for 3D ground truth during training are two promising directions for this problem.

For the late integration strategy, a mesh is currently used as a proxy to combine Implicit Shape Modeling and NeuS which requires a complete optimization of the latter. A better combination of the two methods is probably possible by using a discretized representation of the geometry, as used in [187, 219, 220] instead of implicitly encoding it in an MLP. This would allow the optimization to directly start from the reconstruction obtained with Implicit Shape Modeling and be more efficient. Another possible limitation of the late integration strategy is the trade-off between exploitation and exploration. More specifically, it corresponds to the confidence in the reconstruction of Implicit Shape Modeling and expresses how much we accept to move away from it. In practice, this aspect is involved at two levels. First, the improved

sampling strategy described in Section 5.2.2 includes two parameters that define how much the sampled rays can be outside the silhouettes and how far we consider sampled points around the initial reconstruction. The trade-off is also included in the final loss with the weight γ that defines how strong the deviations from the initial reconstruction are penalized. Currently, these parameters are fixed for the entire optimization and specified in advance, but it would be interesting to experiment if they can also be optimized or dynamically adjusted during the optimization.

A

Appendix of Chapter 3

A.1 Additional Visual Results

In this first appendix, we provide additional visual results for three experiments presented in Section 3.4.

A.2 Comparison With State of the Art

Qualitative visual comparisons between our proposed method, the considered baseline [162] as well as state-of-the-art single-view reconstruction method PIFuHD [163] are presented in Figure A.1. In particular, we note the improved global quality of the recovered accessories and the reduced level of noise in the reconstructions using our method. Moreover, sharper details on the faces and wrinkles on the clothes are recovered by our approach. Two difficult cases with less usual pose and thin structures are shown in the last two rows. Although our reconstructions contain some noise and missing parts, we can see a significant improvement compared to the other two methods.

A.3 Application to Real-world Data

A crucial aspect of our work is the applicability to real-world data. In Figure A.2 we provide additional comparisons between our method, the considered baseline PIFu [162], state-of-the-art monocular reconstruction method PIFuHD [163], and a 60-view reconstruction obtained with a Multi-view stereo strategy [103]. In this real context, we observe that our method behaves better than the baseline and single-view reconstruction, especially with complex scenes, *e.g.* with accessories. The improvement is less obvious, yet there, with persons in standard poses and without accessories (*e.g.* columns 3 in Figure A.2). In this case, the strategy from [162] already provides good results.

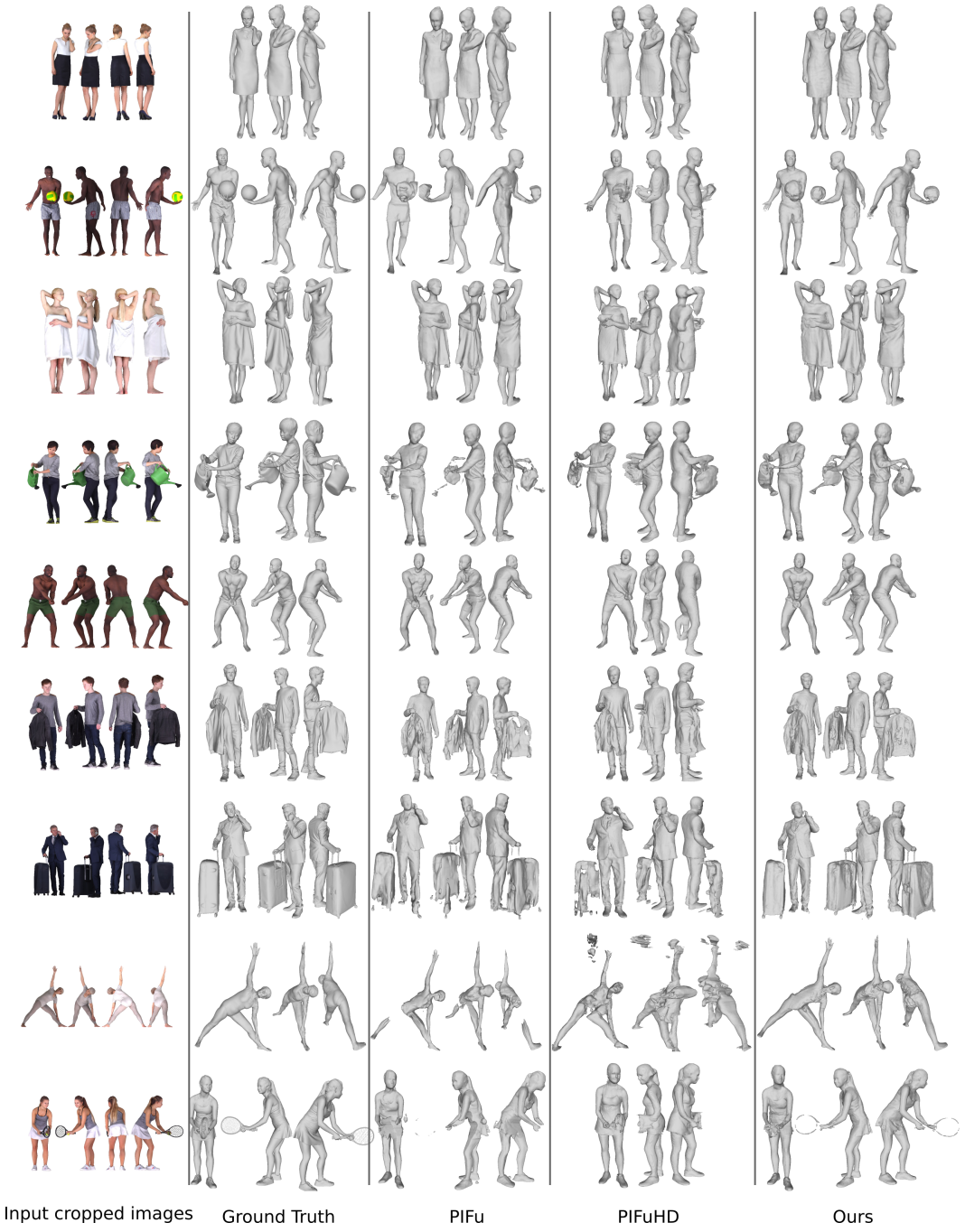


Figure A.1: Qualitative results and comparisons. PIFu [162] and our method take as input the 4 cropped images, whereas PIFuHD [163] receives only the frontal view. The 4 input images are rendered with the rotations around the vertical axis : 10°, 110°, 150°, 300°.

Another interesting comparison in this figure is with multi-view stereo (row b). While the MVS strategy provides robust and accurate estimations of the global shapes, our data-driven strategy yields more local details.

