Fast and Accurate 4D Modeling of Large Multi-Camera Sequences

Vincent Leroy

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

Spécialité : Mathématiques Appliquées

Arrêté ministériel : 25 mai 2016

Présentée par

Vincent LEROY

Thèse dirigée par **Edmond BOYER**
et codirigée par **Jean-Sébastien FRANCO**, INRIA

préparée au sein du **Laboratoire Jean Kuntzmann** dans **l'École Doctorale Mathématiques, Sciences et technologies de l'information, Informatique**

Modélisation 4D rapide et précise de larges séquences multi-caméras

Fast and Accurate 4D Modeling of Large Multi-Camera Sequences

Thèse soutenue publiquement le **17 octobre 2019**,
devant le jury composé de :

Monsieur EDMOND BOYER
DIRECTEUR DE RECHERCHE, INRIA CENTRE DE GRENOBLE RHÔNE-ALPES, Directeur de thèse
Monsieur YASUTAKA FURUKAWA
PROFESSEUR ASSOCIE, UNIVERSITE SIMON FRASER-BURNABY - CANADA, Rapporteur
Monsieur GEORGE VOGIATZIS
PROFESSEUR ASSOCIE, UNIV. ASTON A BIRMINGHAM - ROYAUME-UNI, Rapporteur
Madame FLORENCE BERTAILS-DESCOUBES
DIRECTRICE DE RECHERCHE, INRIA CENTRE DE GRENOBLE RHÔNE-ALPES, Président
Monsieur RENAUD KERIVEN
PROFESSEUR ASSISTANT, ECOLE POLYTECHNIQUE DE PALAISEAU, Examinateur
Monsieur JEAN-SEBASTIEN FRANCO
MAITRE DE CONFERENCES, GRENOBLE INP, Examinateur

Abstract

Recent advances in acquisition and processing technologies have lead to the fast growth of a major branch in media production: volumetric video. In particular, the rise of virtual and augmented reality fuels an increased need for content suitable to these new media including 3D contents obtained from real scenes, since the ability to record a live performance and replay it from any given point of view allows the user to experience a realistic and truly immersive environment. This manuscript aims at presenting the problem of 4D shape reconstruction from multi-view RGB images, which is one way to create such content. We especially target real life performance capture, containing complex surface details. Typical challenges for these capture situations include smaller visual projection areas of objects of interest due to wider necessary fields of view for capturing motion; occlusion and self-occlusion of several subjects interacting together; lack of texture content typical of real-life subject appearance and clothing; or motion blur with fast moving subjects such as sport action scenes. An essential and still improvable aspect in this matter is the fidelity and quality of the recovered shapes, our goal in this work.

We present a full reconstruction pipeline suited for this scenario, to which we contributed in many aspects. First, Multi-view stereo (MVS) based methods have attained a good level of quality with pipelines that typically comprise feature extraction, matching stages and 3D shape inference. Interestingly, very recent works have re-examined stereo and MVS by introducing features and similarity functions automatically inferred using deep learning, the main promise of this type of method being to include better data-driven priors. We examine in a first contribution whether these improvements transfer to the more general and complex case of live performance capture, where a diverse set of additional difficulties arise. We then explain how to use this learning strategy to robustly build a shape representation, from which can be extracted a 3D model. Once we obtain this representation at every frame of the captured sequence, we discuss how to exploit temporal redundancy for precision refinement by propagating shape details through adjacent frames. In addition to being beneficial to many dynamic multi-view scenarios this also enables larger scenes where such increased precision can compensate for the reduced spatial resolution per image frame. The source code implementing the different reconstruction methods is released to the community at http://deep4dcvtr.gforge.inria.fr/.

Résumé

Les récentes avancées technologiques dans le domaine de l'acquisition et du calcul ont permis une croissance rapide d'une branche de production de média: la capture volumétrique. En particulier, la montée en puissance de la réalité virtuelle et augmentée engendre un besoin accru de contenus adaptés à ces nouveaux médias, notamment des contenus 3D obtenus à partir de scènes réelles. En effet, la possibilité d'enregistrer une performance et de la rejouer sous n'importe quel point de vue permet de créer une expérience dans un environnement réaliste et immersif pour l'utilisateur. Ce manuscrit présente le problème de la reconstruction de forme 4D à partir d'images RVB multi-vues, qui est une des stratégies permettant de créer un tel contenu. Nous nous intéressons particulièrement la capture de performances dynamiques en situations réelles, contenant des détails de surface complexes. Les défis typiques de ces situations de capture incluent une plus faible densité d'observation des objets d'intérêt en raison des champs de vision plus larges nécessaires pour capturer le mouvement; des occultations et auto-occultations de plusieurs sujets en interaction; un manque de texture typique de l'apparence et des vêtements du sujet réel; ou du flou de bougé avec des sujets en mouvement rapide tels que des scènes d'action sportive. Un aspect essentiel et qui peut encore être amélioré à cet égard est la fidélité et la qualité des formes récupérées, notre objectif dans ce travail.

Nous présentons un pipeline complet de reconstruction adapté à ce scénario, pour lequel ce travail apporte de nombreuses contributions. En premier lieu, on peut noter que les méthodes basées sur la technologie stéréo multi-vues (MVS) ont atteint un bon niveau de qualité avec des pipelines qui comprennent généralement l'extraction de descripteurs caractéristiques, une étape de mise en correspondance et l'inférence de forme 3D. Mais il est surtout intéressant de noter que des travaux très récents ont réexaminé le problème de stéréo et stéréo multi-vues en introduisant des fonctions de similarité automatiquement inférées à l'aide d'apprentissage profond. La principale promesse de ce type de méthode étant d'inclure un meilleur a-priori, appris sur les données réelles. Dans une première contribution, nous examinons dans quelle mesure ces améliorations sont transférées au cas plus général et complexe de la capture de performances dynamiques, où diverses difficultés supplémentaires se présentent. Nous expliquons ensuite comment utiliser cette stratégie d'apprentissage pour construire de manière robuste une représentation de forme à chaque instant, à partir desquelles une séquence de modèles 3D peut être extraite. Une fois que nous obtenons cette représentation à chaque instant de la séquence capturée, nous expliquons comment il est possible d'exploiter la redondance

temporelle pour affiner la précision des modèles en propageant les détails des formes observées aux instants précédents et suivants. En plus d'être bénéfique pour de nombreux scénarios dynamiques à vues multiples, cela permet également de capturer des scènes plus grandes où une précision accrue peut compenser la résolution spatiale réduite. Le code source des différentes méthodes de reconstruction est rendu public à la communauté á l'adresse suivante: http://deep4dcvtr.gforge.inria.fr/.

Acknowledgements

This research work was founded by France National Research grant ANR-14-CE24-0030 ACHMOV.

Firstly, I would like to thank both my supervisors for giving me the opportunity to perform my Ph.D study in the MORPHEO team. I am wholeheartedly grateful that they trusted in my potential and continuously supported me during my Ph.D study.

Besides my advisor, I would also like to thank the rest of my thesis committee: Dr. Bertails-Descoubes, Dr. Furukawa, Prof. Vogiatzis, and Dr. Keriven, for their insightful comments and encouragements.

I would like to thank, in order of appearance, Ali, for introducing me to research and anatomy animation, Thomas, Julien and Mickael for *gently* introducing me to homebrewing, Vagia, Benjamin and Aurela for helping me at the beginning of my Ph.D, Adnane and Romain, whithout whom the second gaou would not be niata oh ah, Jinlong for guiding me through the streets of Munich and the strange but nevertheless interesting restaurant in Bucharest, and finally, Victoria, Nitika, Haroon, Pao, Ram, Gwendal, Claude, Di, Boyao, Robin, and all the fellow labmates I could have forgotten, for sharing pleasant cheese tasting evenings and for helping me refine my pizza dough, beer and gravlax salmon recipes.

I am particularly grateful to Julien, Matthieu, Tomas, Eymeric and Edmond for cheering me up through the painful annual *Morpheo Charmant Som trail* and the numerous tedious morning shortcuts.

Moreover, I would like to give a special thank to Matthieu and Jean that provided me with a great stress relief through uninterrupted blundering from both sides in Chess, Go and Shogi games.

I dedicate this particular acknowledgements paragraph to Elisa that excitedly offered me to help in the writing of this whole section, then quietly waited for the manuscript publication.

I would also like to thank my family, my parents and my sisters for supporting me throughout my PhD, with restless skiing, climbing, hiking, mushroom picking, surfing, and related activities.

Finally, I dedicate the rest of this thesis to Marion, whose presence helped me face the stressful events that occur during the life of a Ph.D student. It is hard for me to express how much I owe you, and how much you contributed to the success of this adventure, through endless laughs, arbitrary nonsense, shared accomplishments and bilateral passion. For the support you provided me and for everything else, I hope that we can share a lot more in the future and that I can help overcome your hard times as well.

Contents

List of Figures

List of Tables

List of Abbreviations

AR	Augmented Reality
ARAP	As-Rigid-As-Possible
CAD	Computer Aided Design
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CT	Census Transform
CVT	Centroidal Voronoï Tesselation
GLOH	Gradient Location and Orientation Histogram
GPU	Graphics Processing Unit
HMD	Head Mounted Display
IR	Infrared
MC	Marching Cubes
MVS	Multi-View Stereopsis
NCC	Normalized Cross-Correlation
POV	Point Of View
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RGB	Red Green and Blue color system
ROC	Receiver Operating Characteristic
SAD	Sum of Absolute Distances
SfM	Structure from Motion
SIFT	Scale-Invariant Feature Transform

SLAM	Simultaneous Localization And Mapping
SLIC	Simple Linear Iterative Clustering
SSD	Sum of Squared Distances
ToF	Time of Flight
TSDF	Truncated Signed Distance Function
VR	Virtual Reality
ZSAD	Zero-mean variant of SAD
ZSSD	Zero-mean variant of SSD
ZNCC	Zero-mean variant of NCC

Chapter 1

Introduction

1.1 Motivations

A main goal of Computer Vision lies in digitizing and modeling the 3D world, through one or many light-based sensors, in an automated manner. In particular in this manuscript, our objective is to construct an informative representation of this world, that includes 3D shape geometry, appearance and 3D motion: a 4D Model. This representation is much richer compared to a single video that only captures a 2D projection of the appearance of the scene through time, as it allows the user to navigate through a captured dynamic scene in a highly realistic manner.

The high computational power offered by recent technology advances in computer science allows to model and vizualize captured scenes with a high degree of details and fidelity, thanks to Virtual Reality (VR) head mounted displays (HMD), such as Oculus Rift and HTC Vive, or Augmented Reality (AR) glasses and mobile devices such as Hololens, Magic Leap Lightwear, Vuzix Blade AR, Google Glass, Meta 2 and others. All of these are brought with new developpers tools embedded on mobile devices, such as the ARKit and ARCore, that enable a straightforward registration and synthetic editing of the environment.

This novel domain and technologies allow for numerous new applications mostly revolving around the visualization, analysis and edition of such models, from the quantitative analysis of 3D shapes and motions (metrology), to the creation of immersive 3D content for the movie, VR, AR, telepresence or video game industries. 4D content creation is now becoming a topic of interest with a lot of excitement for immersive realistic experiences, showing a tremendous potential for example in virtual prototyping, architecture, construction or chemistry. In addition to this, the ability to

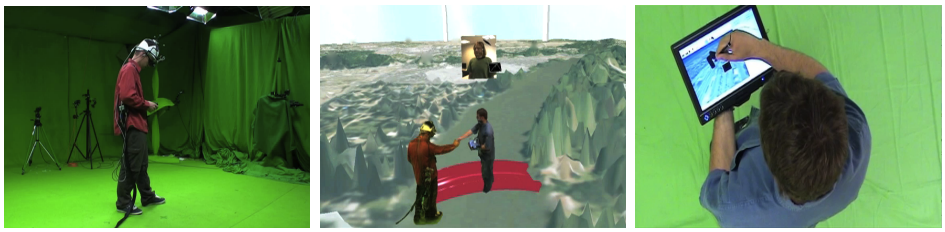

Figure 1.1: Early example of telepresence: a user at Grenoble (*left*) in a 3D modeling studio and another at Bordeaux (*right*) can meet in a virtual environment (*middle*). Figure extracted from [94].

remotely experience possibly destroyed or protected sites or cultural heritages could greatly decrease the human impact and help us preserve our legacy. Finally, part of the excitement arises from conceiving new solutions to therapy and rehabilitation with these new technologies.

Examples of virtual interactive environments are the works of [95, 94] that implemented a precursor immersive telepresence application (Figure 1.1) and an early real-time pipeline for interaction between synthetic and captured subjects (Figure 1.3).

Moreover, the past decade has seen many reconstruction strategies successfully achieving accurate results on full frame static objects. We want to extend reconstruction beyond the scope of such works by choosing to focus on challenging scenes, of mid-scale size (dozen square meters or more), with possibly fast motions and multiple subjects. We examine in particular the case of capture studios large enough for such scenes allowing for numerous moving 4D capture scenarios, for instance sport moves with running, combat, or dancing over a large area. Addressing this use case enhances the creative possibilities for the applications associated to 4D content creation. For example, the reconstruction pipeline presented in this manuscript was used to create content for Augmented Reality in the fashion industry, as seen in figure 1.2. For these experiences to be faithful, thus truly immersive, the detail level is critical, requiring the models to be as accurate as possible. To this aim, many different capture strategies exist, as explained in the following section.

Figure 1.2: An example of application for fashion of the presented pipeline for an AR vizualization of a new collection of Zara (summer 2018). The user can vizualize models wearing the new collection in motion from any point of view, by pointing their smartphones at an empty stand.

1.2 Capture Technologies

The first step towards 4D modeling consists in obtaining digital acquisitions of this world, that can later be processed by a computer. There exist many technologies to this end, mainly divided into two categories: **active** and **passive** systems. While **passive** systems only gather natural light emaning from the environment, **active** systems need an additional source of light, providing more prior information and allowing more accurate results. On the other hand, **passive** systems, that are less demanding, are easier to setup and tend to scale better. We will first provide an overview of the main acquisition technologies along with a brief explanation. In a second time, in 1.3, we compare them and argue about our strategy choice.

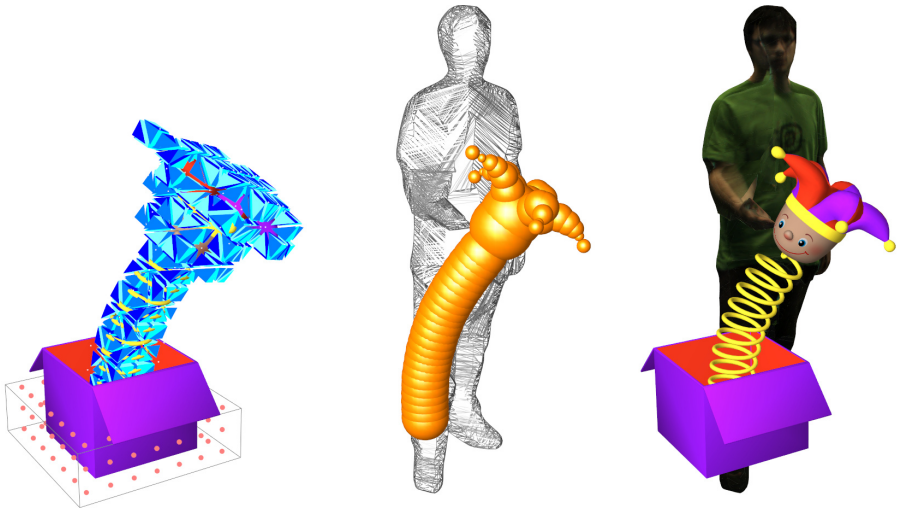

Figure 1.3: Early example of interactions in a virtual environment: an interactive deformable object, *left* collides with the 3D-reconstructed mesh, *middle* allowing real-time interactions with the user in *right*. Figure extracted from [95].

1.2.1 Active Systems

Marker Triangulation: Motion Capture Specific infrared reflecting markers are placed on the observed subject. The scene is illuminated with infrared light and captured with infrared cameras. The reflected light highlights the markers that are then tracked from multiple calibrated points of view. The 3D position of the markers is then obtained by triangulating their projection on the images. This strategy allows to reconstruct sparse 3D points lying on the observed shape (*i.e.* one 3D point per infrared marker).

Laser Triangulation A Laser dot is repeatedly projected in different directions in a scene and detected by a nearby camera. Once again, the position of every dot is obtained by triangulation using the camera's and Laser's positions and orientations. This method outputs point clouds.

Time of Flight (ToF) A laser ray is cast for every pixel of an image, and distance to the laser target is computed by calculating the phase shift between emitted laser and reflected wave. This technology builds depths maps, possibly in real-time (*e.g.* Kinect sensors).

Structured light A known geometric pattern is projected on the object (usually light stripes). The distortion of the pattern on the object gives information about the relative depth of the surface at every pixel, allowing to compute the distance to the object (depth) for every pixel, thus creating depth maps.

Photometric Stereo A scene is captured with different lighting conditions. Shading variations provide normal information about the observed surface. This information is then used to extract the shape.

1.2.2 Passive Systems

Silhouette based A scene is captured with multiple calibrated cameras and the object of interest is segmented in the images. Given the camera's intrinsic and extrinsic properties, it is then possible to backproject the contour of the shape from an image and create a conical volume. The intersection of multiple cones from different points of view defines a convex hull, containing the observed shape. By construction, this method **cannot capture concavities** of an object. This is the first efficient passive method that allowed dense reconstruction of moving humans, that can possibly run in real-time [33].

Shape From Defocus This area of research aims at leveraging the blur arising when an object moves away from the focal plane of the camera. It consists in retrieving the depth information of a scene exploiting the blurring variation of a number of images captured at different focus settings. This strategy can mostly be used for **still scenes** as multiple images with different focuses have to be sequentially taken, complicating the process of capturing fast moving sequences.

Stereopsis The human visual perception system consists of two eyes and a processing unit. The combination of the two sensors allows the brain to exploit the known spacing between the eyes to triangulate points that should correspond to the same 3D point. Similarly, by analyzing the images and finding corresponding pixels, one is able to triangulate them and reconstruct the observed object. Considering a calibrated camera setup, doing this with more than 2 viewpoints is known in the litterature as Multi-View Stereopsis (MVS). Most of the existing MVS methods consider camera spacing to be small.

Structure from Motion (SfM) This strategy is very similar to MVS, the difference is, instead of having multiple sensors simultaneously recording, different points of view are obtained using the same moving sensor. In both instances, the correspondences between images need to be found, but in this case, the viewpoint spacing can be arbitrarily large and the camera's motion has to be estimated on-the-fly, which means that the observed scene can only remain **mostly immobile**.

1.2.3 Other modalities

It is to be noted that other strategies based on different modalities exist as well, such as but not limited to, contact sensors, seismic reconstructions, X Ray imaging and CT scanners, Magnetic Resonance Imaging (MRI), or Sonars.

1.3 Modality Motivation

In this section we will discuss the different modalities introduced in 1.2 that motivated our main focus.