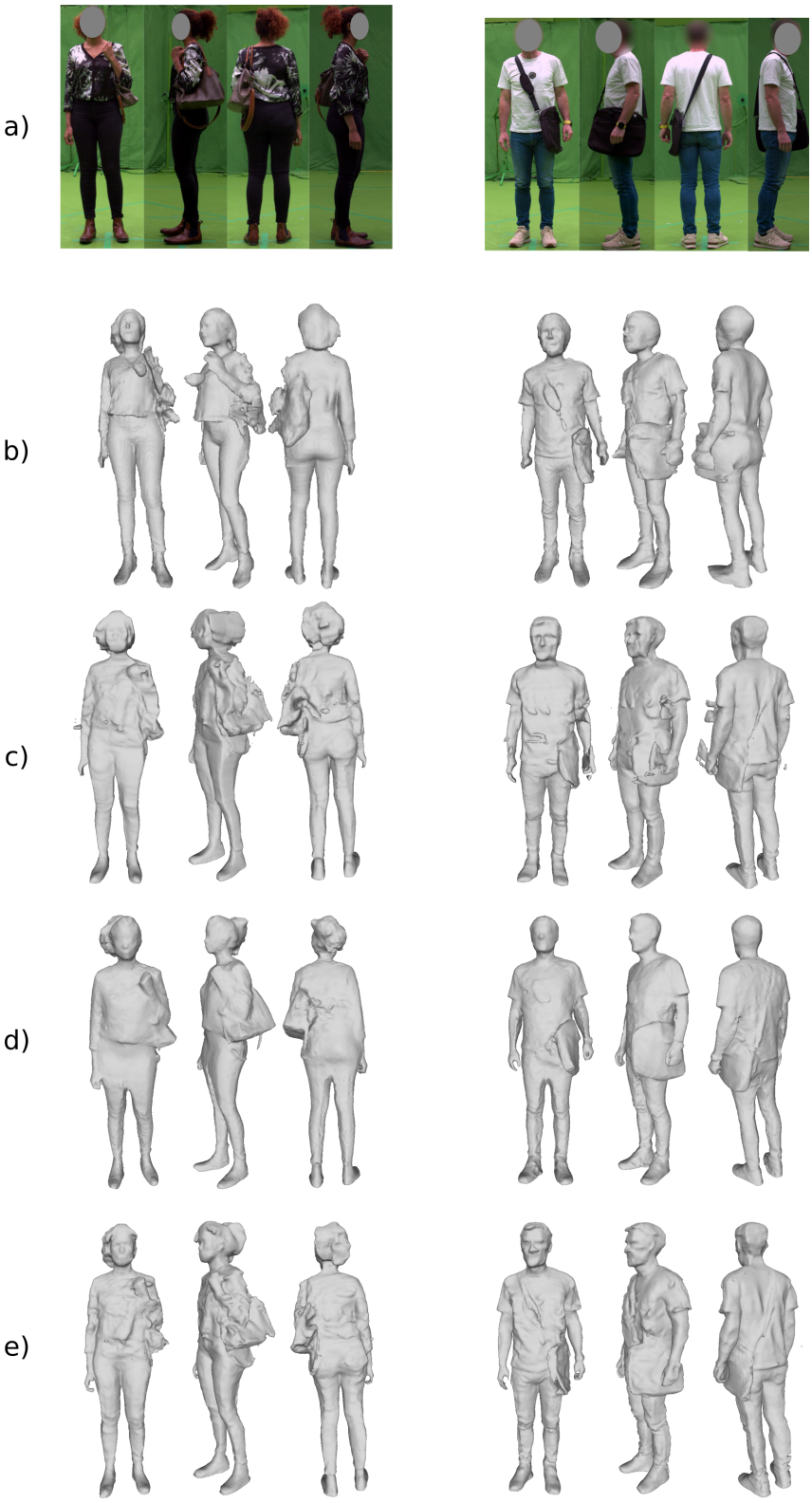


Figure A.2: Qualitative results and comparisons with a real capture apparatus: a) real RGB images. b) single frontal view reconstruction using PIFuHD [163]. c) 4-view reconstructions using PIFu [162]. d) 60-view reconstructions using a multi-view stereo approach [103]. e) 4-view reconstructions using our method.

A.4 Ablation Visual Results

Here, we show additional visual results of our ablation to evaluate the impact of our contributions. Quantitatively, disabling the view fusion or the context encoding module both affects the reconstruction performance. From the results shown in Figure A.3, we clearly see that the multi-head self-attention view fusion module is crucial for both the global quality and the local geometric details. On the other hand, the local 3D context encoding impacts more the global quality of the reconstruction and helps avoiding holes or missing parts.

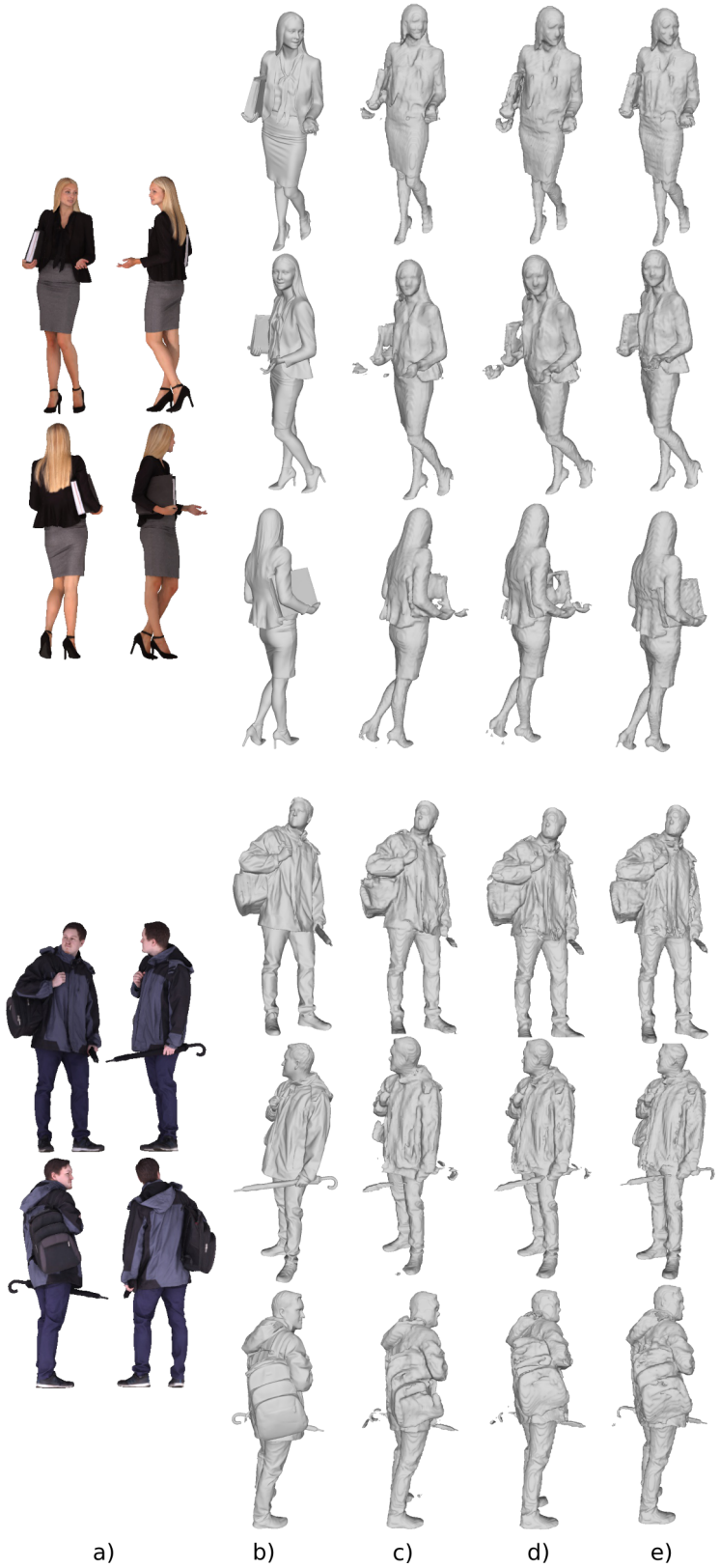


Figure A.3: Ablation studies of our approach: a) Input cropped images. The 4 input images are rendered with the rotations around the vertical axis : 0°, 90°, 180°, 270°. b) Ground truth models. c) Ours without the attention-based view fusion module. d) Ours without the local 3D context encoding. e) Our full method.

B

Appendix of Chapter 4

In the second appendix, we provide additional details and visual results for experiments presented in Section 4.4.

B.1 Multi-View Reconstruction From Real Data

In Table B.1 we show the detailed version of the Table 4.2 and in Figure B.1 we give additional qualitative visual comparisons between our proposed method and the considered baselines: COLMAP [167], ACMMP [204], IDR [217], Neus [197], NeuralWarp [34], PatchmatchNet [195] and CasMVSNet [60] on the DTU [82] dataset. The reconstruction settings are similar to the comparison in the Section 4.4.3.

Variants	IDR [217]			NeuS [197]			NeuralWarp [34]			COLMAP [167]			ACMMP [204]			PatchmatchNet [195]			CasMVSNet [60]			Ours		
	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg
Scan024	1.76	1.50	1.63	0.90	0.75	0.83	0.52	0.47	0.50	0.32	0.50	0.41	0.39	0.33	0.36	0.33	0.26	0.30	**0.29**	0.29	**0.29**	0.35	**0.25**	0.30
Scan037	2.16	1.55	1.86	1.09	0.88	0.98	0.80	0.61	0.70	0.57	0.66	0.62	0.66	0.44	0.55	0.56	0.45	**0.51**	**0.47**	0.58	0.52	0.61	**0.43**	0.52
Scan040	0.65	0.61	0.63	0.58	0.54	0.56	0.38	0.37	0.38	0.27	0.43	0.35	0.37	0.28	0.33	0.28	0.29	0.29	**0.24**	0.34	0.29	0.29	**0.25**	**0.27**
Scan055	0.57	0.37	0.47	0.40	0.34	0.37	0.40	0.37	0.39	0.25	0.44	0.35	0.26	0.27	0.27	0.27	**0.25**	**0.26**	0.32	0.39	0.36	**0.25**	0.26	**0.26**
Scan063	1.43	0.63	1.03	1.62	0.64	1.13	1.00	0.58	0.79	0.70	0.45	0.58	1.35	0.35	0.85	0.84	**0.26**	0.55	0.64	0.27	0.45	**0.45**	0.40	**0.43**
Scan065	0.88	0.69	0.78	0.68	**0.51**	0.59	0.80	0.82	0.81	0.32	1.60	0.96	0.32	0.72	0.52	0.34	0.98	0.66	**0.27**	1.42	0.84	0.50	**0.51**	**0.50**
Scan069	0.88	0.66	0.77	0.68	0.52	0.60	0.92	0.73	0.82	0.39	0.52	0.46	0.43	0.37	0.40	0.38	0.32	0.35	**0.31**	0.33	**0.32**	0.44	**0.27**	0.36
Scan083	1.10	1.55	1.32	1.33	1.57	1.45	0.85	1.55	1.20	0.48	0.62	0.55	0.47	0.56	0.51	0.57	**0.50**	0.54	0.36	0.51	**0.43**	**0.32**	1.02	0.67
Scan097	1.30	0.99	1.15	1.06	0.84	0.95	0.85	1.33	1.09	0.57	0.56	0.57	0.46	0.39	0.43	0.58	**0.31**	0.45	**0.42**	0.32	**0.37**	0.51	0.34	0.42
Scan105	-	-	0.64*	0.78	0.78	0.78	0.59	0.78	0.69	0.46	0.63	0.54	0.50	0.52	0.51	0.55	0.48	0.52	**0.33**	0.51	0.42	0.34	**0.27**	**0.31**
Scan106	0.73	0.60	0.66	0.53	0.52	0.52	0.57	0.78	0.67	0.29	0.57	0.43	0.32	**0.33**	0.32	0.31	0.34	0.33	**0.25**	0.40	0.33	**0.25**	0.34	**0.29**
Scan110	1.09	0.68	0.89	1.71	1.16	1.44	0.90	0.57	0.73	0.44	0.43	0.44	0.45	0.34	0.39	0.49	**0.20**	0.34	**0.34**	0.23	**0.29**	0.41	0.36	0.38
Scan114	0.45	0.38	0.41	0.34	0.38	0.36	0.42	0.41	0.41	0.26	0.36	0.31	0.26	0.27	0.27	0.39	**0.18**	0.29	**0.23**	0.19	**0.21**	0.26	0.20	0.23
Scan118	0.54	0.46	0.50	0.48	0.43	0.45	0.71	0.55	0.63	0.30	0.50	0.40	0.30	0.34	0.32	0.37	**0.25**	0.31	**0.28**	0.39	0.33	0.30	0.26	**0.28**
Scan122	0.72	0.43	0.57	0.57	0.41	0.49	0.55	0.46	0.50	0.30	0.45	0.37	0.30	0.31	0.31	0.34	**0.22**	0.28	0.26	0.34	0.30	**0.26**	**0.22**	**0.24**
Mean	1.02	0.79	0.89	0.85	0.68	0.77	0.68	0.69	0.69	0.40	0.58	0.49	0.46	0.39	0.42	0.44	**0.35**	0.40	**0.34**	0.43	0.38	0.37	0.36	**0.36**

*Table B.1: Quantitative evaluation on DTU [82] (49 or 64 images per model). Best scores are in **bold**. (* pre-trained model issue with Scan105, we report the IDR paper results).*