Marker based reconstruction is a well-known and mature processus that allows an accurate capture of sparse interest points on a shape. It has been widely used in the industry for the past decades in different contexts, *e.g.* to animate synthetic characters using skeleton based deformations, driven by natural motion.

Since this strategy only provides a discrete set of points, it cannot always faithfully represent shapes in the case of local non-rigid deformation (*e.g.* fabric folds, skin wrinkles, ...). Furthermore, Humans excel at detecting unnatural local deformations of the shape and appearance through space and time, especially in the area of the faces and the hands. This phenomenon is known as the "uncanny valley" and limits the potential of such strategy due to the lack of realism of animated characters. For these reasons, we favor denser reconstruction solutions, that provide results in a much more complete and realistic fashion.

The other **active** systems allow for high fidelity digitalizations at the cost of more complex and usually expensive setups, where more constraints are to be fulfilled and the framerate is often very low. Moreover, when it comes to scalability, combining several instances of them may induce interferences and is generally harder due to the range shortcomings of ToF sensors for example, contrary to any passive systems. The works of [26, 32] make use of active random infrared projectors to add texture information in

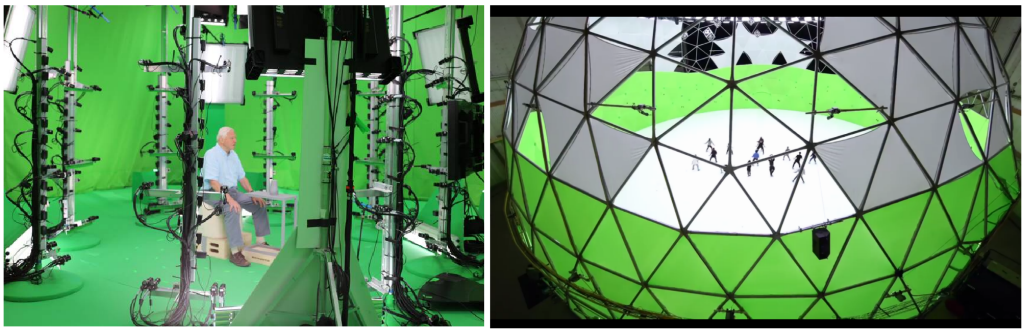

Figure 1.4: (*left*) An example of capture area of the active platform from [26] (credit **Hold the World**). (*right*) The immense capture area of **Intel Studios** [4], a passive capture platform able to record tens of subjects (humans, horses, ...) at the same time.

contrastless regions, improving reconstruction quality. But the scalability of this strategy remains very low as the capture area is only at most a few square meters (approx. $5m^2$), limitating the creative potential of the capture system. On the contrary, and as depicted in figure 1.4 passive platforms such as [4] achieve capture areas up to $930m^2$.

Passive acquisition technologies are all based on the concept of photography discovered in the early 19^{th} century, which allowed a planar sensor to capture the appearance of the world with the use of a projective optic system and a light-sensitive substance. This technology then evolved, in the late 20^{th} century, with the invention of the digital camera. The idea was to replace the chemical sensor (film) with a mosaic photosensor, in order to capture digital images, that can then be automatically processed by a computer. Since then, digital recording devices have been dramatically improved and we are now able to capture high resolution and high framerate images at a relatively low cost, to record any kind of visual event. For example, smartphones (*e.g.* Xperia XZ) can capture up to almost 1000 frames per second (fps), and specialized cameras attain millions, even trillions fps.

Among all passive modeling strategies, initial silhouette based modeling has been overtaken by **MVS** that has proven to be an accurate 4D modeling approach, allowing the most faithful, thus realistic reconstruction of fast moving subjects. Because we aim at capturing large dynamic scenes, we chose to focus our work on passive MVS solutions, that circumvent a lot of the shortcomings of active strategies and allow for flexible setups and applications. The next section presents the problem of passive Multi-View Stereo Reconstruction and explains the key steps of the process.

Figure 1.5: An example of monocular ambiguity: the ash mounds (*left*) above can be percieved as craters when the image is flipped vertically. Figure from [34].

1.4 Problem Statement

Our goal is to model the 4D world lying underneath the successive 2D projections of its appearance as captured by one or many planar camera sensors. In particular, we aim at inverting the image formation process to recover the scene geometry that generated the observations. Recovering scene geometry from a single image is an underdetermined problem since different scenes with different appearances and shading could lead to the same observations *e.g.* figure 1.5. Because of this, solving the single-view problem has been done either by adding strong constraints on the observed shapes, or applying strong priors/regularizers during the reconstruction of the models, at the cost of fine geometry details [112, 133, 58].

As explained earlier in section 1.1, the human binocular perception cortex exploits the spacing between the eyes to triangulate points on the retinas and thus build a partial representation of the depth of the observed object, from a given point of view. Considering a Multi-View Stereopsis setup, the addition of more points of view enables a more complete reconstruction of the scene and adds some disambiguation of the problem thanks to more observations.

In addition to this, the human brain does not only perceive 3D information using stereo matching but also using the world's motion to accumulate information over time. The main intuition is that natural shapes stay roughly the same through time and thus may not completely change between camera frames. More formally, we assume that inside a time window, we observe the same instance of a shape undergoing arbitrary motion. If we assume that this motion is piecewise smooth, it is possible to decou-

ple shape details from motion deformation and use this decomposition to aggregate observations through time. We thus investigate the temporal aspect of captured sequences and propose a strategy to propagate visual cues through neighboring frames and accumulate temporal information in order to improve over per-frame reconstruction quality.

Our objectives in this manuscript are to perform MVS and temporal integration with a passive mid-scale dynamic setup. The following section explains the problems and the challenges raised in such a scenario.

1.5 Challenges and Difficulties

In this work, we focus on large scale dynamic shape capture and reconstruction, also known as *performance capture*, with possibly fast motions and multiple people. Figure 1.6 shows a typical example of dynamic shape capture, containing multiple people, undergoing fast motion. This scene was captured with 64 cameras, at resolution 2048×2048 with focal lengths from $15mm$ to $25mm$ and $10mm \times 20mm$ sensors.

Technical Challenges Contrary to standard MVS scenarios, where the shapes often reproject on the full image, we can see that the subjects reproject on small portions of the image (typically 10 to 20 percent of the images), with the presence of strong motion blur and occlusions. Increasing the acquisition space of multi-camera set-ups raises challenges since it generally requires larger camera field of views and more distant cameras, leading to lower pixel coverage of the scene for fixed sensor resolutions and wider baseline (spacing between cameras).

Figure 1.6: A challenging dynamic scene with fast motions and a mid-scale acquisition space, hence low image resolution on shapes in addition to motion blur and occultations. Temporal integration helps recovering highly detailed models.

In particular, the average baseline between a camera and its 10 closest neighbors is $0.188m$ when considering a standard MVS dataset [62], where it usually goes up to $2.5m$ in our capture scenarios. Because of these differences, it is extremely hard to design a dynamic performance reconstruction system that achieves the same accuracy than usual results on standard MVS datasets. In order to overcome this difficulty, we expect scene details to be accessible by considering observations not only over space, with different cameras, but also over time with moving objects. This requires to first perform static per-frame reconstruction *e.g.* [129, 62, 41] and then to go beyond and turn to temporal redundancy for detail refinement.

Shape Representation Another challenge lies in the choice of shape representation for the scene. Although there exist no continous solution to exactly represent arbitrary surfaces with a limited amount of memory, discrete solutions can reasonably approximate regular shapes. The most common solutions for this are sparse representations such as point clouds or depth maps (a depth map can be trivially transformed into point clouds), and dense representations such as sets of unconnected primitives, surface meshes, or isosurface/isocontour of an implicit form.

Point clouds are a sparse representation which consists of 3D samples of the shape. They are the easiest to obtain: once equipped with a tool to estimate surface presence likelihood in space, in order to recover a full shape, one needs to recover the 3D positions that satisfy a criterion based on this measure. By construction, this function may be ambiguous, as different parts of an object can share the same appearance (*e.g.* repetitive patterns, neighboring points on the shape, ...) and thus contains noise. Typically, if one were to recover photoconsistent points in space using only this criterion, it would give a noisy set of points, with a lot of outliers and a fuzzy detection around the true surface. This comes from the fact that the Lambertian assumption is an innacurate approximation that, firstly, is often violated and secondly, relies on an estimation of the surface's visibility, that can only be computed once an estimation of the shape is known.

Moreover, a sparse representation is not easy to use for the application of new viewpoint rendering, since it consists of unconnected points. Rendering a new view from it would lead to an incomplete image, and requires a realistic interpolation procedure between points, which is a non-trivial problem, especially if the point cloud lacks completeness.

For this application, dense representations, such as meshes or unconnected primitives are to be preferred, since interpolation is inherent to the representation. The most practical method among these is the surface mesh, because it is easy to manipulate, its connectivity is always defined (contrary to the unconnected sets of primitives, where a stitching has to be done between neighboring primitives), it is easy to equip with appearance (texture map, vertex color, triangular color sampling ...) and most of the existing manipulations and rendering softwares are designed to work with it. But directly extracting a closed surface from a noisy point set in space can be a hard problem. There exist two mainstream solutions to this. The first one consists in initializing a mesh that is a coarse estimation of the shape. This mesh is then iteratively deformed to maximize photoconsistency, with a regularizer term enforcing smoothness of the result. The latter offers a tradeoff between surface details and smoothness. The major drawback of such solutions, apart from the difficulty to choose a correct regularizer, is

that most of the time the deformed shape has a fixed topology and thus results quality strongly depends on the initialization of the shape estimation. The second typical solution to this problem is to transform the surface presence (photoconsistency) information into an indicator function that conveys the information of interior and exterior of the observed shape. The interface can then be extracted as an isosurface of this form. This approach allows for topology changes but has generally a larger memory footprint than surface based methods.

The following section describes all the steps and strategy choices necessary to per-frame reconstruction and temporal integration, from the input images to the reconstructed shape.

1.6 Work Overview

As depicted in Figure 1.7, our approach takes as input a set of N synchronized multi-view sequences of length T, $\{I_i^t\}_{i=1}^N$ along with their respective calibrations $\{\pi_i\}_{i=1}^N$, and silhouettes $\{\Omega_i^t\}_{i=1}^N$ extracted using a simple background substraction procedure. Camera parameters are computed in a pre-calibration step, and we assume the calibration parameters remain constant throughout the whole sequence. The output is the corresponding 3D shape at every time step $t \in T$. As detailed in the following subsections, these can be computed in four major steps: the first step of MVS consists in first, finding a way to detect shape presence in the captured 4D space: photoconsistency. Then, this information is used to build a shape representation that can later be refined using temporal redundancy of information. Finally, this representation can be adapted to be taken as input for all sorts of applications. A litterature review of the subject will be presented in Chapter 2, in order to better position this pipeline compared to other works.

1.6.1 Detection of Shape Presence

Considering the physical properties of matter, what appears on the images is the light diffused and reflected at the interface between a transparent medium and an opaque volume. We aim at reconstructing this interface, which can be represented as a closed surface or volumes enclosed by closed surfaces. Most of the works in MVS and stereo reconstruction in general base their methods on the *Lambertian* assumption: a point lying on this surface should have the same appearance when projected on images that see it. Thus, one can recover the observed surface by finding the 3D points

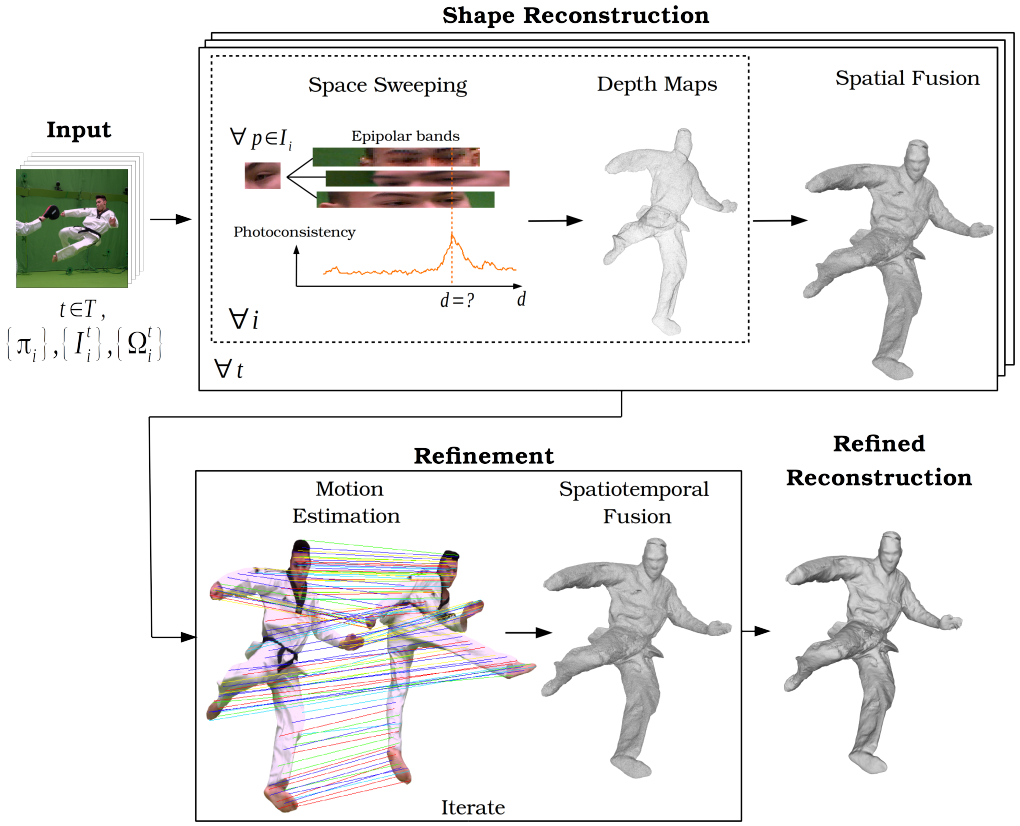

Figure 1.7: Method overview and notations.

that project consistently on different cameras. The indicator function that provides a measure of this coherence is called *photoconsistency*. This assumption is correct only when the observed object is truly Lambertian, and when the visibility of the surface from different points of view is the same. In practice, the performances of standard photoconsistency metrics tend to decrease when trying to reconstruct details finer than the support domain (*e.g.* thin structures, fine folds in a clothing, small facial details), in the presence of strong specularities or when the baseline increases significantly. In this work, we propose to train a CNN to compute photoconsistency and feed this learned criterion to a reconstruction pipeline specifically devised for performance capture scenarios.

1.6.2 Shape Representation

Once given a tool to detect matter presence in space, we want to extract the relevant information, through the use of a mathematical representation of the shape. As explained in the previous section 1.5, a good strategy for this consists in defining an interior indicator from which we can extract a surface mesh.

One way to encode this indicator is through the use of depth maps. Dense depth maps are built similarly to point clouds except that a surface detection along the rays going through every pixel of the image is enforced, directly linking point cloud's density to the capture's resolution. Even though samples (thus pixels) have a neighborhood relationship inherited from the sensor lattice, this topology does not relate to the shape topology, because of image discontinuities produced by the projection. Nevertheless, it is possible to aggregate multiple such depth maps from different points of view into a single 3D implicit form, such as the TSDF or variants [29, 136]. Utilizing depth maps explicitely enforces a visibility model and thanks to the fusion process, provides a robust filtering scheme while still preserving high frequency details.

As for many recent multi-view stereo reconstruction methods [43, 22, 51], ours first estimates per camera depth maps, followed by depth fusion, allowing therefore each camera to provide local details on the observed surface with local estimations.

1.6.3 Propagation of Temporal Cues

As explained in section 1.4, we then present a method to exploit details redundancy through time for precision refinement. In addition to being beneficial to many dynamic multi-view scenarios this also enables larger scenes where such increased precision can compensate for the reduced spatial resolution per image frame. With precision and scalability in mind, we will propose a symmetric (non-causal) local time-window geometric integration scheme over temporal sequences, where shape reconstructions are refined framewise by warping local and reliable geometric regions of neighboring frames to them. This is in contrast to recent comparable approaches targeting a different context with more compact scenes and real-time applications. These usually use a single dense volumetric update space or geometric template, which they causally track and update globally frame by frame, with limitations in scalability for larger scenes and in topology and precision with a template based strategy (see 2.3.2 for more details).

Building a full 4D representation is a tough problem since it is tedious to decouple motion from shape deformation and noise as they depend on many *a-priori* unknown parameters such as topology, local stiffness, appearance variations, etc... For this reason, we do not seek to build a full 4D model, rather we propose to improve 3D shapes where we safely can, by computing inter-frame motion and iteratively using it to propagate visual cues through time. Similar to Non Local Filtering in image processing [20], but with additional motion priors, we believe that our templateless and local iterative approach is a first step towards temporal shape super-resolution.

1.6.4 Surface Extraction

Once the depth map fusion and temporal integration is performed, the last step of the pipeline consists in utilizing the built model for the various applications. The output of the process depends on the application but is usually divided into two intrinsically linked parts: geometry and appearance. For VR or AR, it is not obvious how to easily exploit TSDFs as shape representation since the memory consumption of an explicit storage is proportional to the volume of the capture scene and it does not encode appearance. We want a representation that best resembles reality but that is also light and easy to render, as explained earlier. A common and versatile solution to compress and manipulate shape information efficiently is to use surface meshes equipped with an appearance *i.e.* color information.

1.7 Outline

Chapter 2 gives a brief history and overview of existing works around the problems of passive multi-view stereo reconstruction, dynamic shapes reconstruction, and existing datasets. In chapter 3 we start from a traditionnal handcrafted photoconsistency metric and we then take this strategy a step further by replacing this measure with a learned version. This version is based on CNNs and exploits their ability to learn local photometric configurations near surfaces observed from multiple viewpoints. We will compare these methods, quantitatively and qualitatively, in different scenarios. The next chapter (4) explains how to construct depth maps from these metrics with a restriction on searched space and accumulation scheme.

We then explain in chapter 5 how to aggregate the computed multi-view depth maps at every frame and we introduce a scalable method for temporal local filtering, gathering information from neighboring time steps.

To validate our claims and to evaluate such filtering scheme, we also present a synthetic dynamic dataset similar to our performance capture scenario.

Finally, we present a method to extract a surface mesh in the last chapter (6). In addition to this, we provide comparisons of our reconstructions to multiple state-of-the-art methods for single frame MVS in the performance capture scenario.

1.8 List of Publications

- V. Leroy, J.-S. Franco, E. Boyer. Multi-view dynamic shape refinement using local temporal integration, Proceedings of the IEEE International Conference on Computer Vision, ICCV 2017
  HAL page: https://hal.archives-ouvertes.fr/hal-01567758v2

- V. Leroy, J.-S. Franco, E. Boyer. Shape Reconstruction Using Volume Sweeping and Learned Photoconsistency, Proceedings of the European Conference on Computer Vision ECCV 2018
  HAL page: https://hal.archives-ouvertes.fr/hal-01849286v2

- V. Leroy, J.-S. Franco, E. Boyer. Apprentissage de la Cohérence Photométrique pour la Reconstruction de Formes Multi-Vues, Reconnaissance des Formes, Image, Apprentissage et Perception RFIAP 2018
  HAL page: https://hal.archives-ouvertes.fr/hal-01857627

- (*submitted*) V. Leroy, J.-S. Franco, E. Boyer. Volume Sweeping: Learning Photoconsistency for Multi-View Shape Reconstruction, International Journal of Computer Vision, IJCV

Chapter 2

Related Works

This chapter aims at providing a high-level overview of the related works on the topics of Multi-View Stereopsis, focusing in particular on photo-consistency, performance capture and temporal integration. More in-depth descriptions of the related works on these topics will appear at the beginning of every respective chapter if needed.