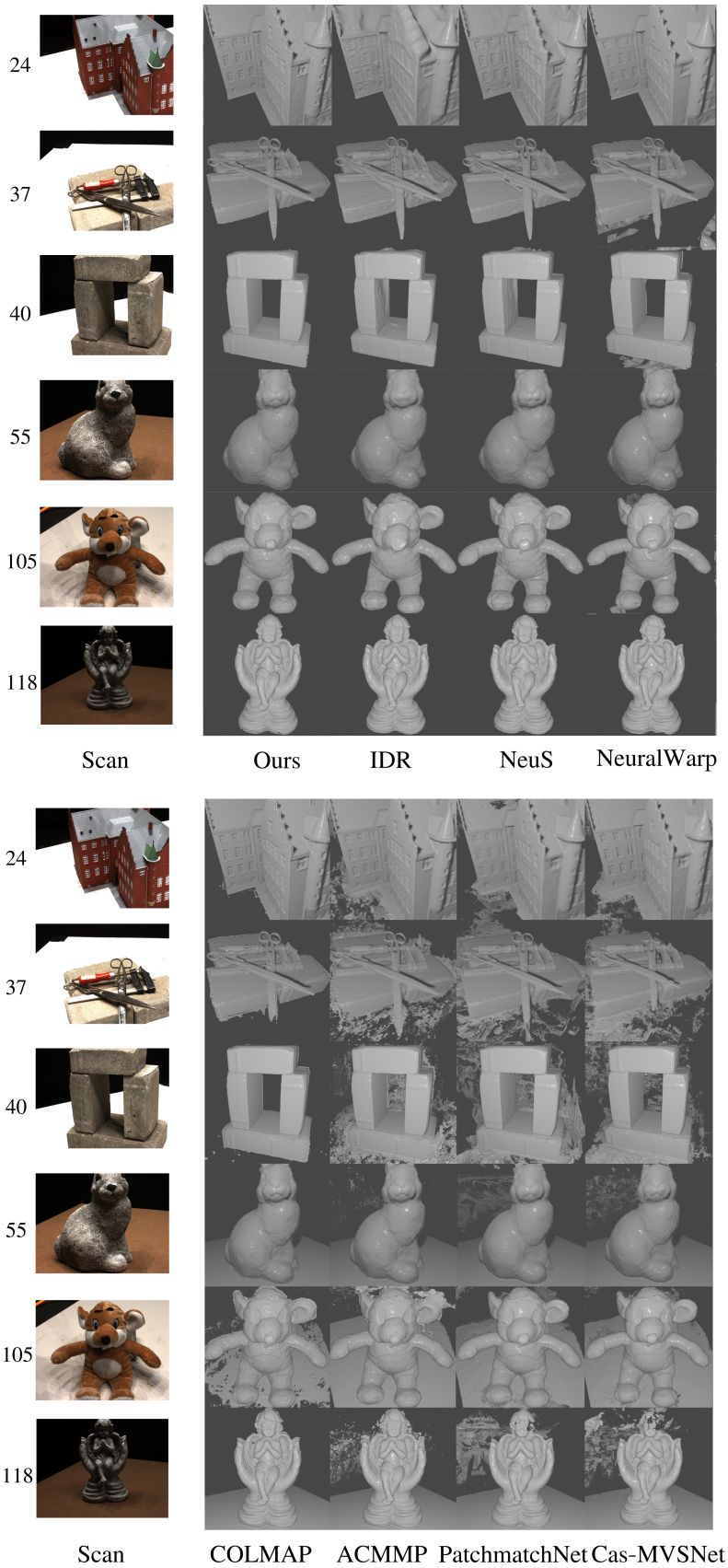


Figure B.1: Qualitative comparisons on DTU.

B.2 Multi-View Reconstruction From Synthetic Data

In Figure B.2, we provide additional visual comparisons between our method applied with the baseline photo-consistency prior defined in Section 4.2.3, COLMAP, ACMMP, IDR, NeuS, PatchmatchNet and CasMVSNet. We use 19 synthetic images rendered from the Renderpeople [2] meshes. The reconstruction settings are similar to the comparison in the Section 4.4.4. We can observe that our method is able to reconstruct very accurate and detailed meshes. Our results contain more details (*e.g.* faces, cloth wrinkles) and less noise than the other methods.

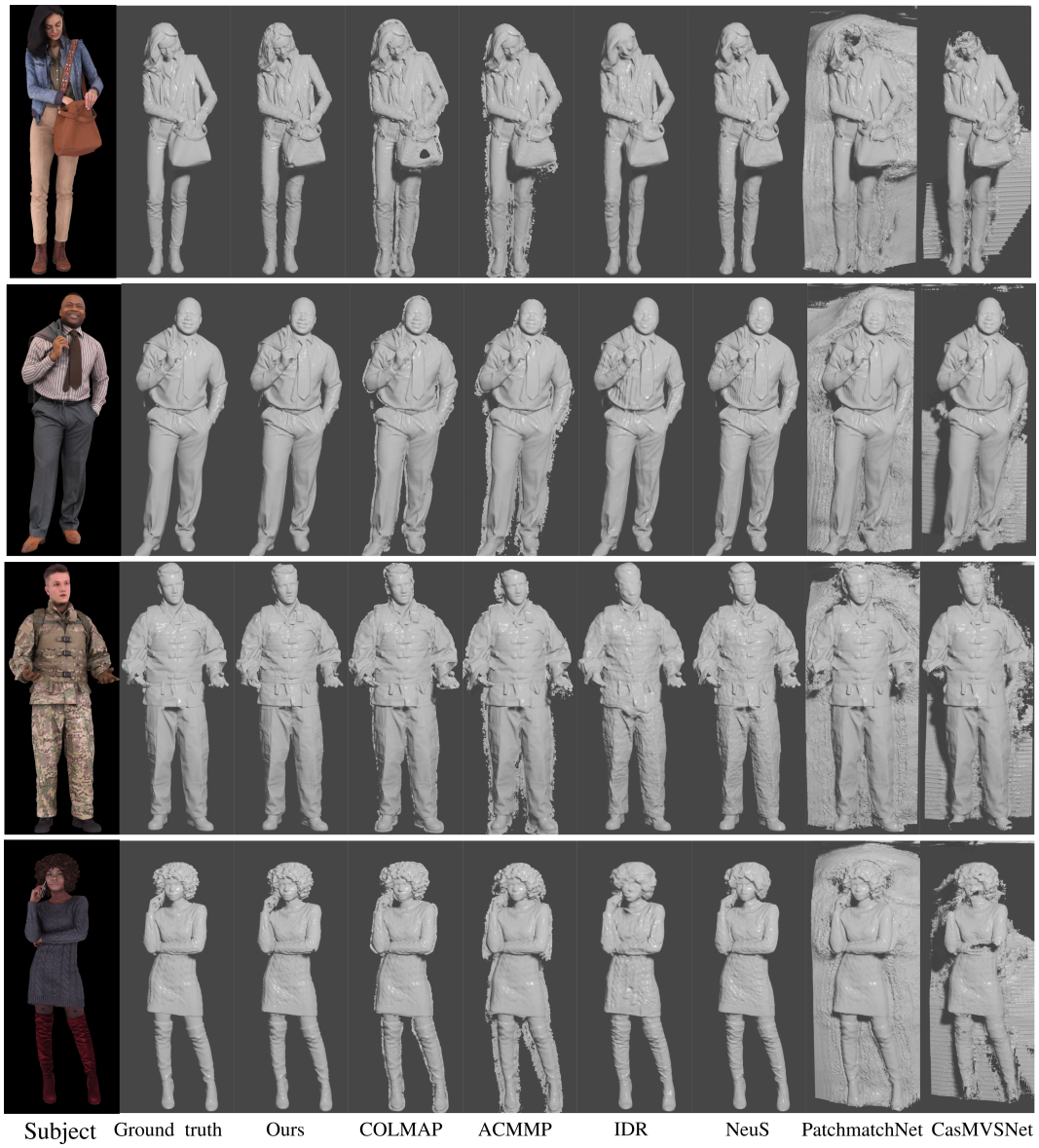


Figure B.2: Qualitative comparisons on Renderpeople.

B.3 Multi-View Reconstruction From Real Human Captured Data

In Figure B.3, we provide additional visual comparisons between our method and COLMAP, ACMMP, NeuS, PatchmatchNet and CasMVSNet. The reconstruction settings are similar to the comparison in the Section 4.4.5. We can observe that our method reconstructs detailed surfaces with limited noise even on some difficult parts as the black pants on the fourth column. COLMAP also performs quite well but has difficulties with the black bag, the pants and the hair. ACMMP is less precise but we mention that a single optimization iteration was used due to RAM's limitation, even with 64Gb. NeuS reconstructs a nice watertight surface but lacks high-frequency details and exhibits poor geometries at different locations due to appearance ambiguities. The deep MVS methods PatchmatchNet and CasMVSNet have much more difficulties reconstructing accurate surfaces. This illustrates the generalization issue with the full end-to-end learning-based methods when the inference scenario is substantially different from the training one (*i.e.*DTU).

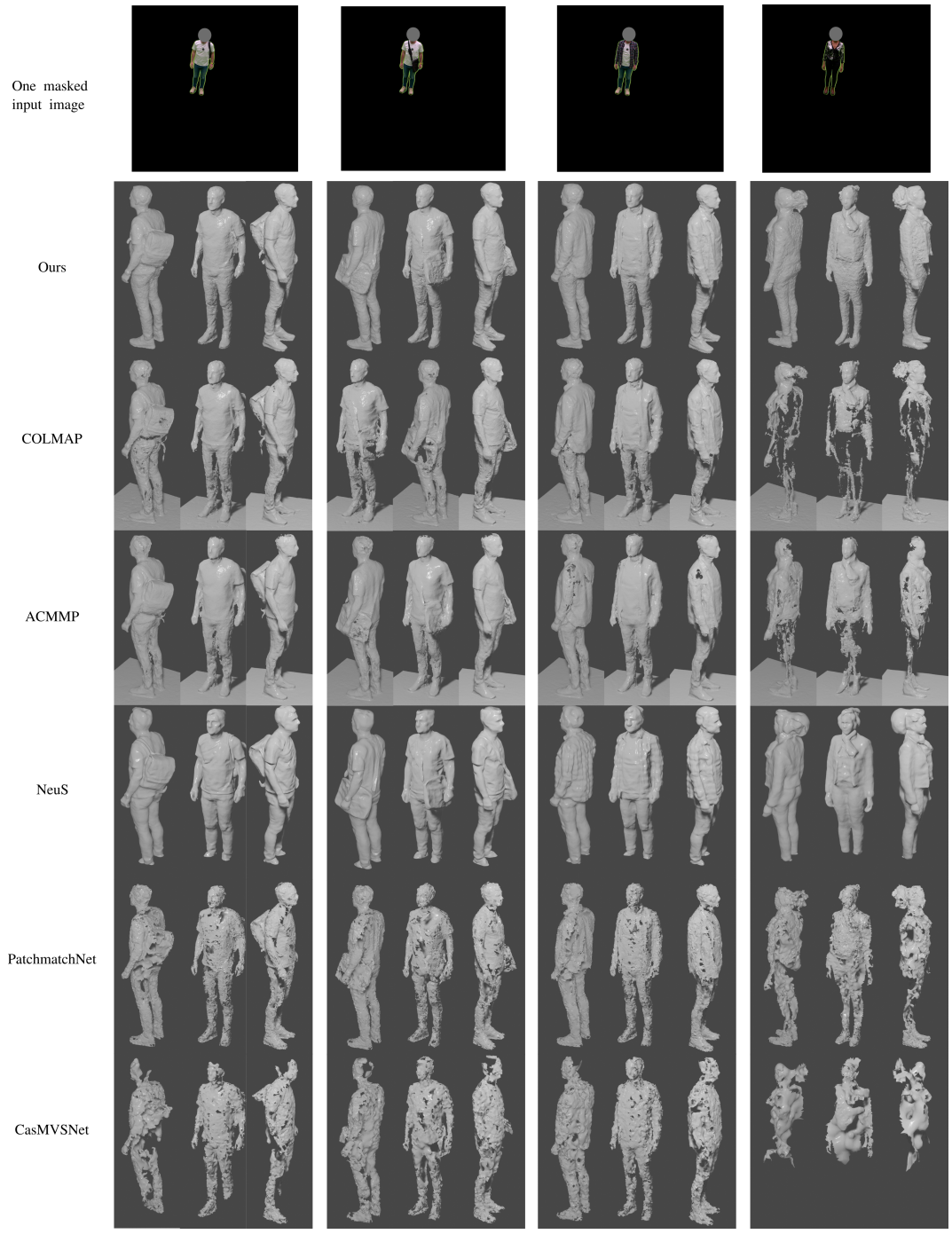


Figure B.3: Qualitative comparisons using 65 images from a multi-camera platform.

Bibliography

[1] Dtueval-python. https://github.com/jzhangbs/DTUeval-python.

[2] Renderpeople, 2018. https://renderpeople.com/3d-people/.

[3] Hassan Afzal, Djamila Aouada, David Font, Bruno Mirbach, and Björn Ottersten. Rgb-d multi-view system calibration for full 3d scene reconstruction. In *2014 22nd International Conference on Pattern Recognition*, pages 2459–2464. IEEE, 2014.

[4] Thiemo Alldieck, Marcus A. Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Detailed human avatars from monocular video. In *International Conference on 3D Vision*, pages 98–109, 2018.

[5] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2293–2303, 2019.

[6] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*, pages 408–416. 2005.

[7] Alexandru O Balan, Leonid Sigal, Michael J Black, James E Davis, and Horst W Haussecker. Detailed human shape and pose from images. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[8] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009.

[9] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021.

[10] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.

[11] Jon Louis Bentley. Multidimensional binary search trees in database applications. *IEEE Transactions on Software Engineering*, (4):333–340, 1979.

[12] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pages 1–11, 2011.

[13] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter V. Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: automatic estimation of 3d human pose and shape from a single image. In *European Conference on Computer Vision*, pages 561–578, 2016.

[14] Alexandre Boulch. Generalizing discrete convolutions for unstructured point clouds. 2019.

[15] Aljaz Bozic, Pablo Palafox, Justus Thies, Angela Dai, and Matthias Nießner. Transformerfusion: Monocular rgb scene reconstruction using transformers. *Advances in Neural Information Processing Systems*, 34: 1403–1414, 2021.

[16] Adrian Broadhurst, Tom W Drummond, and Roberto Cipolla. A probabilistic framework for space carving. In *IEEE International Conference on Computer Vision*, volume 1, pages 388–393. IEEE, 2001.

[17] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv preprint arXiv:1608.04236*, 2016.

[18] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[19] Ziyun Cai, Jungong Han, Li Liu, and Ling Shao. Rgb-d datasets using microsoft kinect or similar sensors: a survey. *Multimedia Tools and Applications*, 76(3):4313–4355, 2017.

[20] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

[21] Neill DF Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *European Conference on Computer Vision*, pages 766–779. Springer, 2008.

[22] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

[23] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021.

[24] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. *arXiv preprint arXiv:2203.09517*, 2022.

[25] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1538–1547, 2019.

[26] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Visibility-aware point-based multi-view stereo network. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3695–3708, 2020.

[27] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11594–11604, 2021.

[28] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2524–2534, 2020.

[29] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6968–6979, 2020.

[30] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision*, pages 628–644. Springer, 2016.

[31] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.

[32] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015.

[33] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Annual Conference on Computer Graphics and Interactive Techniques*, pages 303–312, 1996.

[34] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6260–6269, 2022.

[35] Jeremy S De Bonet and Paul Viola. Poxels: Probabilistic voxelized volume reconstruction. In *International Conference on Computer Vision (ICCV)*, pages 418–425, 1999.

[36] Amaël Delaunoy, Emmanuel Prados, Pau Gargallo I Piracés, Jean-Philippe Pons, and Peter Sturm. Minimizing the multi-view stereo reprojection error for triangular surface meshes. In *BMVC 2008-British Machine Vision Conference*, pages 1–10. BMVA, 2008.

[37] Frank Dellaert and Lin Yen-Chen. Neural volume rendering: Nerf and beyond. *arXiv preprint arXiv:2101.05204*, 2020.

[38] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022.

[39] Simon Donne and Andreas Geiger. Learning non-volumetric depth fusion using successive reprojections. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7634–7643, 2019.

[40] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (ToG)*, 35(4):1–13, 2016.

[41] S M Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, David P Reichert, Lars Buesing, Theophane Weber, Oriol Vinyals, Dan Rosenbaum, Neil Rabinowitz, Helen King, Chloe Hillier, Matt Botvinick, Daan Wierstra, Koray Kavukcuoglu, and Demis Hassabis. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018.

[42] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks. *IEEE Transactions on neural networks and learning systems*, 32(5):2075–2089, 2020.

[43] Gabriele Fanelli, Thibaut Weise, Juergen Gall, and Luc Van Gool. Real time head pose estimation from consumer depth cameras. In *Joint pattern recognition symposium*, pages 101–110. Springer, 2011.