2.1 Multi-View Stereo Reconstruction

Multi-view stereo reconstruction (MVS) is a longstanding active vision problem [104]. Initially applied on static scenes, the extension to performance capture of dynamic scenes has become increasingly popular. Stereo and MVS-based approaches are a modality of choice for high fidelity capture applications [41, 110, 42, 91, 62, 87, 102], possibly complementing other strategies such as depth-based reconstruction [90, 60, 26, 32, 31], Shape From Shading [54, 106] and Structured Light reconstruction [121, 98], by addressing shortcomings that include limited range, sensitivity to high contrast lighting, and interference when increasing the number of viewpoints. We do not include in this section a full state of the art in binocular stereo since the we tackle a different problem. Nevertheless, accurate surveys and descriptions of existing methods can be found in [19, 100]. Precursor space carving methods [105] and [71] were among the first efficient MVS methods, based on pixel color difference to incrementally remove matter from a coarse discretized volume. The tendency quickly turned towards global optimization methods, inspired by the work of [36] that proposes a variational approach using level-sets and local correlations, or [40, 39] which optimize meshes or particles systems to minimize a photometric criterion. These methods gave birth to numerous global strategies such as [97, 30] where an

initial shape representation is optimized by minimizing the reprojection error, using a global photometric criterion based on the whole image. These methods quickly became popular since the global gradient descent minimization on the shape was efficient to recover smooth and overall satisfactory results thanks to a finely tuned regularization term. But this term came with a trade-off: surface smoothness against local details. Interestingly, a few years later, the tendency switched back to local solutions with [41], a patch-based region growing strategy and semi-global approaches [22, 129] using respectively sparse or dense local detections and global graph-cuts, which were able to recover fine details of the shapes while still being robust through the use of thoroughly designed filtering steps based on visibility. Ray clique inference methods [119] have also been devised to embed occlusion co-dependencies within a ray deep in the estimation process.

2.2 Photoconsistency

The first step towards MVS consists in finding a way to detect surface presence in the 3D space, solely based on the reprojected appearance on the images. This was already known before 1977 [82] where the authors were already familiar with the concept that a 3D scene point should have the same response to filter kernels in two binocular images. Even though in a biologically oriented manner, they propose to convolve primitive filters on images and then present an algorithm that performs similar sparse feature matching with an outlier removal procedure. A first and more formal theory of human stereo vision [83] was later presented, in 1979. The term voxel consistency was only introduced in 1997 by [105] and later renamed photoconsistency [71]. This term was used to characterize the Lambertian property of observed surfaces: a point of the shape that is visible from multiple cameras is photoconsistent if the colors at its projection on the images is the same. That is, given the true visibility of every 3D point of the scene, every point should convey the same local appearance information. The main difficulty of the reconstruction task in this framework consists in estimating this visibility information. In fact, true visibility can only be recovered if the true shape is known, and on the other hand, the true shape can only be recevered with a perfect visibility model. This entanglement makes MVS a NP-hard problem in this framework [18, 69]. While considering various shape representations, for instance point clouds [41], fused depth maps [88, 85], meshes [109, 72], multi-resolution subdivision surfaces [89] or volumetric discretizations [71, 28, 119], most MVS methods infer 3D shape information by relying on the photoconsistency principle that rays observ-

ing the same scene point should convey similar photometric information, with various strategies to overcome the visibility estimation problem. The next section focuses on methods that are all based on this principle, known as the Lambertian assumption.

2.2.1 Lambertian Assumption Based Solutions

In its simplest form, view consistency can be measured by considering projected color variances among views, as used in early works [71] with limited robustness. The idea of these works is to start with a coarse approximation of the shape to recover, in the form of a regular voxel grid. From this grid, voxels that are not consistent among the views in which they reproject are iteratively removed from the representation. The strength of such strategy lies in the iterative representation refinement that provides an approximate of the visibility of 3D points in space at every iteration. One weakness of this work lies in the pixel-wise consistency term, that lacks robustness in a noisy real-life case. For example, multi-camera calibration most of the time introduces perturbations as the reprojection error is never perfectly minimized. Such pertubations induce misalignments between pixels thus making the consistency noisy. For this reason, in stereo and short baseline situations, normalized forms of 2D window correlation were introduced to characterize similarity under simple lighting and constrast changes, thus relaxing the consistency term and allowing for small pixel misalignments. For broader geometric and photometric resilience, various features based on scale-invariant gradient characterizations [80, 15, 86] have then been designed, some specialized for the dense matching required for the MVS problem [115]. A more extensive description of these strategies is provided in 3. Interestingly, image features have been successfully applied to moving sequences in recent works *e.g.* [87, 74]. Generally, MVS methods characterize photoconsistency in a Lambertian framework either with a symmetric, viewpoint agnostic, combination of all pairwise similarities, *e.g.* [69, 97], or with a per image depth map determination through sweeping strategies as [27, 85].

One of the main limitation of these strategies comes from the fact that most of the natural surfaces are not truly Lambertian. Thus, many works explicitely or implicitely tried to circumvent this approximation, as explained in the following section.

2.2.2 Discarding the Lambertian Assumption

Contrary to the latter works, many authors tried to explicitely model the reflectance of the observed surface, for example in the Shape From Shading scenario [101, 66]. Some passive MVS works try to tackle the same problem using *e.g.* per image post-processing [139] or corrective terms in the photoconsistency evaluation [30].

While classic MVS approaches have been generally successful, recent works aimed at learning stereo photoconsistency have underlined that additional priors and more subtle variability co-dependencies are still discoverable in real world data. Several works leverage this by learning how to match 2D patch pairs for short baseline stereo, letting deep networks infer what features are relevant [130, 81, 137, 120]. Recent works extend this principle to MVS, with symmetric combination of 2D learned features [51]. The advantages of these data-driven methods lie in the capability of the networks to learn invariances in the photoconsistency term, observing natural shapes thus removing, to some extent, the Lambertian approximation (some of these methods still try to match similar extracted features which is still a hidden Lambertian assumption, depending on the learned invariances). Very recent works [46, 59] also focus on reconstructions in the sparse capture scenario, *i.e.* very few cameras capture the subjects, greatly increasing the baseline between cameras.

The common limitation of such methods with 2D receptive fields is the difficulty to correctly capture 3D correlations with hence both false positive and false negative correlations arising from the 2D projection. Some first attempts to learn 3D priors exist by *e.g.* explicitly co-estimating depth and surface normals for each input pixel [44]. Consequently, a number of learned MVS methods resort to full volumetric 3D receptive fields instead, to broaden the capability to any form of 3D correlation in the data [25, 63, 64, 135].

2.3 Dynamic Scene Reconstruction

Performance Capture, *i.e.* dynamic scene reconstruction, where moving subjects are observed by multiple cameras or sensors, has initially been approached on a frame-by-frame basis, with various techniques such as shape from silhouettes [11, 38], MVS [88] possibly combined with keypoint anchors [109]. A natural and more recent strategy to improve shape details and consistency consists in turning to the temporal aspect of such captures by propagating strong cues through neighboring frames in the sequence.

We present recent works towards this direction in this section, considering all modality sensors and passive or active methods with no distinction.

2.3.1 Temporal Constraints

Various techniques have attempted to leverage temporal constraints in the reconstruction reasoning, *e.g.* with 4D Delaunay Triangulation-based carving [11] or with a smooth filtering prior over an extracted 4D hypersurface with MVS constraints [47]. Some approaches consider some form of local propagation based on optical or scene flow matching, to guide smoothness of an inference [73], or the spatial search for feature matching [87]. Some works enforce topological constraints over a sequence [92, 87], ensuring consistent extraction of thin objects (rope) [92] or ensuring a particular silhouette topology [87].

Among inspirational methods, [123] propose an early 6D carving method which carve photometrically inconsistent pairs of voxels for two subsequent frames of a scene. [118] also consider such temporal frame pairs to locally enhance the surface by propagating some stereo information along optical flow, with limited improvements, whereas we generalize this principle for full time windows and local shape alignments. [140] propose a templateless multi-view integration strategy, but only considering quasi-rigid deformations and [24] demonstrate a technique to volumetrically accumulate silhouette cues across time through rigid alignments, principles which we will apply for our MVS based method. [97, 89] simultaneously estimate multi-view stereo and scene flow, but do not use the resulting local matching to propagate stereo information from several frames around a reconstruction.

2.3.2 Template Based solution

The 4D capture problem is very often formulated as a template-based shape tracking and alignment problem. The template may be a laser-scanned [125, 13] or reconstructed surface [132], and use underlying kinematic [125, 9], body-space [16], volumetric [13] cage-based deformations [113] or surface-cohesion [21] constraints to model the non-rigid deformability of the scene. While most methods track a single template for the whole sequence, thus not closely adjusting to the topological and geometric reality of the observed data at each frame, [49, 26] build and track keyframed templates which are discarded every few frames but are locally more faithful to the data. More recent works propose to adapt template-based strategies to the monocular case [134, 50] where a preprocessing step consists in building a detailed template of the observed human, and then deform it using a single RBD

stream. No method of this family refines the reconstructed representation as proposed here.

2.3.3 Causal approaches

Several relevant approaches exist that tackle the problem, such as KinectFusion [61], DynamicFusion [90], Volume Deform [60], Fusion4D [32] or Motion2Fusion [31] showing how a TSDF representation can be used to accumulate passed geometry information over a static or non-rigid object in real-time. But these methods rely on a global non-rigid tracking step aligning passed data to the current frame, which is prone to accuracy and topological drift, especially in the presence of topological splits or merge and fast motion which are common in many datasets. Scalability is also an issue with large scenes due to dense volumetric reference shape representation. Our approach targets a different, offline context where scalability and precision are the main goal, achieved through implicit TSDF representation, robust local propagation and geometry refinement.

Large scene reconstructions. All previous approaches address 4D reconstruction scenarios where the acquisition area is limited to a few square meters. Only a handful of approaches address larger scenes, *e.g.* [23] applies TSDF depth-map fusion on large static scenes, and [48] reconstruct players in stadium events with frame-by-frame reconstructions. More generally, Structure from Motion works focus on large static scenes, *e.g.* "*Building Rome in a Day*" [12] that aims at providing a pipeline for extremely large image collections to reconstruct large scenes such as neighborhoods or cities, or [129] that presents a graph-cut based solution for large scale static outdoor scenes that is used by the Accute3D company [1]. But all these methods do not however address temporal filtering enhancements.

2.4 Datasets

The first benchmark for stereo reconstruction was introduced in the early 2000s [2] and focused on binocular disparity estimation. In 2007, [99] was used to study the intricate relationship between monocular and binocular environment understanding. With the fast growing interest in autonomous driving, the last decade has then seen many other binocular benchmarks introduced such as [45, 67, 96] specialized for driving scenarios *i.e.* stereo rigs mounted on moving cars in dynamic environments, or with varying weather for the latter.

Considering the MVS problem, the first public benchmark was introduced in the mid-2000s [6]. It consisted of an online evaluation system, where the authors submitted their reconstructions on two different objects and results were publicly displayed and compared. Since then, many other datasets were presented including [110, 62, 8, 7], some of them [68, 103] utilizing the same evaluation strategy. However obtaining ground truth equipped MVS datasets in the case of performance capture remains an open problem. The closest would be the FAUST dataset [17], introduced in 2014. This paper presents a pipeline for high quality human shape reconstruction aiming at evaluating 3D mesh registration. The main limitations come from the difficulty of the task, that forced the capture platform to be small and short motion ranges. In order to obtain high quality details in the reconstruction, subjects could not wear complex clothes, and specific patterns were drawn on their body to disambiguate the reconstruction task. Since the appearance is corrupted, and this dataset only comprises small motion range and limited clothings, it is hard to exploit it for our scenario. In this thesis, our strategy to address these shortcomings is to validate individual frames on standard static MVS datasets on one hand, and building synthetic 4D datasets on the other.

Chapter 3

Photoconsistency

As explained in chapter 1, the first step in the reconstruction pipeline consists in finding a way to detect surface presence using the appearance of the object captured from multiple points of view. In this chapter, we first present in 3.1 a photoconsistency estimation derived from traditionnal metrics. Since these are all handcrafted solutions, we then propose to take this strategy a step further by replacing it with a learned version (3.2). This version is based on CNNs and exploits their ability to learn local photometric configurations near surfaces observed from multiple viewpoints. We will finally compare these methods, quantitatively and qualitatively, in different scenarios (3.3).

3.1 Gradient Based Photoconsistency

The measure of photoconsistency was introduced based on the physical properties of matter, most of the time considering a Lambertian assumption. The initial basic criteria consists in using absolute pixel color difference: a point that lies on the surface should receive the same light (intensity and spectrum), also known as the brightness constancy assumption.

A consensus on comparing local windows of the images instead of single pixels seems to have risen in the MVS case, mainly because it offers more robustness to the types of noise that arise in this scenario, namely reprojection errors, sensor noise, blur, vignetting or non-Lambertian nature of real surfaces. After the images patches are aligned with an homography depending on intrinsic and extrinsinc camera parameters, one can compute many different metrics on it, such as, but not limited to, Sum of Absolute Distances (SAD), Sum of Squared Distances (SSD), Normalized Cross-Correlation (NCC), Sum of Haming Distances (SHD) or Census transform

(CT). Many criteria implementing local and global normalization or zero-mean variants such as ZNCC, ZSAD, ZSSD or adaptative window size, have been proposed to relax the constraints and gain some invariances, *e.g.* in intensity as cameras may have different expositions/aperture breaking the brightness constancy assumption. Overall the goal is to make the method more robust to all kinds of noise. One notable trend is gradient based histogram descriptors, introduced with [79], and later derived in many variations such as SIFT [80], GLOH [86], Daisy [114]. These descriptors aim at pooling and binning responses to various image gradient filters, to build gradient histograms that describe an area of an image. Many image description methods and studies were proposed in the litterature in the last decade and we refer to [53, 80, 86, 114] for more extensive surveys and tests on these strategies.

3.1.1 Photoconsistency measure

In order to detect surface presence in space, we make use of a photoconsistency measure evaluated along the rays cast through every pixel of every image and based on pairwise photometric discrepancy. While Normalized Cross Correlation has been extensively used over the past [41, 92, 35, 127], as explained in the previous section 3.1, recent advances in image descriptors have demonstrated the benefit of gradient based descriptors, such as SIFT, GLOH, DAISY [80, 86, 114], especially with noisy photometric information. We chose DAISY as it experimentally gives the best results in the wider baseline MVS context.

For a point $x \in \mathbb{R}^3$ and given two images I_i and I_j, the pairwise photometric discrepancy $g_{i,j}(x)$ at x is given by the Euclidean distance between the two descriptors D_i and D_j of the point's projection in the images:

$$g_{i,j}(x) = (D_i(\pi_i(x)) - D_j(\pi_j(x)))^2. \tag{3.1}$$

The photoconsistency measure $\rho_i(x)$ at x, given all the images, is then computed as a normalized robust vote of the image descriptors $D_j(\pi_j(x))$ at x that are similar to $D_i(\pi_i(x))$. In contrast to [127], who consider only local minima of the pairwise discrepancy $g_{i,j}(x)$ and interpolate them, we consider all the discrepancy values. This is based on our observations that, in the mid-scale context, surface points are less likely to define local minima of $g_{i,j}(x)$ than in the small-scale case that presents short baselines. Hence our photoconsistency measure $\rho_i(x)$ is:

$$\rho_i(x) = \sum_{j \in C_i} \bar{\omega}_j W(g_{i,j}(x)), \tag{3.2}$$

where: the normalized values $\bar{\omega}_j$ of $\omega_j = \cos(\theta_{ij})$ weights camera contributions around camera i using the angle θ_{ij} between the optical axes of camera i and j; C_i is the subset of cameras j such that $\omega_j > 0.7$; and $W()$ is a robust voting function, a Gaussian Parzen-Window in the descriptor space in our experiments. Note that 1 is therefore the best score $\rho_i(x)$ when all cameras in C_i present the exact same image descriptors at x and 0 the worst.

The above photoconsistency measure implicitly assumes Lambertian surfaces and while robust to a certain extent to specularities it can still fail when strong highlights occur. Also regarding occlusions, we expect ρ to present local maxima where rays intersect the surface even in the presence of occlusions.

3.2 Data driven solution

In this section, we propose a solution to replace the handcrafted method presented previously with a data driven strategy. Existing learning based state-of-the-art methods can be separated in two main categories: image based and geometry-aware strategies. Image based solutions, such as [130, 81, 137], tackle the photoconsistency problem by matching similar learned features extracted from the images. In particular, these works allow to relax the Lambertian assumption, by making the network learn its own invariances.

On the other hand, geometry aware methods take camera relative positions and calibrations into account by different means. [120] give an insight of the geometry to the network by enforcing it to predict optical flow between neighboring views, *i.e.* parallax between the points of view. [25, 64, 63, 135, 57] all back-project image colors or features in a volumetric discretization of the reconstructed space, thus implicitely encoding camera geometries. Then operations on this grid such as aggregation, 3D convolutions or pooling are performed to gather information from all the points of view. Spatial relationships between points of the reconstructed shape can thus be encoded and exploited by the networks.

While casting correlations in 3D similarly to these previous works, our approach proposes several key differences: our volumetric receptive field is a back-projected image region, similar to some binocular stereo [65] or image-based rendering [37] works, where the latter only uses the grid as proxy without explicitely extracting 3D information. This enables a sweeping search strategy along viewing rays, which proves a robust search strategy as plane sweeping in stereo reconstruction. This scheme also avoids

decorrelating camera resolution and 3D receptive field resolution, as with *e.g.* voxels, the volumetric receptive field being defined as a backprojection along pixel rays. Additionally, this volumetric receptive field learns local pairwise correlations, a lower level and easier task than learning occupancy grid patterns.

The following sections explain how we compute surface presence probability by using a volume sweeping strategy that samples multi-view photoconsistency along rays.

3.2.1 Learning Surface Presence Probability

For a point along a viewing ray, the photoconsistency is estimated using a discretized 3D volumetric patch around that point. In such a 3D patch, at each point within, color information from the primary camera ray incident to that point is paired to the color information of the incident ray of another camera. We collect these paired color volumes for every other camera than the primary. A trained CNN is used to recognize the photoconsistent configurations given pairs of color samples within the 3D patch. The key aspects of this strategy are:

- The per camera approach, which, by construction, samples the photoconsistency at a given location as captured and thus enables more local details to be revealed compared to global approaches, as shown in Figure 6.5.

- The 3D receptive field for the photoconsistency evaluation, which resolves some 2D projection ambiguities that hindered 2D based strategies, as tested in 3.3.1.

- The learning based strategy using a convolutional neural network, which outperforms traditional photometric features when evaluating the photoconsistency in dynamic captured scenes, as demonstrated by our experiments in 3.1.

The following sections focus on our main contributions, namely the 3D volume sampling and the learning based approach for the photoconsistency evaluation. Note that for the final step we use the TSDF to fuse depth information and [74] to get a 3D mesh from the fused depths, as explained in chapters 5 and 6.

3.2.2 Surface Detection by Volume Sweeping

Our reconstruction approach takes as input N images $\{I_i\}_{i=1}^N$, along with their projection operators $\{\pi_i\}_{i=1}^N$, and computes depth maps, for the input images, that are subsequently fused into a 3D implicit form. This section explains how these maps are estimated. Given a pixel p in an input image i, the problem is therefore to find the depth d along its viewing ray of its intersection with the observed surface. The point along the ray of pixel p at depth d is noted $r_i(p, d)$. Our approach searches along viewing rays using a likelihood function for a point to be on the surface given the input color pairs in the evaluation volume. In contrast to traditional methods that consider hand-crafted photoconsistency measures, we learn this function from multiview datasets with ground-truth surfaces. To this purpose we build a convolutional neural network which, given a reference camera i and a query point $x \in \mathbb{R}^3$, maps a local volume of color pair samples around x to a scalar photoconsistency score $\rho_i(x) \in [0..1]$. The photoconsistency score accounts in practice for color information from camera i at native resolution, and for other camera colors and their relative orientation implicitly encoded in the volume color pair construction. These important features allow our method to adapt to specific ray incidences. Its intentionally asymmetric nature also allows subsequent inferences to automatically build visibility decisions, *e.g.* deciding for occlusion when the primary camera i's color is not confirmed by other view's colors. This would not have been possible with a symmetric function such as [51].

We thus cast the photoconsistency estimation as a binary classification problem from these color pairs around x, with respect to the reference image i and the other images. In the following, we first provide details about the 3D sampling regions before describing the CNN architecture used for the classification and its training. We then explain the volume sweeping strategy that is subsequently applied to find depths along rays.

3.2.3 Volume Sampling

In order to estimate photoconsistency along a viewing ray, a 3D sampling region is moved along that ray at regular distances. Within this region, pairs of colors backprojected from the images are sampled. Each pair contains a color from the reference image and its corresponding color in another image. Samples within the 3D region are taken at regular depths along viewing rays in the reference image (see Figure 3.1). The corresponding volume is a truncated pyramid that projects onto a 2D region of constant and given dimension in the reference image. This allows the 3D sampling to

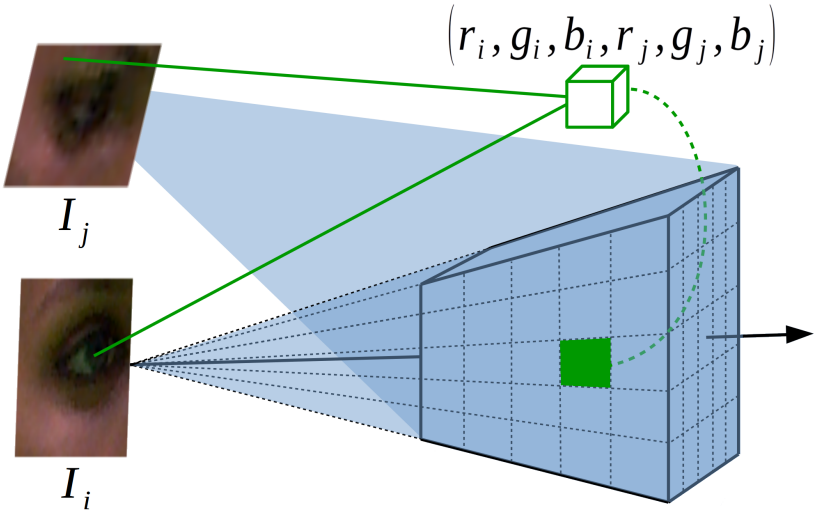

Figure 3.1: The 3D volume used to estimate photoconsistency along rays from the reference image i. k^3 samples within the volume are regularly distributed along viewing rays and contain color pairs as back-projected from images i and j. At a given depth along a ray from i each image $j \neq i$ defines a pairwise comparison volume.

adapt to the camera perception properties, *e.g.* resolution and focal length.

More precisely, consider the back-projection $r_i(p, d)$ at depth d of pixel p from the reference image i. The k^3 input sample grid used to compare pairs of colors from images $\{i, j\}_{j \neq i}$ is then the set of back-projected pixels in a k^2 window centered on p, regularly sampled from depth $d - k\lambda/2$ to $d + k\lambda/2$, with λ chosen s.t. spacing in the depth direction is equal to inter-pixel distance from the reference camera at that depth. Every sample contains the reference color of the originating pixel in image i and the color of the point projected on camera j.

Volume sampling is always performed with the same orientation and ordering with respect to the reference camera. Convolutions are thus consistently oriented relative to the camera depth direction.

Intuitively, computing a traditional photoconsistency term based on a planar support works under the assumption that the surface can be locally approximated by planes. Also, since our network is characterizing ray co-incidency inside a volume unit, we totally discarded the Lambertian approximation and leave the decision to the CNN.

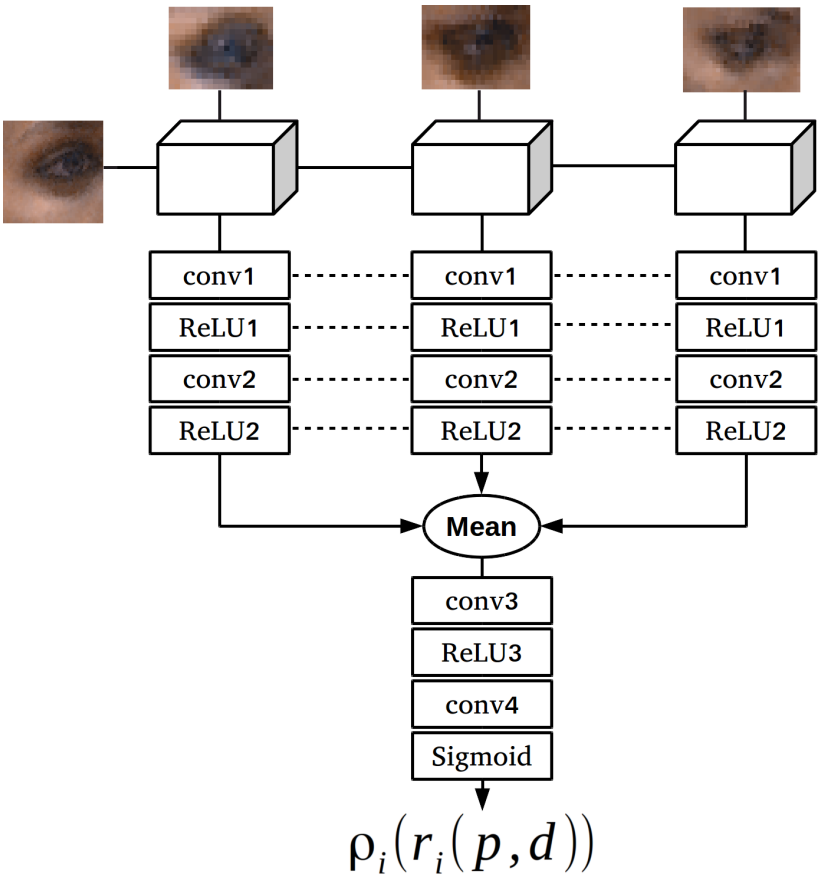

Figure 3.2: CNN architecture. Each cube is a pairwise comparison volume with k^3 samples that contain 6 valued vectors of RGB pairs and over which 3D convolutions are applied. The output score $\rho_i(r_i(p,d)) \in [0..1]$ encodes the photoconsistency at depth d along the ray from pixel p in image i.

Volume Size In practice, we choose $k = 8$. Our strategy is to learn pairwise photoconsistent configurations along rays. This way, decisions for the surface presence are conditioned to the observation viewpoints, which implicitly enforce visibility rules since only one 3D point per ray can be detected. This is in contrast to more global strategies where such per viewpoint visibility is less easy to impose, as with regular voxel grids, *e.g.* [63] with 32^3 or 64^3 grids. In addition, by considering the surface detection problem alone, and letting the subsequent step of fusion integrate depth in a robust and consistent way, we simplify the problem and require little spatial coherence, hence allowing for small grids. We provide a more detailed study of the performances of the classifiers with various depth values

in section 3.3.

3.2.4 Multi-View Neural Network

As explained in the previous section, at a given point x along a viewing ray we are given $N-1$ volumes colored by pairs of views, *i.e.* $(N-1)\times k^3$ pairs of colors, and we want to detect whether the surface is going through x. To this aim, we build siamese encoders similarly to [51], with however 3D volumes instead of 2D patches. Each encoder builds a feature given a pairwise volume. These features are then averaged and fed into a final decision layer. Weight sharing and averaging are chosen to achieve camera order invariance.

The network, depicted in Figure 3.2, is derived from the AlexNet architecture [70] with a siamese strategy following the works of [130, 51]. The inputs are $N-1$ colored volumes of size $k^3 \times 6$ where RGB pairs are concatenated at each sample within the volume. Convolutions are performed in 3D over the 6 valued vectors of RGB pairs. The first layers (encoders) of the network process every volume in parallel, with shared weights. Every encoder is a sequence of two convolutions followed by non-linearities, and max-pooling with stride. Both convolutional layers consist of respectively 16 and 32 filters of kernel $4 \times 4 \times 4$, followed by a Rectified Linear Unit (ReLU) and a max-pooling with kernel $2 \times 2 \times 2$ with stride 2. We then average the obtained $2 \times 2 \times 2 \times 32$ features and feed the result to a 128 filter $1 \times 1 \times 1$ convolutional layer, followed by a ReLU and a final $1 \times 1 \times 1$ decision layer, for a total of $72K$ parameters . The network provides a score $\rho_i(r_i(p,d)) \in [0..1]$ for the photoconsistency at depth d along the ray from pixel p in image i.

We experimented with this network using different configurations. In particular, instead of averaging pairwise comparison features, we tried max-pooling which did not yield better results. Compared to the volumetric solution proposed by [63], the number of parameters is an order of magnitude less. As mentioned earlier, we believe that photoconsistency is a local property that requires less spatial coherence than shape properties.

3.2.5 Network Training

The networks were implemented using TensorFlow and trained from scratch using the DTU Robot Image Dataset [62], which provides multiview data equipped with *ground-truth* surfaces that present an accuracy up to $0.5mm$. From this dataset 11 million k^3 sample volumes were generated, from which we randomly chose 80 percent for training, and the remaining part for

evaluation. Both positive and negative samples were equally generated by randomly sampling volumes up to $20cm$ away from ground truth points, where a volume is considered as positive when it contains at least μ ground truth points. In theory, the network could be trained with any number of camera pairs, however, in practice, we randomly choose from one up to 40 pairs. Training was performed with the binary cross entropy function as loss. Model weights are optimized by performing a Stochastic Gradient Descent, using Adaptive Moment Estimation on $560,000$ iterations with batch size of 50 comparisons, and with a random number of compared cameras (from 2 up to 40). Since our sampling grids are relatively small and camera dependent, we are able to generate enough sample variability for training, without the need for data augmentation.

3.3 Evaluations

Our main goals in this section are (i) to evaluate whether and how our learned photoconsistency contributes with respect to existing methods and (ii) to verify whether these transfer to the more complex case of generic 3D capture scenes in practice, *e.g.* humans with complex clothing. To this aim, we perform various evaluations to verify and quantify the benefit of our learned multi-view similarity. We start by providing multiple validation experiments to justify the choices for the learning and reconstruction strategies in 3.3.1.

Then, we build experiments to test the main claim of improvement with production capture data in 3.3.2. To this goal we use several dynamic sequences which exhibit typical difficulties of such data. In particular, we mainly focus on the Kinovis acquisition platform [5], which consists of 68 RGB cameras, of resolution 2048×2048 with focal lengths varying from $8mm$ to $25mm$. We achieve very significant qualitative improvements compared to the handcrafted approach of [74], without fine-tunning and despite the difference of capture setup used for training. To do so, we replaced the handcrafted photoconsistency estimation with our learning approach in the reconstruction pipeline. We will provide in the next chapters more exhaustive quantitative and qualitative (resp. 4.2.1 and 6.2.2) comparisons to state of the art methods on various datasets, both handcrafted and learning based.

3.3.1 Validation

We previously formulated the problem of surface detection along viewing rays as a binary classification problem 3.2.2. In order to assess the benefit of our volumetric strategy, we first focus on different classifiers performances. We provide in 3.3.1.1 receptive field comparisons on the training dataset this to enhance the advantage of casting and learning correlations in 3D. In addition to this, section 3.3.1.2 provides a study of the depth hyperparameter of the receptive field of our network. Finally, since preliminary results of [75] seemed to show a better robustness to a larger baseline, we design an experiment with cameras that are further apart to better quantify this improvement in section 3.3.1.3.

Section 3.3.1.2 shows that a volume size of $8 \times 8 \times 8$ is a preferred trade-off, thus will be used from now on, when not specified.

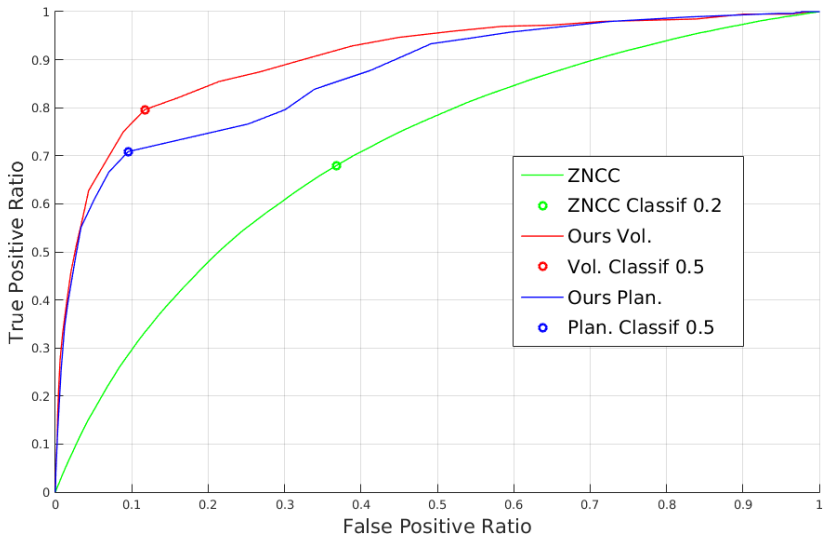

Figure 3.3: ROC Curves of three different classifiers, ZNCC, planar and volumetric receptive fields, on the DTU Dataset [62]. Circles represent thresholds that optimize sensitivity + specificity with the values 0.2, 0.5 and 0.5 respectively.

3.3.1.1 Classifiers Study

In this paragraph, we compare performances of different classifiers based on various receptive fields:

1. Zero-Mean Normalized Cross Correlation (*ZNCC*): ZNCC is applied over the samples within the volumetric support region.

2. Learning (CNN) with a planar support: a planar equivalent of our volumetric solution, with the same architecture and number of weights, in a fronto-facing plane sweeping fashion.

3. Learning (CNN) with a volumetric support: our solution described in the previous sections.

Figure 3.3 shows, with the classifiers' ROC curves, that the most accurate results are obtained with a volumetric receptive field and learning. Intuitively, a volumetric sampling region better accounts for the local non-planar geometry of the surface than planar sampling regions. This graph also emphasizes the significantly higher discriminative ability of learned correlations compared to deterministic ones.

3.3.1.2 Volume Sampling

To further demonstrate this, we then proceed to a study on the impact of the depth parameter of the sampling volume. While keeping a 8×8 pixels reprojection on the images, we study the performances on classifiers with receptive fields varying in depth. Figure 3.4 shows classifiers performances with depth values ranging from 1 to 12. To perform this experiment, we had to diminish the networks number of parameters to fit the 12 depth training in memory and keep reasonable training and testing times, explaining the worse performances compared to previous ROC curves. This experiment demonstrates that the more information the network gathers along the ray the better the detection of the surface is. We choose a depth of 8 as it gives the best trade-off between computational complexity and performance.

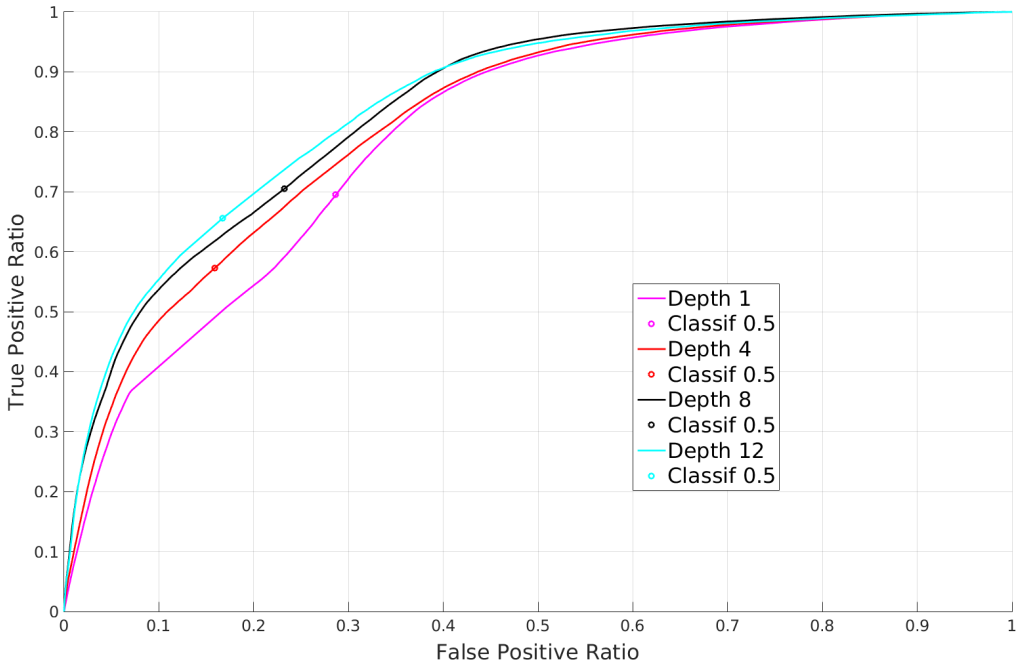

Figure 3.4: ROC Curves of four different classifiers using receptive fields with various depths. Circles represent thresholds that optimize sensitivity + specificity.

3.3.1.3 Baseline Study

We now evaluate the robustness to various baselines by accounting for a higher number of cameras and more distant cameras in the classification.

Table 3.1: Classifier accuracy (%).