[44] Olivier Faugeras and Renaud Keriven. Complete dense stereovision using level set methods. In *European conference on computer vision*, pages 379–393. Springer, 1998.

[45] Paolo Favaro and Stefano Soatto. A geometric approach to shape from defocus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):406–417, 2005.

[46] Yutong Feng, Yifan Feng, Haoxuan You, Xibin Zhao, and Yue Gao. Meshnet: Mesh neural network for 3d shape representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8279–8286, 2019.

[47] Michael Firman. Rgbd datasets: Past, present and future. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 19–31, 2016.

[48] Pascal Fua. From multiple stereo views to multiple 3-d surfaces. *International Journal of Computer Vision*, 24(1):19–35, 1997.

[49] Pascal Fua and Yvan G Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal of Computer Vision*, 16(1):35–56, 1995.

[50] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2009.

[51] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9 (1-2):1–148, 2015.

[52] Valentin Gabeur, Jean-Sébastien Franco, Xavier Martin, Cordelia Schmid, and Grégory Rogez. Moulding humans: Non-parametric 3d human shape estimation from single images. In *IEEE/CVF International Conference on Computer Vision*, pages 2232–2241, 2019.

[53] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *International Conference on 3D Vision*, pages 402–411. IEEE, 2017.

[54] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Gipuma: Massively parallel multi-view stereo reconstruction. *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e. V*, 25(361-369):2, 2016.

[55] Kyle Gao, Yina Gao, Hongjie He, Denning Lu, Linlin Xu, and Jonathan Li. Nerf: Neural radiance field in 3d vision, a comprehensive review. *arXiv preprint arXiv:2210.00379*, 2022.

[56] Andrew Gardner, Jinko Kanno, Christian A. Duncan, and Rastko R. Selmic. Classifying unordered feature sets with convolutional deep averaging networks. *CoRR*, abs/1709.03019, 2017.

[57] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *IEEE/CVF International Conference on Computer Vision*, pages 7154–7164, 2019.

[58] Andrew Gilbert, Marco Volino, John Collomosse, and Adrian Hilton. Volumetric performance capture from minimal camera viewpoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 566–581, 2018.

[59] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020.

[60] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020.

[61] Peng Guan, Alexander Weiss, Alexandru O Balan, and Michael J Black. Estimating human shape and pose from a single image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1381–1388. IEEE, 2009.

[62] Yuan-Chen Guo, Di Kang, Linchao Bao, Yu He, and Song-Hai Zhang. Nerfren: Neural radiance fields with reflections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18409–18418, 2022.

[63] Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions On Graphics (TOG)*, 38(2): 1–17, 2019.

[64] Marc Habermann, Weipeng Xu, Michael Zollhofer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5052–5063, 2020.

[65] Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. Meshcnn: a network with an edge. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

[66] Wilfried Hartmann, Silvano Galliani, Michal Havlena, Luc Van Gool, and Konrad Schindler. Learned multi-patch similarity. In *IEEE International Conference on Computer Vision*, pages 1586–1594, 2017.

[67] Tong He, John P. Collomosse, Hailin Jin, and Stefano Soatto. Geo-pifu: Geometry and pixel aligned implicit functions for single-view human reconstruction. In *Annual Conference on Neural Information Processing Systems*, pages 9276–9287, 2020.

[68] Tong He, Yuanlu Xu, Shunsuke Saito, Stefano Soatto, and Tony Tung. Arch++: Animation-ready clothed human reconstruction revisited. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11046–11056, 2021.

[69] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall, Jonathan T Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5875–5884, 2021.

[70] Paul Henderson and Vittorio Ferrari. Learning to generate and reconstruct 3d meshes with only 2d supervision. *arXiv preprint arXiv:1807.09259*, 2018.

[71] Vu Hoang Hiep, Renaud Keriven, Patrick Labatut, and Jean-Philippe Pons. Towards high-resolution large-scale multi-view stereo. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1430–1437. IEEE, 2009.

[72] Heiko Hirschmuller and Daniel Scharstein. Evaluation of stereo matching costs on images with radiometric differences. *IEEE transactions on pattern analysis and machine intelligence*, 31(9):1582–1599, 2008.

[73] Yang Hong, Juyong Zhang, Boyi Jiang, Yudong Guo, Ligang Liu, and Hujun Bao. Stereopifu: Depth aware clothed human digitization via stereo vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 535–545, 2021.

[74] Berthold KP Horn and Michael J Brooks. The variational approach to shape from shading. *Computer Vision, Graphics, and Image Processing*, 33(2):174–208, 1986.

[75] Binh-Son Hua, Minh-Khoi Tran, and Sai-Kit Yeung. Pointwise convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 984–993, 2018.

[76] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, 2018.

[77] Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Linjie Luo, Chongyang Ma, and Hao Li. Deep volumetric video from very sparse multi-view performance capture. In *European Conference on Computer Vision*, pages 351–369, 2018.

[78] Zeng Huang, Yuanlu Xu, Christoph Lassner, Hao Li, and Tony Tung. Arch: Animatable reconstruction of clothed humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3102, 2020.

[79] Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo. *arXiv preprint arXiv:1905.00538*, 2019.

[80] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. *Advances in Neural Information Processing Systems*, 31, 2018.

[81] Arjun Jain, Thorsten Thormählen, Hans-Peter Seidel, and Christian Theobalt. Moviereshape: Tracking and reshaping of humans in videos. *ACM Transactions on Graphics (TOG)*, 29(6):1–10, 2010.

[82] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014.

[83] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2307–2315, 2017.

[84] Li Jiang, Shaoshuai Shi, Xiaojuan Qi, and Jiaya Jia. Gal: Geometric adversarial loss for single-view 3d-object reconstruction. In *European Conference on Computer Vision*, pages 802–816, 2018.

[85] Wen Jiang, Nikos Kolotouros, Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Coherent reconstruction of multiple humans from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2020.

[86] Danilo Jimenez Rezende, SM Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. *Advances in Neural Information Processing Systems*, 29, 2016.

[87] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022.

[88] Shanon X Ju, Michael J Black, and Yaser Yacoob. Cardboard people: A parameterized model of articulated image motion. In *Proceedings of the second international conference on automatic face and gesture recognition*, pages 38–44. IEEE, 1996.

[89] Kristine Aavild Juhl, Xabier Morales, Ole de Backer, Oscar Camara, and Rasmus Reinhold Paulsen. Implicit neural distance representation for unsupervised and supervised classification of complex anatomies. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 405–415. Springer, 2021.

[90] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018.

[91] Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. *Advances in neural information processing systems*, 30, 2017.

[92] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3907–3916, 2018.

[93] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013.

[94] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Eurographics Symposium on Geometry Processing*, volume 7, 2006.

[95] Petr Kellnhofer, Lars C Jebe, Andrew Jones, Ryan Spicer, Kari Pulli, and Gordon Wetzstein. Neural lumigraph rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4287–4297, 2021.

[96] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4501–4510, 2019.

[97] Alexander Krull, Eric Brachmann, Frank Michel, Michael Ying Yang, Stefan Gumhold, and Carsten Rother. Learning analysis-by-synthesis for 6d pose estimation in rgb-d images. In *Proceedings of the IEEE international conference on computer vision*, pages 954–962, 2015.

[98] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[99] Shiyi Lan, Ruichi Yu, Gang Yu, and Larry S Davis. Modeling local geometric structure of 3d point clouds using geo-cnn. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 998–1008, 2019.

[100] Aldo Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on pattern analysis and machine intelligence*, 16(2):150–162, 1994.