Camera #	5	20	49
ZNCC	64.98	65.46	65.58
Ours Plan.	80.67	77.87	75.92
Ours Vol.	**82.95**	**84.84**	**83.45**

Table 3.1 shows the accuracy of the classifiers with a varying number of cameras and for the optimal threshold values in Figure 3.3. As already noticed in the literature, *e.g.* [41, 91], a planar receptive field gives better results with a narrow baseline and the accuracy consistently decreases when the inter-camera space grows with additional cameras. In contrast the classifier based on a volumetric support exhibits more robustness to the variety in the camera baselines. This appears to be an advantage with large multi-camera setup as it enables more cameras to contribute and reduces hence occlusion issues.

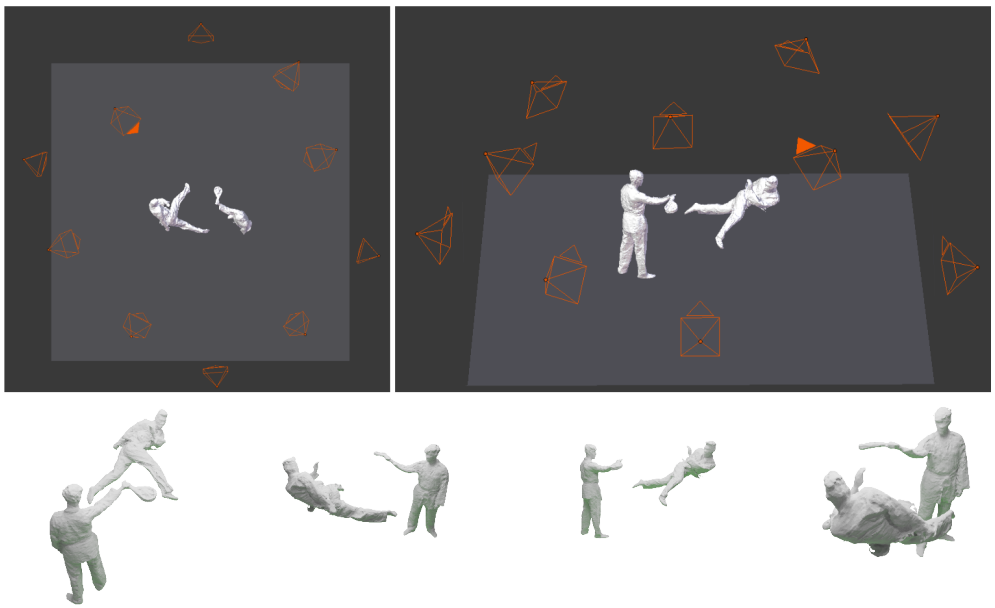

Figure 3.5: An example of sparse synthetic performance capture data generation. *(top)* Top and side view of the 10 cameras positionned around a surface. *(bottom)* Four examples of generated points of view.

To push this experiment further, we design an experiment to test the robustness of our approach on a sparse capture platform, with lower scene

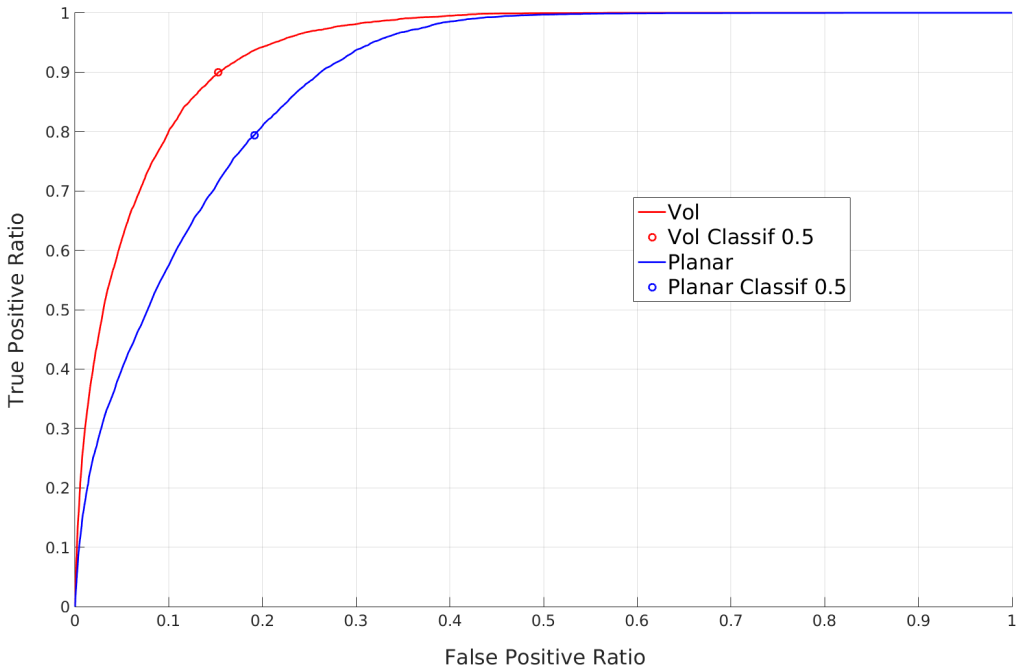

Figure 3.6: ROC Curves of two different classifiers using planar and volumetric receptive fields, on the sparse synthetic data. Circles represent thresholds that optimize sensitivity + specificity.

coverage and wider baseline. Since no ground truth exists for this kind of performance capture scenario, we simulate it using of a realistic rendering engine to create a synthetic dataset. Similar to [5] in terms of camera parameters and capture volume, we chose to render only 10 randomly placed cameras, evenly distributed on an hemisphere around the capture volume. The average spacing between a camera and its 10 closest neighbors is $8.03m$ in this case, where it is $2.5m$ for the 68 POV kinovis platform and $0.188m$ in the 49 POV DTU case. For this experiment, we set the neighboring camera acceptation threshold $\cos(\theta_{ij})$ to 0.1, meaning that we accept almost orthogonal cameras. The synthetic cameras render the scene using Filmic Blender [3], a photorealistic configuration for Blenders Cycles ray-tracing engine. The images are generated with random parameters, *i.e.* the cameras parameters vary, in terms of position, orientation, focal length, and pixels number of samples, the latter directly affecting sensor noise. With this platform, we rendered a dozen of models such as procedurally generated geometric shapes, real life reconstructions or CAD models with various appearances. The multiview networks are trained from scratch on

these synthetic examples, and evaluated on unseen synthetic data. Figure 3.5 shows an example of our synthetic platform as well as the generated synthetic data. We show in figure 3.6 the impact of a volumetric support: when the baseline between the cameras becomes extreme, it offers more robustness compared to a planar support, which appears very slanted in the compared view. Even though it is only a synthetic dataset, we believe that it gives interesting insights on the versatility of our volume sweeping strategy for the performance capture scenario. A qualitative result of this improved robustness is shown in figure 3.10. The area of the face is highly occluded, and the volumetric support helps recovering a smoother surface. Also note the details of the belt: the volume allows a sharp reconstruction of finer details, where a plane cannot handle finer geometry details.

3.3.2 Qualitative Evaluation

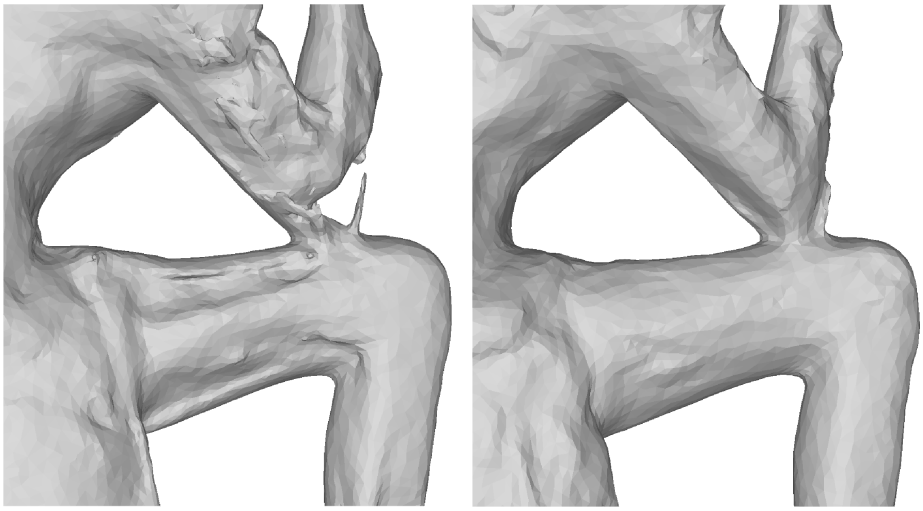

Figure 3.7: Close up view of the arm region in Figure 3.9. (*Left*) Results from [74], (*right*) our reconstruction

One of our main goals is to verify whether a learning based strategy generalizes to the performance capture scenario and how it compares to state-of-the-art deterministic approaches in this case. To this purpose, we perform reconstructions of dynamic RGB sequences captured by a setup largely different from the training one, *i.e.* a hemispherical setup with 68 cameras of $4M$ resolution with various focal lengths, as provided in [74] along with reconstructions obtained with a deterministic approach. In

this scenario, standard MVS assumptions are often violated, *e.g.* specular surfaces, motion blur, wider baseline and occlusions, challenging therefore the reconstruction methods.

For these evaluations, we extract surfaces using the depth estimation method described in 4 and the surfaces were extracted using the approach described in chapter 6 without temporal integration. No other modification was applied, in particular the network previously trained was kept as such without any fine tuning. Figure 3.9 shows a reconstruction using our learning method compared to DAISY based photoconsistency [74]. Even though [74] performs well in contrasted regions, the patch based descriptors reach their limits in image regions with low contrast or low resolution. Figure 3.7 and 3.8 give such examples. They show that our learning based solution helps recover finer surface details, while strongly decreasing noise in low contrast regions. The results obtained also demonstrate strong improvements in surface details, such as dress folds, that were undetected by the deterministic approach. In addition, they demonstrate lower levels of noise, particularly in self-occluded regions, and more robustness to motion blur as with the toes or tongue-in-cheek details that appear in figure 3.8-bottom.

From a qualitative perspective the results obtained in figure 3.9 with our learning-based similarity again exhibit drastic improvements, in particular picking up detail from signal barely above noise levels in completely dark areas of the image, that are almost undetected by the classic MVS method.

In the next qualitative experiment, we study the impact of a volumetric support compared to the equivalent planar one (see sec. 3.3.1) in figure 3.10. The area of the face is highly occluded, and the volumetric support helps recovering a smoother surface. Also note the details of the belt: the volume allows a sharp reconstruction of finer details, where a plane cannot handle finer geometry details. A video demonstrating results on dynamic sequences is available online: `https://hal.archives-ouvertes.fr/hal-01849286`. We refer to section 6.2.2 for more qualitative comparisons with state-of-the-art methods.

3.4 Conclusions

We presented in this chapter multiple strategies to estimate photoconsistency for MVS that can be used in the performance capture scenario. First, we described a traditional approach using handcrafted image descriptors based on histograms of gradients. We then took the approach a step further and proposed a method to replace this term with a data-driven solution. The proposed strategy leverages the recent success of CNNs in a per-camera,

geometry aware manner. We trained the network in a supervised manner on a standard static MVS database equipped with *ground-truth*. We then demonstrated the generalization capabilities of our network by using it in the performance capture scenario with no fine-tunning. We believe this is possible because we designed the network to focus on learning a local property of coinciding colored rays, which is a low-level task, simpler than trying to infer complex spatial coherence, as done in other works *e.g.* [135, 63].

Another important aspect of our evaluation consists in the comparison of traditional plane based photoconsistency using 2D convolutions to the rather new trend of casting rays in a volume and performing 3D convolutions. We showed that using a volumetric receptive field allows for better robustness and helps disambiguate the surface detection step. We provided quantitative evaluations of different classifiers performances on real data in the narrow baseline case and in the performance capture scenario to showcase this. We also designed a synthetic experiment reproducing a realistic sparse setup with only a few cameras observing the subjects. We showed that using 3D convolutions helps to gather information from cameras further apart, with aggregation of views that could be almost orthogonal. Finally, we performed reconstructions using both the introduced learning strategy and the previous handcrafted solution and give qualitative comparisons to better enhance the differences between strategies. The learned photoconsistency allowed for better reconstruction quality, providing more faithful details while decreasing noise compared to the handcrafted solution.

The next chapter 4 explains how to make use of the introduced surface detection probability to build a shape representation.

Figure 3.8: (*Top*) input images, (*middle*) result with [74], (*bottom*) result with our method. Motion blur and low contrast are visible in the input images . Best viewed magnified.

Figure 3.9: Challenging scene captured with a passive RGB multi-camera setup [5]. (*left*) one input image, (*center*) reconstructions obtained with classical 2D features [74], (*right*) proposed solution. Our results validate the key improvement of a CNN-learned disparity to MVS for performance capture scenarios. Results particularly improve in noisy, very low contrast and low textured regions such as the arm, the leg or even the black skirt folds, which can be better seen in a brightened version of the picture in Figure 6.5.

Figure 3.10: (*Left*) 3 input images, (*middle*) plane based classifier, (*right*) volumetric classifier. The face is highly occluded (*left*) yielding noisier and less accurate reconstructions when using a planar receptive field, whereas the volume counterpart yields smoother and more accurate details.

Chapter 4

Depth Map Construction

Until now, we saw that the captured shape gives rise to different observations from multiple points of view and we explained in chapter 3 how to compare local regions of these images in order to detect surface presence in space. A simple thresholding can then be applied to extract the 3D points that belong to the surface. But this information is noisy and ambiguous since the probability of surface detection is also high when close but not onto the real surface. These false positive detections will often result in *"thick"* detections of the surface, as seen in Figure 4.1.

Although, many strategies exist to set the thresholding parameter locally to improve detection performances, *e.g.* [127, 63], one common strategy that proved to be more efficient consists in transforming this surface presence probability into an interior/exterior indicator, while the shape we are looking to reconstruct is the interface of this indicator. The idea behind this strategy is to filter the noisy surface detection responses in a region and find the most suitable candidates among them by taking advantage of the visibility and occlusions of every point in space according to every camera. The first works that introduced this kind of approaches were the space carving methods of [105, 71], later adapted to different problems such as scene flow estimation [123] or non Lambertian reconstruction [139]. Many different strategies ramified around the explicit formulation of the visibility information such as surface evolution optimizations *e.g.* [97, 47], local surface detection denoizing with visibility filtering [41, 77], graph-cut based strategies [129], ray potentials [119] or finally, constraints embedded in a depth maps estimation framework *e.g.* [27, 43, 91, 93, 92, 127].

As explained earlier, we choose to make use of depth maps for the simplicity of computation and because the TSDF fusion strategy proved to be extremely efficient in the performance capture scenario, with *e.g.* [61, 90, 32].

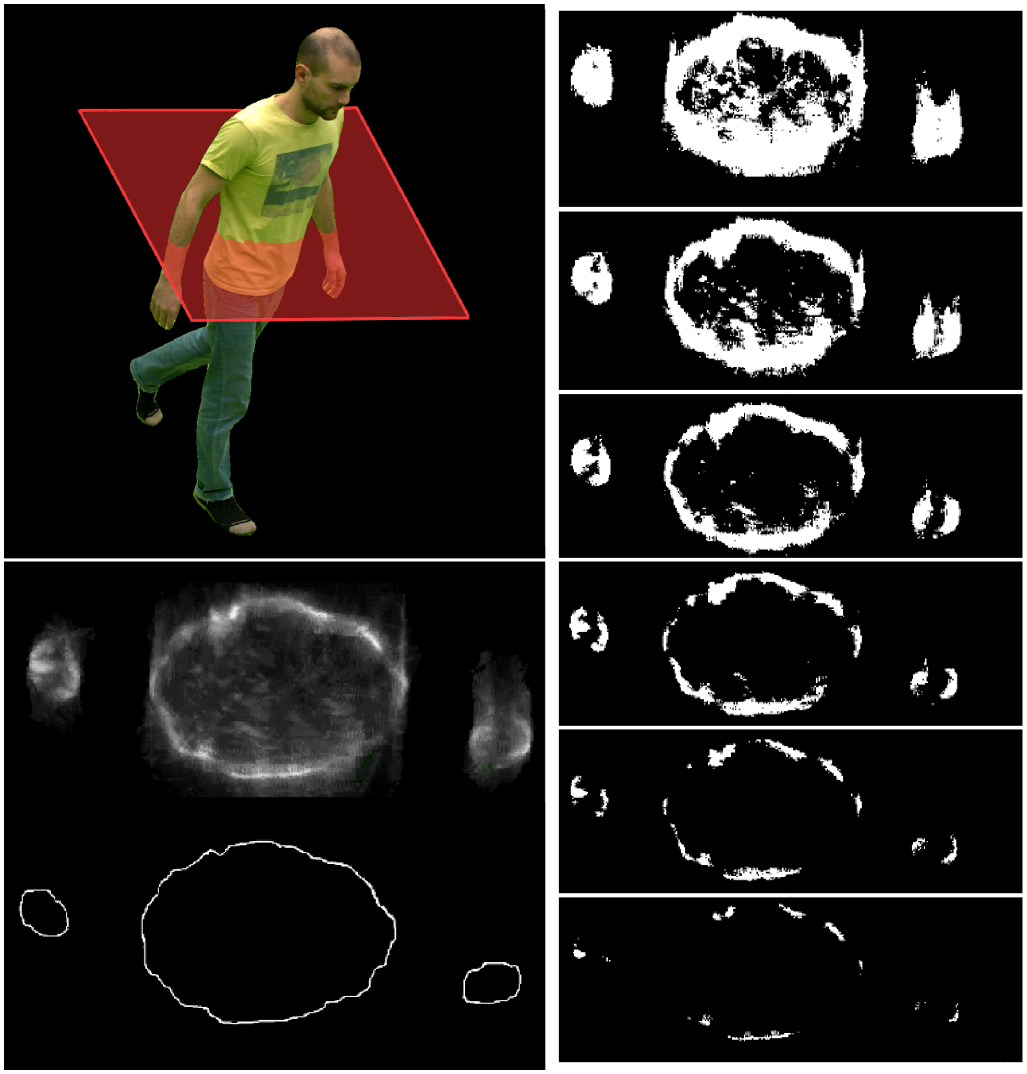

Figure 4.1: An example of surface detection probability. A 2D slice (*red*) of the input shape (*top left*) is used as support to display photoconsistency (*middle left*). The thresholded detections (*right*) with increasing threshold value (*from top to bottom*) are very noisy and thick or incomplete, and remain far from our objective (*bottom left*).

The main difficulty now is that the exact visibility information can only be derived from the true observed shape, while the true shape can only be correctly recovered with the true visibility. In this chapter, we will explain how to compute, with a local strategy, depth estimations along rays according to the photoconsistency values. It consists in a probability accumulation along rays coming from every camera, with a spatial restriction based on silhouette information.

The idea for our depth estimation method is that we want to design a local depth estimation strategy and limit spatial coherence constraints. This comes from the fact that the MVS problem for performance capture in a large capture area differs from the standard MVS in many ways. The main problem is the expected shape detail level compared to the observations. Standard static reconstruction methods most often consider hundreds of points of view, with the object correctly centered in the images. In our case, we want to capture larger areas without moving the cameras, thus needing smaller focal lengths, making every subject reproject on small portions of the images. Applying strong regularization priors while constructing the depth maps is a very successful strategy for standard MVS, *e.g.* recent works of [63, 57] but does fail in our scenario, as shown in the experimental comparisons in chapter 6.2.2. The philosophy behind our local strategy is to extract noisy depthmaps containing high frequency details, and let the subsequent fusion step extract detailed and denoized shapes.

4.1 Single Depth Estimation

Our objective here is, for every pixel of every camera, to find the distance to the capture subject. That is, a positive real value. We can safely assume that the subject lies inside the capture platform, such that possible depths also have an upper bound. The strategy now consists in discretizing the range of possible depths, and pick the most probable candidate according to some procedure. Different works tackled this problem in the MVS scenario and proposed solutions to this problem with coherence constraints on the neighborhood via graph-cuts [127, 126, 107] or via normal diffusion [44]. We tried in our experiments to reimplement the graph-cut strategy with no success for the performance capture scenario. On the other hand, in our case the silhouettes of the observed shapes can be recovered with a background extraction procedure, allowing us to mask pixels that do not carry information about the observed subject. The following sections explain how to use them further, and restrict the search of possible depths along rays to a small volume.

4.1.1 Confidence Volume

When available, the silhouettes $\{\Omega_i\}_{i=1}^{N}$ define, by extrusion, a 3D visual hull that is assumed to contain the observed object. In practice, silhouettes are prone to various errors such as holes or missing parts and do seldom guarantee this containment property with the visual hulls. In addition, our objective is primarily to reduce the search space along viewing rays to segments that are likely to intersect the object surface more than exactly locate the visual hull. Consequently we define the confidence volume V as:

$$\begin{aligned} V = \{x \in \mathbb{R}^3 :\ & \exists^{>\alpha} i\,(\pi_i(x) \in I_i) \\ & \wedge \exists^{>\beta} i\,(\pi_i(x) \in \Omega_i)\}, \end{aligned} \tag{4.1}$$

that is the locus of points in $\mathbb{R}^3$ for which there exist $i > \alpha$ images where they project and $i > \beta$ silhouettes to which they belong. α, β are two user defined constants that restrict weakly supported depth predictions with α and enable predictions away from the exact visual hull when $\beta < \alpha$. Intuitively, V is a dilated version of the visual hull in the space region seen by at least α images, as shown in fig 4.2.

4.1.2 Depth Prediction

For each pixel in every silhouette, depth is predicted along the viewing line using maxima of the photoconsistency measure ρ introduced before. As mentioned before, the photometric information can often be unreliable in mid-scale scenarios. In order to prevent false detections of maxima far from the surface, we adopt a conservative scheme where search for maxima along the viewing rays start from the confidence volume and stop when the accumulated photoconsistency reaches a threshold, hence limiting surface penetration along rays. In spirit, this is similar to [91] who define and integrate interior probabilities along rays using however a photoconsistency measure taken from [127] (see the discussion on photoconsistency measures in the previous paragraph).

More precisely, the best depth candidate d_i^p along ray $r_i(p, d)$ leaving camera i through pixel p is determined as:

$$d_i^p = \begin{cases} d_V(p) \text{ if } \max\limits_{d \in [d_V(p), d_{max}]} \rho_i(r_i(p,d)) < \tau_{photo}, \\ \operatorname{argmax}\limits_{d \in [d_V(p), d_{max}]} (\,\rho_i(r_i(p,d))) \text{ otherwise.} \end{cases} \tag{4.2}$$

Where $d_V(p)$ is the first depth value along $r_i(p, d)$ inside the confidence volume V, τ_{photo} a minimum photoconsistency value below which we fall

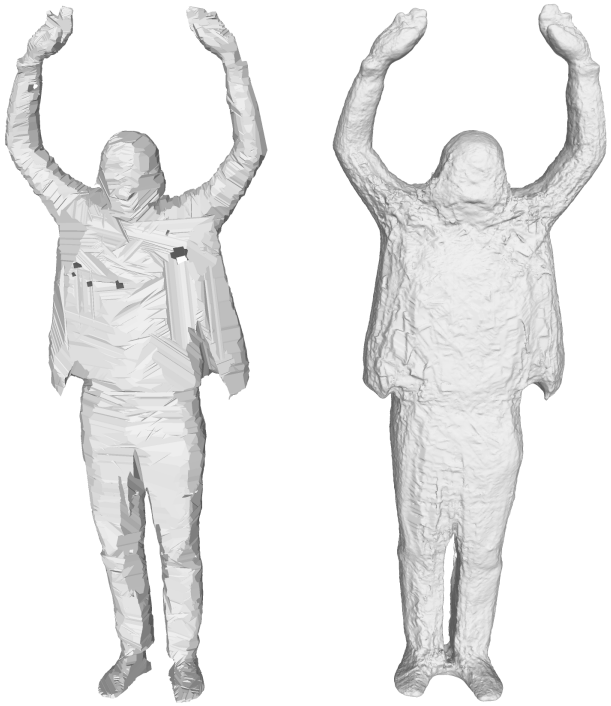

Figure 4.2: Left: the Confidence Volume with $\alpha = \beta = 54$, equivalent to the Visual hull with the 54 cameras that see the subject; Right: the Confidence Volume with $\alpha = \beta = 10$.

back to silhouette information and the confidence volume, and d_{max} the search limit such that:

$$\int_{x=d_V(p)}^{d_{max}} \rho_i(r_i(p,x))\mathrm{d}x \leq \rho_{max} \tag{4.3}$$

As shown in figure 4.3, the noisy photoconsistency in the performance capture scenario leads to a lot of extreme holes in the reconstructions when not using the accumulation scheme, *i.e.* $\rho_{max} = \infty$ (*top row*). The addition of this term ($\rho_{max} = 1.6$) allows for smooth reconstructions, still containing most of the important geometric details (*bottom row*). This term is very similar in spirit to the thresholding parameter in space carving works. In contrast to these methods however, we combine it with a view dependent local maximum, and the depth fusion process (see 5 coupled with a ray-casting based sampling (see 6) ultimately leads to a clean and smooth reconstruction. The value empirically chosen provides a good tradeoff be-

tween details recovery, robustness and visual quality of the reconstructed models in the Kinovis platform.

Finally, neighboring pixels that share a similar appearance are most likely to have similar depths. For this reason, and also to speed up depth map computation, we first perform super pixel clustering on images using SLIC [10] and select a few random samples per super pixel. An exhaustive search (full possible depths range) is performed for these sample pixels in order to provide an approximation for depths within the super pixel. Other pixel depths in the super pixel are then computed around this first approximation $\bar{d}$. This adds spatial consistency to the depth estimations by enforcing similar depths to be close to each other.

4.1.3 Depth Map Filtering

To speed up depth maps computations, we limit the search along a viewing ray to $5mm$ around a coarse depth estimation based on image descriptors [114]. Depths are sampled every $0.5mm$. This helps enforcing some spatial coherence between neighboring depths that share the same appearance. As a post processing step, we simply add a soft bilateral filter, similarly to [51], accounting for color, spatial neighborhood, and probability of the detection. It efficiently filters out outliers with little impact on the computation burden, which motivates our choice in a 4D dynamic context.

4.2 Evaluations

We first provide in this section a quantitative evaluation of our depth estimation method. To this aim, we use the DTU Robot Image Dataset [62], equipped with structured light scans, with a $0.5mm$ measured accuracy. We use point to point accuracy and completeness with the code provided by the authors. For the quantitative evaluations, best results were constructed with $\rho_{max} = \infty$, *i.e.* without accumulation along rays. The interest of this accumulation scheme will then be studied in the qualitative evaluation, where we demonstrate the importance of this brick for the performance capture scenario.

4.2.1 Quantitative Evaluation

We compare to Furukawa et al. [41], Campbell et al. [22] and Tola et al. [115], as well as to additional learning-based results from Ji et al. [63] and Hartmann et al. [51]. To conduct a fair comparison with [51], which is a

Table 4.1: Reconstruction accuracy and completeness (in mm).

	Acc.		Compl.	
Measure	Mean	Med.	Mean	Med.
Tola et al. [116]	**0.448**	**0.205**	0.754	0.425
Furukawa et al. [41]	0.678	0.325	0.597	0.375
Campbell et al. [22]	1.286	0.532	**0.279**	**0.155**
Ji et al. [63]	0.530	0.260	0.892	0.254
Ours (*fused*)	0.490	0.220	0.532	0.296
Hartmann et al. [51]	1.563	0.496	1.540	0.710
Ours (*depthmap*)	**0.599**	**0.272**	**1.037**	**0.387**

patch based approach building a depthmap with a network comparable to ours, we use the result of our volume sweeping approach on only one depth map.

Reconstructions results are depicted in table 4.1. We obtain quality on par with other methods, with a median accuracy and completeness in the range of the ground truth accuracy that we measured around $0.5mm$. It should be noticed that the best accuracy is obtained by Tola et al. [116] which tend to favor accuracy over completeness whereas Campbell et al. [22], in a symetric manner, tend to favor completeness over accuracy. We obtain more balanced results on the 2 criteria, similarly to the widely used approach by Furukawa et al. [41], with however better performances. We also outperform the recent learning based method Surfacenet [63] on most measures in this experiment.

Compared to Hartmann et al. [51], and under similar experimental conditions, our approach obtains better results with 2 orders of magnitude less parameters, thereby confirming the benefit of volumetric receptive fields over planar ones. Compared to Surfacenet [63] (cube size $64 \times 64 \times 64$, sample step $0.4mm$) we obtain reconstructions of slightly better quality with an order of magnitude less parameters.

4.2.2 Qualitative Evaluation

We will study in this section the impact of probability accumulation along rays presented in equations 4.2 and 4.3. We refer to section 6.2.2 for qualitative comparisons with state-of-the-art strategies.

Accumulation term While the accuracy and completeness metrics in the standard MVS scenario of [62] are negatively impacted by this accumu-

lation, it becomes of great importance in the performance capture scenario. We tried with and without the accumulation term for reconstructions of real data captured with the Kinovis platform, and show the results in figure 4.3.

4.3 Conclusions

We explained in this chapter how to make use of the photoconsistency term described in chapter 3 to build depth maps, by first, providing a method to restric depth search, then deriving a method from a standard *winner-take-all* strategy. Our quantitative experiments show that our depth estimation scheme coupled with the learning strategy achieves competitive results on a standard static MVS dataset on which the network was trained, with orders of magnitude less parameters compared to other learning strategies. We also provided an ablation study of our accumulation scheme, showing the importance of this brick in the pipeline for the performance capture scenario. Our depth map strategy is designed to be local, with only small spatial regularization. The philosophy is to let the depth maps capture high frequency details among with noise and let the subsequent fusion step denoise the result. For this reason, we only implement a small bilateral filter and do not enforce strong explicit spatial constraints. These depth maps will then be used to define an implicit form from which can be extracted the 3D shape as explained later in Chapters 5 and 6.

Figure 4.3: Demonstration of the importance of the accumulation scheme in the performance capture scenario. (*top*) Input image, (*midlle*) reconstruction without accumulation, (*bottom*) reconstruction with accumulation. It is visible here that the latter provides smoother and more accurate details.

Chapter 5

Temporal Integration

In this chapter, our objective is to exploit visual cues on dynamic scenes over both space and time in order to recover high precision shape models. As explained in the introduction 1, we particularly consider mid-scale dynamic scenes. We thus favor multi color camera apparatus as they provide flexibility in the acquisition space and time resolution.

Our approach aims at exploiting temporal redundancy over a sliding time window in a sequence of multi-view frames. Within such a time window, we propagate depth cues between frames and gather them over a single shape instance, referred to as "*canonical model*". This allows us to gather more information about the shape and extract, from this canonical model, a denoised and refined reconstruction at every time step. Since the scene is undergoing complex arbitrary motion, one of the main challenges lies in finding a way to robustly accumulate these observations.

We propose a local filtering strategy, similar in spirit to Non Local filtering, where we do not seek to build a full 4D model, rather we propose to improve 3D shapes where we safely can. Our strategy consists in estimating a first reconstruction at every time-step, and use this to approximate scene's motion by finding sparse correspondences. These estimations are then densified and the resulting flow field is used to refine the reconstructions that can then provide better and denser sparse motion estimations. We propose an iterative scheme, alternating between motion estimation and shape refinement and we show that it improves reconstruction accuracy by considering multiple frames. To this purpose, and in addition to real data examples, we introduce a multi-camera synthetic dataset that provides ground-truth data for mid-scale dynamic scenes.

5.1 Motion Estimation

Considering two meshes S^k and S^l at frames k and l, we want to estimate the volumetric motion field $W_k^l : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that maps S^k into S^l. Recall that our objective is to improve shape estimations, hence we do not necessarily need the complete shape motion, as when tracking or estimating scene flow. Instead, we look for reliable sparse motion information in surface regions where temporal integration will therefore benefit to the shape reconstruction. Thus, the estimated 3D motion fields need not fully reproduce the true motion. We equip it with confidence measures that identify valid motion and allow to neglect the surface cues associated with invalid motions when propagating information between frames.

Various methods have been proposed to recover motion information on moving shapes. Depending on the prior assumption on the motion model they range from weakly constrained models with scene flow [122] to locally rigid models with ARAP[14] strategies, as for instance with Kinect and Dynamic fusion[61, 90] or [26] and, at the other end of the spectrum, to stronger priors with articulated models and skinning animations as in [124].

In our context, as we do not seek for a complete and flexible motion model we will favor locally constrained strategies. Another strong aspect of this strategy is that local motion information is more likely to retain finer details compared to global motion estimation that most of the time needs strong regularization. In addition, since we consider mid-scale and dynamic scenes, large displacements can occur between frames which advocates for sparse but robust matching. We therefore opt for 3D features to provide robust 3D matches that will be progressively densified over the alternate iterations of shape and motion estimations. We use MeshHog [138] to detect and match 3D features as it demonstrates a good tradeoff between robustness, completeness and accuracy among other efficient methods such as heat kernel [111] or Harris 3D [108].

Let $\{M^k\}$ be the set of corresponding pairs of 3D features between S^k and S^{k+1} obtained with MeshHog and $m \in \{M^k\}$ such a pair. We attach to m a confidence measure λ_m that favors regions with dense and coherent matches. To this aim, the k-nearest neighbors m_j of m in $\{M^k\}$ are first computed. Let δ_m^j be the discrepancy between the displacement vectors associated to m and m_j. λ_m is then the median of the j values $\mathcal{G}(\delta_m^j)$, where $\mathcal{G}$ is a Gaussian kernel. This conservative strategy favors small regions on S^k where m and its neighbors present similar displacements vectors. As more matches will be added over iterations (see figure 5.5), this can be seen as a growing strategy that progressively extends the motion field around regions where consistent displacements are found over iterations.

Corresponding pairs of MeshHog feature $m \in \{M^k\}$ can be seen as displacement vectors $\{T_m\}$ from S^k to S^{k+1} and given their confidences λ_m, we define the forward motion field $W_k^+ : \mathbb{R}^3 \to \mathbb{R}^3$ and its confidence $\lambda_k^+ : \mathbb{R}^3 \to \mathbb{R}$ as:

$$\begin{aligned} W_k^+(x) &= \sum_{m \in \{M^k\}} \lambda_m \, \mathcal{G}_m(x) \, T_m, \\ \lambda_k^+(x) &= \tfrac{1}{|M^k|} \sum_{m \in \{M^k\}} \lambda_m \, \mathcal{G}_m(x) \end{aligned} \tag{5.1}$$

where $\mathcal{G}_m()$ is a Gaussian kernel that weighs the contribution of m with respect to the spatial distance between x and the feature of m on S^k. The backward motion fields $W_k^-(x)$ that maps S^k onto S^{k-1} is defined in a similar way using MeshHog features between S^k and S^{k-1}. The motion field W_k^l and its confidence λ_k^l are then defined as:

$$W_k^l(x) = \begin{cases} \sum\limits_{t \in [k, l-1]} W_t^+(x) & if \; k < l, \\ \sum\limits_{t \in [k, l+1]} W_t^-(x) & if \; k > l, \\ 0 & if \; k = l, \end{cases} \tag{5.2}$$

$$\lambda_k^l(x) = \begin{cases} \prod\limits_{t \in [k, l-1]} \lambda_t^+(x) & if \; k < l, \\ \prod\limits_{t \in [k, l+1]} \lambda_t^-(x) & if \; k > l, \\ 1 & if \; k = l, \end{cases} \tag{5.3}$$

5.1.1 Spatial Integration

To introduce our integration scheme, we first consider a single frame and the spatial integration of the depth maps d_i for all cameras at that frame. Following several works [29, 61, 90] with a similar objective but in different contexts, *e.g.* small-scale, we fuse all the depth maps into a 3D implicit form and take benefit of the Truncaded Signed Distance Function (TSDF) strategy for that purpose. Our motivation for the TSDF comes from its ability to naturally handle arbitrary depth maps arising from different cameras in addition to different time steps, as shall be dealt with in further sections.

For a point $x \in \mathbb{R}^3$, the truncated signed distance $TD(x) \in \mathbb{R}$ to the surface is defined as the weighted average of all camera predictions $F_i(x)$

with $i \in C$:

$$F_i(x) = \begin{cases} \min(\mu, \eta(x)) & if\ \eta(x) \geq -\mu, \\ \emptyset & \text{otherwise}, \end{cases} \\ \eta(x) = d_i(\pi_i(x)) - \|c_i - x\|, \tag{5.4}$$

and:

$$TD(x) = \frac{\sum\limits_{i \in C_x} \rho'_i(x) F_i(x)}{\sum\limits_{i \in C_x} \rho'_i(x)}, \tag{5.5}$$

where $C_x = \{i \in C : F_i(x) \neq \emptyset\}$ and ρ'_i the photoconsistency measure (3.2) of the estimated depth along the ray passing through x. If d_i is undefined at x, *e.g.* x is outside the camera visibility domain, then camera i does not contribute to the TSDF. When no camera contributes at x but x is inside the confidence volume V then it is considered as inside, *i.e.* $TD(x) < 0$. Note that contributions are weighted by the photoconsistency measure which means that when cameras disagree about the photoconsistency at x, cameras with higher measures have an increased impact whereas cameras with low photoconsistency measures only marginally impact the reconstruction. It can be noted that several measures exist for depth confidence, as listed and evaluated in [55]. In our case, we did not try these as photoconsistency already gave satisfactory results.