[101] Huan Lei, Naveed Akhtar, and Ajmal Mian. Octree guided cnn with spherical kernels for 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9631–9640, 2019.

[102] Vincent Leroy, Jean-Sébastien Franco, and Edmond Boyer. Multi-view dynamic shape refinement using local temporal integration. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3094–3103, 2017.

[103] Vincent Leroy, Jean-Sébastien Franco, and Edmond Boyer. Shape reconstruction using volume sweeping and learned photoconsistency. In *European Conference on Computer Vision*, pages 796–811, 2018.

[104] Vincent Leroy, Jean-Sébastien Franco, and Edmond Boyer. Shape reconstruction using volume sweeping and learned photoconsistency. In *European Conference on Computer Vision*, pages 781–796, 2018.

[105] Ruilong Li, Yuliang Xiu, Shunsuke Saito, Zeng Huang, Kyle Olszewski, and Hao Li. Monocular real-time volumetric performance capture. In *European Conference on Computer Vision*, pages 49–67. Springer, 2020.

[106] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017.

[107] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, and Zhaoyang Lv. Neural 3d video synthesis. *arXiv preprint arXiv:2103.02597*, 2021.

[108] Yiyi Liao, Simon Donne, and Andreas Geiger. Deep marching cubes: Learning explicit surface representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2916–2925, 2018.

[109] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[110] Jin Liu and Shunping Ji. A novel recurrent encoder-decoder structure for large-scale multi-view stereo reconstruction from an open aerial dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6050–6059, 2020.

[111] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2019–2028, 2020.

[112] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2019.

[113] Yongcheng Liu, Bin Fan, Gaofeng Meng, Jiwen Lu, Shiming Xiang, and Chunhong Pan. Densepoint: Learning densely contextual representation for efficient point cloud processing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5239–5248, 2019.

[114] Yongcheng Liu, Bin Fan, Shiming Xiang, and Chunhong Pan. Relation-shape convolutional neural network for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8895–8904, 2019.

[115] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):248:1–248:16, 2015.

[116] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.

[117] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, pages 163–169, 1987.

[118] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[119] Keyang Luo, Tao Guan, Lili Ju, Haipeng Huang, and Yawei Luo. P-mvsnet: Learning patch-wise matching confidence aggregation for multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10452–10461, 2019.

[120] Keyang Luo, Tao Guan, Lili Ju, Yuesong Wang, Zhuo Chen, and Yawei Luo. Attention-aware multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1590–1599, 2020.

[121] Wenjie Luo, Alexander G Schwing, and Raquel Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5695–5703, 2016.

[122] David Marr and Herbert Keith Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 200(1140): 269–294, 1978.

[123] David Marr and Tomaso Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 204(1156):301–328, 1979.

[124] Donald Meagher. Geometric modeling using octree encoding. *Computer graphics and image processing*, 19(2):129–147, 1982.

[125] Paul Merrell, Amir Akbarzadeh, Liang Wang, Philippos Mordohai, Jan-Michael Frahm, Ruigang Yang, David Nistér, and Marc Pollefeys. Real-time visibility-based fusion of depth maps. In *IEEE International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[126] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[127] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *IEEE/CVF International Conference on Computer Vision*, pages 4743–4752, 2019.

[128] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[129] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020.

[130] Masahiro Mori. The uncanny valley: the original essay by masahiro mori. *IEEE Spectrum*, 1970.

[131] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022.

[132] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *European Conference on Computer Vision*, pages 414–431. Springer, 2020.

[133] Armin Mustafa, Akin Caliskan, Lourdes Agapito, and Adrian Hilton. Multi-person implicit reconstruction from a single image. *CoRR*, abs/2104.09283, 2021. URL https://arxiv.org/abs/2104.09283.

[134] Armin Mustafa, Akin Caliskan, Lourdes Agapito, and Adrian Hilton. Multi-person implicit reconstruction from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14474–14483, 2021.

[135] KL Navaneet, Priyanka Mandikal, Mayank Agarwal, and R Venkatesh Babu. Capnet: Continuous approximation projection for 3d point cloud reconstruction using 2d supervision. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 8819–8826, 2019.

[136] Erdogmus Nesli and Sébastien Marcel. Spoofing in 2d face recognition with 3d masks and anti-spoofing with kinect. In *IEEE 6th International Conference on Biometrics: Theory, Applications and Systems (BTAS'13)*, pages 1–8, 2013.

[137] Ramakant Nevatia and Thomas O Binford. Description and recognition of curved objects. *Artificial intelligence*, 8(1):77–98, 1977.

[138] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499, 2016.

[139] Thu H Nguyen-Phuoc, Chuan Li, Stephen Balaban, and Yongliang Yang. Rendernet: A deep convolutional network for differentiable rendering from 3d shapes. *Advances in Neural Information Processing Systems*, 31, 2018.

[140] Michael Niemeyer, Lars M. Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. *CoRR*, abs/1912.07372, 2019.

[141] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *IEEE/CVF International Conference on Computer Vision*, pages 4531–4540, 2019.

[142] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021.

[143] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[144] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.

[145] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021.

[146] N Passalis, S Pedrazzi, R Babuska, W Burgard, D Dias, F Ferro, M Gabbouj, O Green, A Iosifidis, E Kayacan, et al. Opendr: An open toolkit for enabling high performance, low footprint deep learning for robotics. *arXiv preprint arXiv:2203.00403*, 2022.

[147] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[148] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

[149] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 459–468, 2018.

[150] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019.

[151] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020.

[152] Jean-Philippe Pons, Renaud Keriven, and Olivier Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *International Journal of Computer Vision*, 72(2): 179–193, 2007.

[153] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021.

[154] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[155] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In

IEEE/CVF International Conference on Computer Vision, pages 14335–14345, 2021.

[156] Edoardo Remelli, Artem Lukoianov, Stephan Richter, Benoît Guillard, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Meshsdf: Differentiable iso-surface extraction. *Advances in Neural Information Processing Systems*, 33:22468–22478, 2020.

[157] Gernot Riegler, Ali Osman Ulusoy, Horst Bischof, and Andreas Geiger. Octnetfusion: Learning depth fusion from data. In *International Conference on 3D Vision*, pages 57–66. IEEE, 2017.

[158] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12892–12901, 2022.

[159] Karl Rohr. Towards model-based recognition of human movements in image sequences. *CVGIP: Image understanding*, 59(1):94–115, 1994.

[160] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022.

[161] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[162] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *IEEE/CVF International Conference on Computer Vision*, pages 2304–2314, 2019.

[163] Shunsuke Saito, Tomas Simon, Jason M. Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 81–90, 2020.

[164] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J Black. Scanimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2886–2897, 2021.

[165] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb. Kinect range sensing: Structured-light versus time-of-flight kinect. *Computer vision and image understanding*, 139:1–20, 2015.

[166] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[167] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016.

[168] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999.

[169] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 519–528, 2006.

[170] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *IEEE Conference on Computer Vision and Pattern Recognition*, 1:519–528, 2006.

[171] Hedvig Sidenbladh, Michael J Black, and David J Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *European conference on computer vision*, pages 702–718. Springer, 2000.

[172] Leonid Sigal, Alexandru O Balan, and Michael J Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International journal of computer vision*, 87(1):4–27, 2010.

[173] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.1556.

[174] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019.

[175] David Smith, Matthew Loper, Xiaochen Hu, Paris Mavroidis, and Javier Romero. FACSIMILE: fast and accurate scans from an image in less than a second. In *IEEE/CVF International Conference on Computer Vision*, pages 5329–5338, 2019.