5.2 Spatiotemporal Integration

In order to extend the previous spatial integration 5.1.1 to the time domain, we now consider several frames over a temporal window $T = [k - n/2, k + n/2]$ of size n around frame k. In essence, the temporal integration consists then in adding to the TSDF (5.5) depth contributions from the neighboring frames; using to this aim the estimated motion fields $W_k^t : \mathbb{R}^3 \to \mathbb{R}^3$ that map frame k to frame l (as detailed in Sec. 5.1). As mentioned earlier, these contributions should be weighted by the confidence λ we have in the estimated local motion in addition to their photoconsistencies ρ. We define therefore the integrated implicit form $\overline{TD}_k : \mathbb{R}^3 \to \mathbb{R}$ of the observed shape at frame k as:

$$\overline{TD}_k(x) = \frac{\sum_{t\in T} \lambda_k^t(x_k^t) \sum_{i\in C_x^t} \rho_i'^t(x_k^t) F_i^t(x_k^t)}{\sum_{t\in T} \lambda_k^t(x_k^t) \sum_{i\in C_x^t} \rho_i'^t(x_k^t)}, \tag{5.6}$$

$$x_k^t = x + W_k^t(x). \tag{5.7}$$

where $C_x^t = \{i \in C : F_i^t(x) \neq \emptyset\}$ and λ_k^t, ρ_i^t, d_i^t and F_i^t are respectively the motion confidence (Sec. 5.1), the photoconsistency measure (Sec. 3.1.1), the depth prediction (Sec. 4.1.2) and the truncation function (Sec. 5.1.1) at frame t.

5.3 Implicit Form Representation

Given the depth maps $\{d_i^t\}$ estimated for all cameras i and all frames t, we can now fuse depth information over space and time to recover the shape surface mesh S^k at any time instant k. While we consider all cameras in the fusion, we limit the frames taken into account to a temporal window around k, typically 3 to 7 frames in our experiments, within which enough required shape motion information can be obtained with precision. In order to propagate reliable depth cues between frames, our approach seeks for local regions with consistent displacements and high photoconsistencies. This local strategy better prevents the propagation of wrong depth cues which occurs when a global strategy, such as template tracking, is used. Given a temporal window, we assume that each frame t, within the temporal window, corresponds to an instance of the reference shape S^k deformed with respect to a 3D motion field W_k^t, with no topology assumption. The approach consists then in iterating the following steps:

1. For all frame k:
   a) Given inter frame volumetric motions $\{W_k^t\}$ merge all the time window depth maps, warped using $\{W_k^t\}$, into a 3D implicit form.
   b) From the implicit form estimate the 3D mesh S^k.
2. Given the $\{S^k\}$ estimate the motion fields $\{W_k^t\}$.

To initialize the process, we perform spatial integration only in the above step 1 at the first iteration. The two steps are then repeated a few times, typically 3 in our experiments.

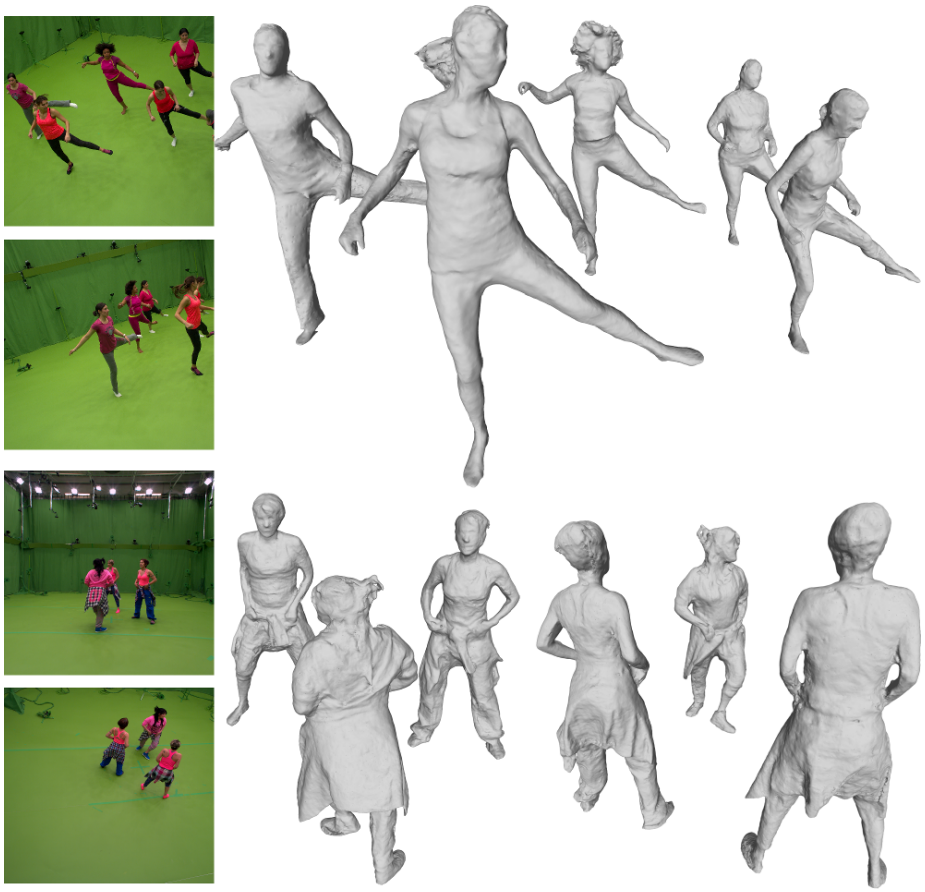

Figure 5.1: Examples of challenging dynamic mid-scale datasets, and our reconstructions.

We can see this filtering process as a 4D equivalent of image based Non Local filtering [20], where we iteratively refine the association confidence with an additional prior over the undergone motion. The deformation prior of this strategy is a scale-invariant local translation model.

5.4 Evaluations

In order to demonstrate the benefit of time integration to recover dynamic scene models we conducted different experiments. First, quantitative results were obtained to evaluate how temporal integration improves shape reconstruction. To this purpose, and since dynamic multi-view benchmarks are not available yet, we created a dynamic dataset equipped with ground

truth data on geometry and appearance. Then, qualitative results on real data were also obtained to illustrate that temporal integration enhances reconstructed shapes quality. The code and all the data used in the following experiments is available to the community at the following address: http://deep4dcvtr.gforge.inria.fr/.

5.4.1 Synthetic Data

Dataset Multiple benchmarks addressing the Static Multi-View Stereo problem, *e.g.* Middleburry [104] or DTU Robot Image Dataset [62], were already made available online. However, to the best of our knowledge, none exists for the dynamic case with surfaces evolving over time. Hence, we built an evaluation dataset with the objective to be as close as possible to real situations with real data while having ground truth information. It should be noticed that such ground truth data is of interest in a context larger than shape recovery and can contribute to tracking or appearance modeling evaluations. The data consists of procedurally generated surfaces, typically clothes, added to real captured data, typically body shapes, for which tracking over time sequences are available. Its main features are:

- The synthetic image generation set-up is similar to real multi camera platforms.
- Underlying shapes and their motions are real captured data and replicate therefore real dynamic situations.
- Local shape deformations are generated and can simulate clothes or any other type of deformation.
- Appearances are generated as well and can yield various effects with low to high contrast textures, specular surfaces, color diffusion, motion blur among others.

Evaluation Given the ground truth data mentioned above we evaluated quantitatively shape reconstructions using standard measures in the field [104, 62], *i.e.* accuracy and completeness. Static and refined reconstructions were performed on a 20 frames synthetic sequence with local clothing deformations, observed by 60 cameras, with a capture volume of approximately 8mx4mx6m.

Figure 5.2 demonstrates how the mean completeness (ratio of ground truth points closer to the reconstruction than a given error) over 10 frames increases with temporal window of sizes 1, 5 and 7. In order to evaluate the

benefit of our local propagation strategy, we also performed comparisons with a strategy based on global surface tracking between adjacent frames [21] very similar in spirit to the tracking method employed in [32]. The global motion was then fed in our temporal integration pipeline similarly to our local strategy. All experiments were conducted using the same set of parameters. Figure 5.2 shows that such global strategy (mesh tracking in the figure) performs worse than our local strategy or even than static strategies (*i.e.* single frame). This is confirmed on real data in Figure 5.4 where the mesh tracking based strategy is prone to erroneous and imprecise estimations, leading to an oversmoothed results.

For the sake of completeness, we also compare to [43], top ranked static Multi-View Stereo Reconstruction method on the DTU dataset [62]. While the accuracy comparison would be unfair since [43] does not take silhouettes into account and hence produces points outside the visual hull, we believe that the completeness that measures how close the ground truth is to the reconstructed surface is on the other hand informative.

This figure also shows min and max completeness values over 20 frames of the synthetic sequence. It shows that the temporal integration impact significantly more the min completeness. It is worth noticing that at approximately the pixel resolution, roughly 3mm here, the min completeness is increased by around 15% with the temporal integration.

5.4.2 Real Data

We also tested our method on different dynamic multi-camera sequences, containing multiple subjects. Every sequence was captured with 68 calibrated RGB cameras (2048×2048) with focal lengths between 8 and $28mm$. Some examples of dynamic mid scale scenes and spatiotemporally refined surfaces are shown in Figure 5.1.

Figures 5.3 and 1.6 depict input images, our reconstructions and the temporal improvement for the former. In addition, Figure 5.3 shows that the temporal refinement preserved details that are filtered out by a spatial smoothing technique (HC Laplacian Smoothing [128]).

Figure 5.4 shows an example of temporal integration with a global mesh tracking strategy, as explained previously. Even though the standing subject is quite well reconstructed, such global approach fails in the case of fast motion and strong topology noise. The temporal integration with a global template motion makes the moving subject's surface noisier and fast moving parts are missing. The thin surfaces such as the belt and the outfit also tend to suffer from the tracking inaccuracies propagated through time and are not correctly recovered with the global mesh tracking strategy.

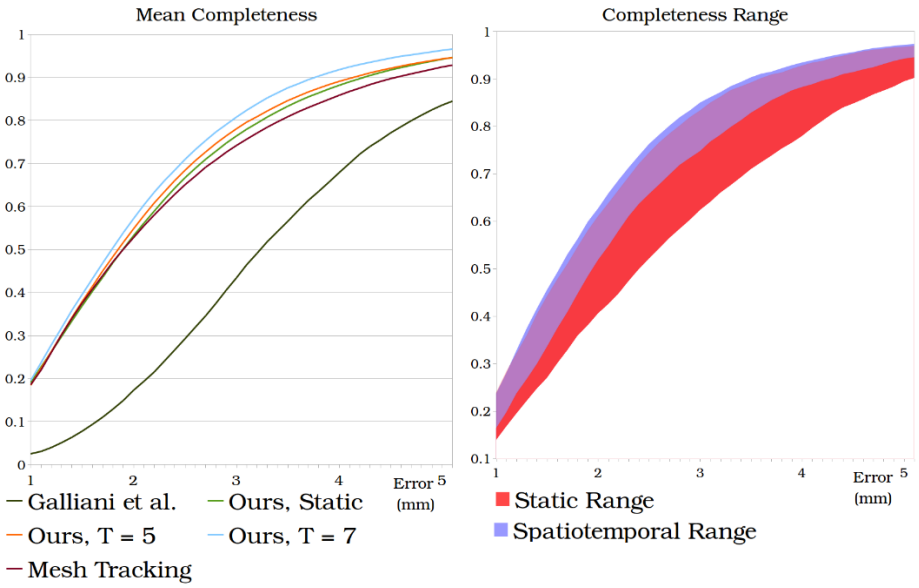

Figure 5.2: (*left*) Mean completeness comparison between [43] and our reconstructions on 10 frames of the synthetic sequence, (*right*) Min and max values of completeness on 20 frames of the synthetic sequence, time window T = 7, iterations = 3.

5.4.3 Region Growing

To better illustrate the region growing behavior of our strategy, we study the number of matches found by MeshHOG over the iterations in Figure 5.5. We can clearly see that number of matches almost doubles over the first iterations but seems to stop increasing after 3 or 4. In our experiments, more than 3 iterations did not lead to much better improvements

Failure case: Even though the shape descriptor matching process proved to be robust to noise, no noticeable improvement could be observed over iterations when the initial reconstructions were too noisy. Tweaking the Gaussian kernel width and confidence acceptation parameters would only lead to shape oversmoothing with no particular interest over standard Laplacian smoothing. Divergence was seldom observed as when the model diverges, wrong descriptor matches are discarded by our filtering scheme, falling back to the per-frame reconstruction result.

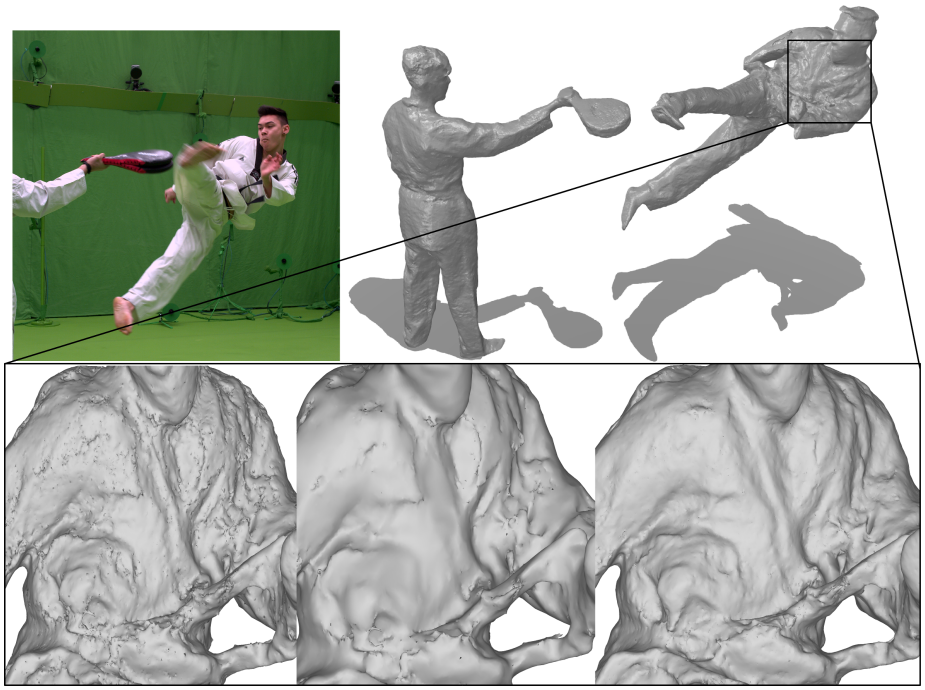

Figure 5.3: (*top*) An input image and our refined reconstruction. (*bottom*) A close-up view on the model, showing the static reconstruction (*left*), spatially smoothed using [128] (*middle*) and our temporal details refinement (*right*). Best viewed magnified

5.4.4 Implementation

Our C++ multithreaded implementation runs as follows on a 16-core Xeon 3.00GHz PC, 32 Gb RAM and with 68 cameras of 2048×2048 pixels: 5-20 min/frame to build the implicit TSDF, depending on total number of silhouette pixels; 5 min/frame for motion estimation; 5 min/frame for the surface extraction, for a final mesh of 3M faces. A GPU implementation could be considered as extension for significant speedup.

5.5 Conclusions

We described in this chapter a method for temporal filtering adapted to the performance capture scenario. Our goal was to gather neighboring frames into our shape representation. For the shape details to be correctly propagated, we first defined how to estimate the local motion. We then used this estimation to refine the shape and alternate between these steps to

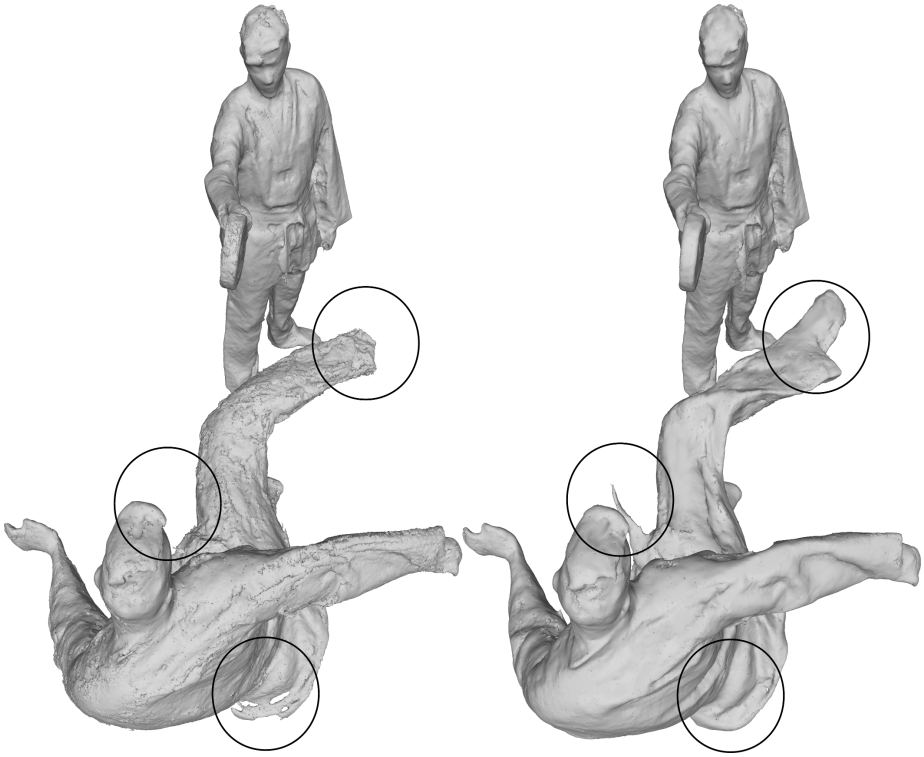

Figure 5.4: Spatiotemporal integration using motion estimation based on global surface tracking (*left*) and using the proposed local detection approach (*right*).

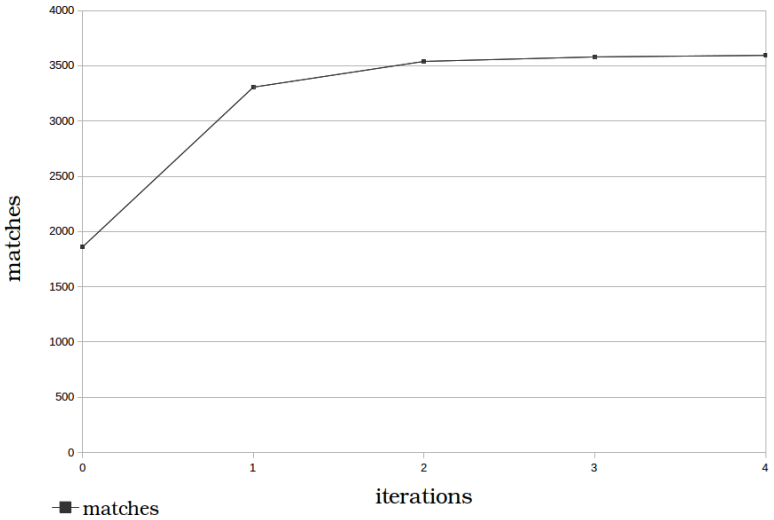

Figure 5.5: Number of found matches over iterations on a performance capture example.

iteratively improve the quality of our reconstructions. We showed both quantitative improvements on a synthetic dataset, and qualitative improvements on performance capture data. The whole refinement strategy was designed to be scalable to larger scenes, thus discarding the use of regular discretizations such as voxel grids. Our method takes into account geometry and appearance to locally transfer details and provide smoother and more detailed reconstructions. The following chapter 6 explains how to extract a surface mesh from our shape representation.