[176] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[177] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *IEEE/CVF International Conference on Computer Vision*, pages 945–953, 2015.

[178] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5459–5469, 2022.

[179] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021.

[180] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. *CoRR*, abs/1902.09212, 2019. URL http://arxiv.org/abs/1902.09212.

[181] Xin Suo, Yuheng Jiang, Pei Lin, Yingliang Zhang, Minye Wu, Kaiwen Guo, and Lan Xu. Neuralhumanfvv: Real-time neural volumetric human performance rendering using rgb cameras. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6226–6237, 2021.

[182] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11358–11367, 2021.

[183] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE international conference on computer vision*, pages 2088–2096, 2017.

[184] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, W Yifan, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735. Wiley Online Library, 2022.

[185] Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Neural-gif: Neural generalized implicit functions for animating people in clothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11708–11718, 2021.

[186] Engin Tola, Vincent Lepetit, and Pascal Fua. A fast local descriptor for dense matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[187] Briac Toussaint, Maxime Genisson, and Jean-Sébastien Franco. Fast gradient descent for surface capture via differentiable rendering. In *3DV 2022-International Conference on 3D Vision*, 2022.

[188] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2626–2634, 2017.

[189] Gül Varol, Duygu Ceylan, Bryan C. Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *European Conference on Computer Vision*, pages 20–38, 2018.

[190] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[191] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[192] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022.

[193] Stefan Wachter and H-H Nagel. Tracking persons in monocular image sequences. *Computer Vision and Image Understanding*, 74(3):174–192, 1999.

[194] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3343–3352, 2019.

[195] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021.

[196] Min Wang, Feng Qiu, Wentao Liu, Chen Qian, Xiaowei Zhou, and Lizhuang Ma. Monocular human pose and shape reconstruction using

part differentiable rendering. In *Computer Graphics Forum*, volume 39, pages 351–362. Wiley Online Library, 2020.

[197] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021.

[198] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5610–5619, 2021.

[199] Simon AJ Winder and Matthew Brown. Learning local image descriptors. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[200] Zhaoxia Xiao and Wenming Huang. Kd-tree based nonuniform simplification of 3d point cloud. In *2009 Third International Conference on Genetic and Evolutionary Computing*, pages 339–342. IEEE, 2009.

[201] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. In *Computer Graphics Forum*, volume 41, pages 641–676. Wiley Online Library, 2022.

[202] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *Advances in Neural Information Processing Systems*, 32, 2019.

[203] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022.

[204] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5483–5492, 2019.

[205] Qingshan Xu and Wenbing Tao. Planar prior assisted patchmatch multi-view stereo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12516–12523, 2020.

[206] Qingshan Xu and Wenbing Tao. Pvsnet: Pixelwise visibility-aware multi-view stereo network. *arXiv preprint arXiv:2007.07714*, 2020.

[207] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018.

[208] Youze Xue, Jiansheng Chen, Weitao Wan, Yiqing Huang, Cheng Yu, Tianpeng Li, and Jiayu Bao. Mvscrf: Learning multi-view stereo with conditional random fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4312–4321, 2019.

[209] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *European conference on computer vision*, pages 674–689. Springer, 2020.

[210] Bo Yang, Sen Wang, Andrew Markham, and Niki Trigoni. Robust attentional aggregation of deep feature sets for multi-view 3d reconstruction. *International Journal on Computer Vision*, 128(1):53–73, 2020.

[211] Hsiao-Chien Yang, Po-Heng Chen, Kuan-Wen Chen, Chen-Yi Lee, and Yong-Sheng Chen. Fade: Feature aggregation for depth estimation with multi-view stereo. *IEEE Transactions on Image Processing*, 29:6590–6600, 2020.

[212] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020.

[213] Ze Yang, Shenlong Wang, Sivabalan Manivasagam, Zeng Huang, Wei-Chiu Ma, Xinchen Yan, Ersin Yumer, and Raquel Urtasun. S3: Neural shape, skeleton, and skinning fields for 3d human modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13284–13293, 2021.

[214] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, pages 767–783, 2018.

[215] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE/CVF IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[216] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020.

[217] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[218] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. volume 34, 2021.

[219] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. *arXiv preprint arXiv:2112.05131*, 2021.

[220] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *IEEE/CVF International Conference on Computer Vision*, pages 5752–5761, 2021.

[221] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021.

[222] Anzhu Yu, Wenyue Guo, Bing Liu, Xin Chen, Xin Wang, Xuefeng Cao, and Bingchuan Jiang. Attention aware cost volume pyramid based multi-view stereo network for 3d reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175:448–460, 2021.

[223] Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2021)*, June 2021.

[224] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4353–4361, 2015.

[225] Jure Zbontar, Yann LeCun, et al. Stereo matching by training a convolutional neural network to compare image patches. *J. Mach. Learn. Res.*, 17(1):2287–2318, 2016.

[226] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *arXiv preprint arXiv:2008.07928*, 2020.

[227] Xudong Zhang, Yutao Hu, Haochen Wang, Xianbin Cao, and Baochang Zhang. Long-range attention network for multi-view stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3782–3791, 2021.

[228] Youmin Zhang, Yimin Chen, Xiao Bai, Suihanjin Yu, Kun Yu, Zhiwei Li, and Kuiyuan Yang. Adaptive unimodal cost volume filtering for deep stereo matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12926–12934, 2020.

[229] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012.

[230] Enliang Zheng, Enrique Dunn, Vladimir Jojic, and Jan-Michael Frahm. Patchmatch based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517, 2014.

[231] Yang Zheng, Ruizhi Shao, Yuxiang Zhang, Tao Yu, Zerong Zheng, Qionghai Dai, and Yebin Liu. Deepmulticap: Performance capture of multiple characters using sparse multiview cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6239–6249, 2021.

[232] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *IEEE/CVF International Conference on Computer Vision*, pages 7738–7748, 2019.

[233] Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *IEEE Transsactions on Pattern Analysis and Machine Intelligence*, to appear, 2021.

[234] Boyao Zhou, Jean-Sébastien Franco, Federica Bogo, and Edmond Boyer. Spatio-temporal human shape completion with implicit function networks. In *International Conference on 3D Vision, 3DV 2021, London, United Kingdom, December 1-3, 2021*, pages 669–678. IEEE, 2021. doi: 10.1109/3DV53792.2021.00076. URL https://doi.org/10.1109/3DV53792.2021.00076.

[235] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Josh Tenenbaum, and Bill Freeman. Visual object networks: Image generation with disentangled 3d representations. *Advances in Neural Information Processing Systems*, 31, 2018.

[236] Luyang Zhu, Konstantinos Rematas, Brian Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Reconstructing nba players. In *European conference on computer vision*, pages 177–194. Springer, 2020.

[237] Pierre Zins, Yuanlu Xu, Edmond Boyer, Stefanie Wuhrer, and Tony Tung. Data-driven 3d reconstruction of dressed humans from sparse views. In *2021 International Conference on 3D Vision (3DV)*, pages 494–504. IEEE, 2021.

[238] Pierre Zins, Yuanlu Xu, Edmond Boyer, Stefanie Wuhrer, and Tony Tung. Multi-view reconstruction using signed ray distance functions (srdf). *arXiv preprint arXiv:2209.00082*, 2022.

[239] Ehsan Zobeidi and Nikolay Atanasov. A deep signed directional distance function for object shape representation. *arXiv preprint arXiv:2107.11024*, 2021.

[240] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (ToG)*, 33(4):1–12, 2014.