Chapter 6

Surface Reconstruction

From the implicit forms of the shape detailed in the previous chapter 5.7, we now present how to extract the 3D shape mesh at frame k as the zero level set of the associated implicit function $\overline{TD}_k(x)$. A vast majority of methods consider the Marching Cube [78] (MC) approach for that purpose [41, 61, 91]. It is to be noted that variants of this method have been proposed, such as Marching Tetrahedra [117] or a learning based approach coined Deep Marching Cubes [76].

Although MC would also work in our case we consider instead a different strategy that addresses some of the limitations of MC: MC is based on a regular discretization of the space and hence dilutes precision inside the shape, unless a specific strategy such as subdivision is applied at the surface; MC is not guaranteed to provide manifold meshes, again unless specific and costly additional steps are performed. Novel surface extraction methods based on Delaunay tetrahedrization or its dual the Voronoï Tesselation have been proposed [56, 131] to overcome such limitations, showing that good precision can be obtained with discretizations of shapes instead of space.

We present in this chapter the surface extraction method we developped using a variation of [131] based on ray casting. We will then perform experiments to better enhance the strengths of our strategy and finally provide comparisons with multiple state-of-the-art MVS works on performance capture data.

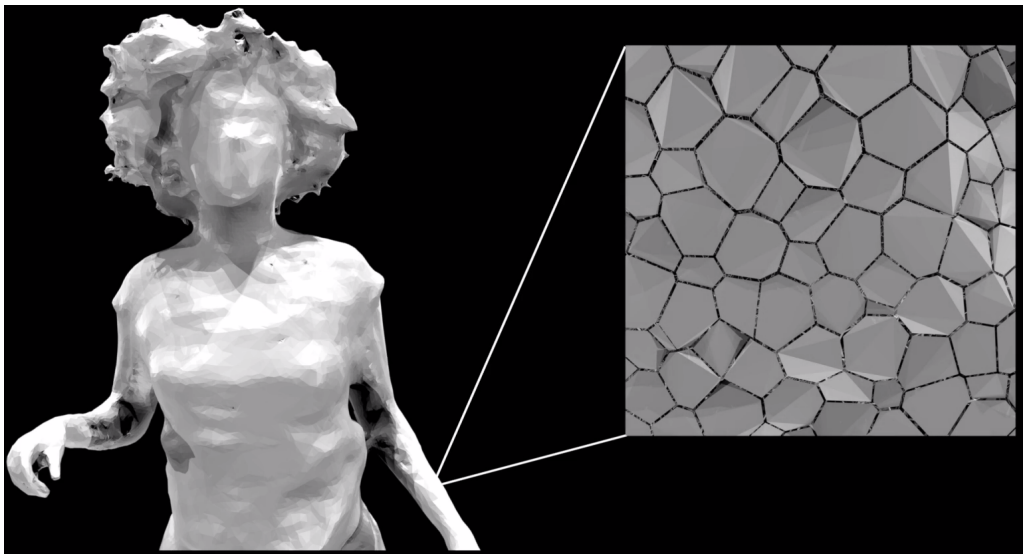

Figure 6.1: An extracted surface (*left*). We display the shrunk clipped faces (*right*) to better enhance the 3D Voronoï diagram lying underneath the triangular mesh.

6.1 Shape Mesh Generation

We devise a simple yet efficient version of Voronoi Tesselation that specifically accomodates mid-scale multi-view capture scenarios. The main steps of the algorithm are described in figure 6.2 and are as follows:

1. Sample points inside the implicit form defined by the TSDF. This is achieved by randomly selecting pixels in all images and computing the point, along each pixel rays, inside but close to the surface according to the TSDF. The process is iterated until a user defined number of 3D points is reached.

2. Determine the Voronoi diagram: given the points inside the shape surface, a Voronoi diagram of this set of points is computed.

3. Clip the Voronoi diagram with the zero level set of the TSDF. This operation extracts the intersection of the Voronoi cells with the surface.

In the above strategy, sampling points close to the surface, and originating from image viewpoints, ensures that the 3D discretization is denser

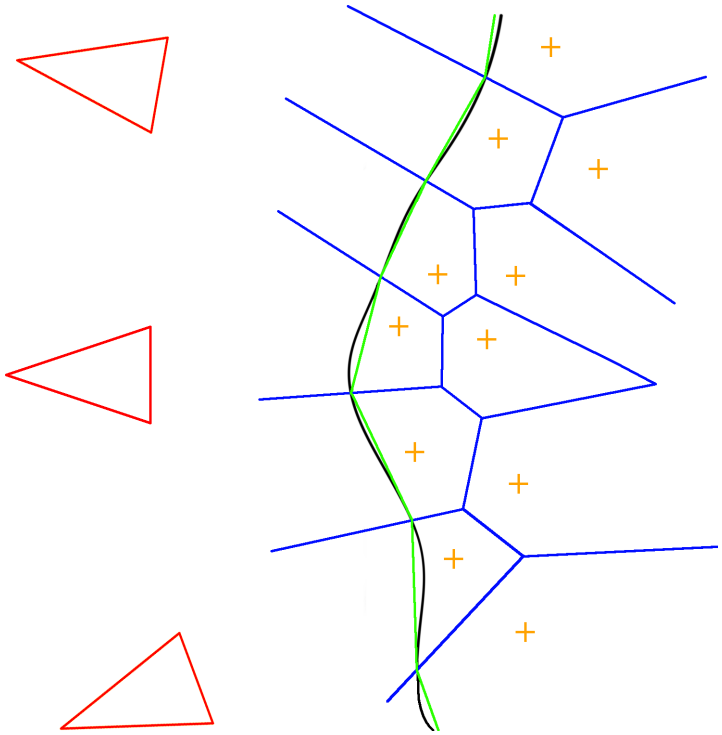

Figure 6.2: Our surface extraction procedure. The zero level of the implicit form (**black**) is observed by different cameras (**red**). They are used to provide the inside samples (**orange**) that will be used as the centroids for the Voronoï tessellation. This tessellation is finally clipped at the zero-level set and the final surface (**green**) can be extracted.

on the surface than inside the volume and also denser on surface regions observed by more images. The latter enables more precision to be given to surface regions for which more image observations are available.

We visualize in figure 6.1 an example extracted surface. To better illustrate the extraction process of the zero level-set, we shrink the clipped faces of the cells.

6.2 Experiments

In this section, we will first present experiments on our surface extraction strategy in order to validate the claims of this chapter 6.2.1. Then, we provide overall qualitative results on various performance capture datasets, both passive 6.2.2.1 and active 6.2.2.2 setups, in which we compare our

results to multiple state-of-the-art reconstruction methods, handcrafted [26] or learning based [63, 135].

6.2.1 Surface Extraction

First, we devise an experiment on synthetic data to verify the claims made in the previous section 6.1 about the sampling density variation due to observations. We design a synthetic capture setup similar to the ones described in 3 and 5. We capture a head model with 40 cameras and apply our static reconstruction method on it. Figure 6.3 shows two input views in the top row, and our results in the bottom rows. The bottom side of the bust is never seen by any camera. We show in figure 6.4 the difference in sampling depending on the observations. The horizontal bottom side of the model is never observed, yet still correctly reconstructed. On the other hand, the triangles of the mesh in that area are much larger than the ones in the vertical upper part, which is observed many times by the cameras. This strategy allows for complete reconstructions of captured shapes with an adaptative sampling density depending on the observations of the object, focusing more samples in the regions where the details can be recovered.

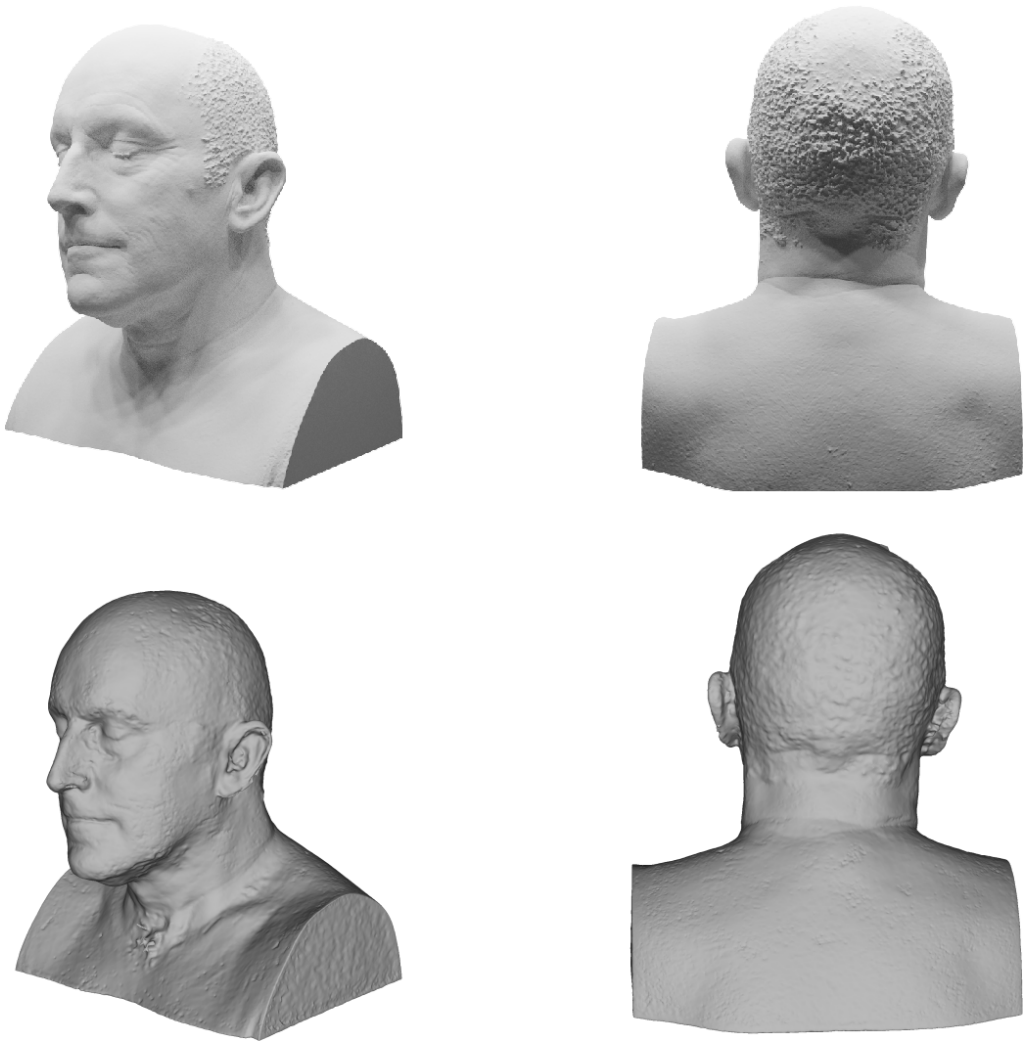

Figure 6.3: Two points of view of a synthetic model (*top*) and the result of our reconstruction (*bottom*).

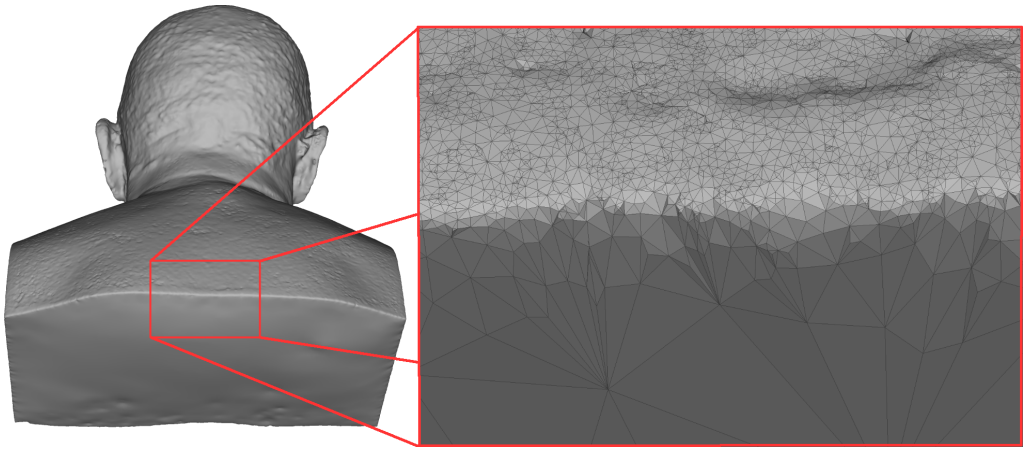

Figure 6.4: A close-up of the extracted surface (left) at the limit between well-observed and unseen regions. The top part of the close-up is seen by many cameras whereas the bottom part is never observed.

6.2.2 Performance Capture Reconstructions

This section aims at providing informative qualitative comparisons to state-of-the-art strategies. We especially focus on the performance capture scenario. To conduct fair experiments, since all other methods perform per-frame reconstructions, all the following reconstructions shown below are obtained using only the information available at a single frame, *i.e.* without temporal integration.

6.2.2.1 Kinovis Data

We first focus on data captured by [5], that is a hemispherical setup with 68 cameras of various focal lengths. In this scenario, standard MVS assumptions are often violated, e.g. specular surfaces, motion blur and occlusions, challenging therefore the reconstruction methods. A video demonstrating our results and providing comparisons on dynamic sequences is available online: https://hal.archives-ouvertes.fr/hal-01849286.

First, we compare with a recent learning based approach [63] using the code available online (see Figure 6.5). Reconstructions with this approach were limited to a tight bounding box and different values for the volume sampling step were tested. The best results were obtained with a $2mm$ step. To conduct a fair comparison with our method, all points falling outside the visual hull were removed from the reconstruction. In this scenario, the point cloud obtained using [63] appeared to be very noisy and incomplete

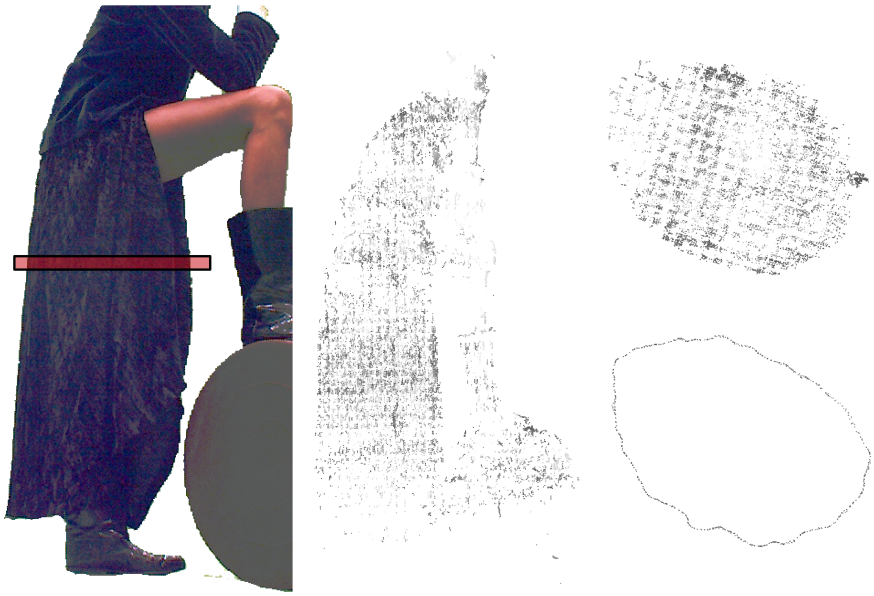

Figure 6.5: Qualitative comparison with [63]. (*Left*) input image with the horizontal section in *red*, (*middle*) point cloud with [63], (*right-top*) point cloud horizontal section with [63] (*right-bottom*) point cloud horizontal section with our approach.

(see Figure 6.5-middle), plaguing the subsequent surface extraction step. Figure 6.5-left also shows a horizontal section of the model in a poorly contrasted image region of the dress. The global strategy used in [63] wrongly reconstruct many surface points inside the shape volume (top figure), as a result of the ambiguous appearance of the dress. In contrast, our approach (bottom figure) correctly identifies surface points by maximizing learned correlations along viewing rays.

In addition to this, we also compare to results of [135] provided by the authors in Figure 6.6. This method outputs a rather dense point colored cloud but similarly to results from [63], extracting a smooth surface from this point cloud remains a difficult task due to strong noise and missing data. Since the method uses custom and undocumented calibration parameters, it was not straightforward to remove points lying outside the visual hull. Moreover, the precision of their point cloud restricts its usage for performance capture and realistic reconstructions rendering. Figure 6.7 provides a close-up of the face of a subject. The level of detail of their point cloud is not fine enough to correctly capture facial details, compared to the density of our output surface.

Figure 6.6: (*top*) Results provided by [135] on the kick 540 sequence. (*middle*) Poisson Reconstruction of their point cloud. (*bottom*) Our result.

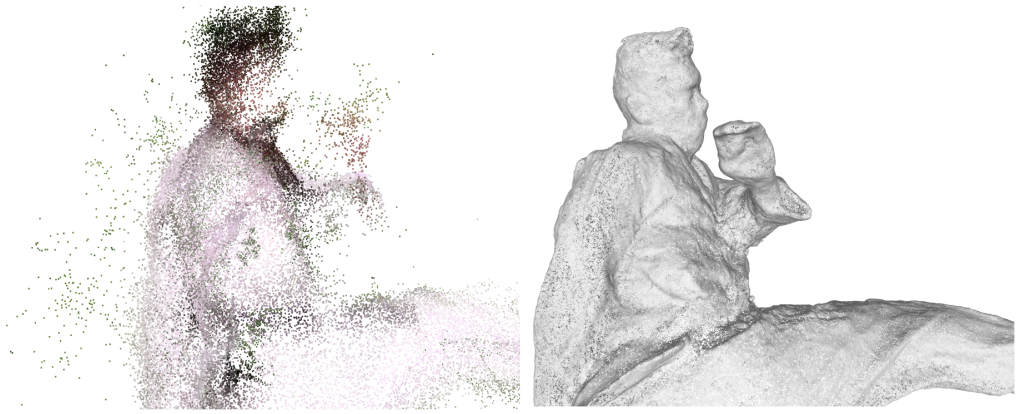

Figure 6.7: Point clouds density comparison between results provided by [135] (*left*) and our output (*right*). Best viewed magnified.

6.2.2.2 Active Capture Platform

Finally, we also performed reconstruction of a scene captured with the active system of [26]. This setup consists of 52 RGB cameras mounted as stereo pairs but also differs from the previous dynamic capture scenario, as it also features an active system, projecting random infrared dots on the shape. 52 infrared cameras, also paired on stereo rigs then capture the reprojected spots on the shape, resulting in highly contrasted images, allowing to disambiguate the photoconsistency computation, especially in textureless regions without interfering with the visible appearance of the subject. In figures 6.8, we compare to results provided by the authors. While [26] make use of all the data available, we restrict our method to work with RGB images only. On the other hand, we allow cameras that are far apart to participate in the computation of the photoconsistency. Our results demonstrate the potential of our method, showing detailed reconstructions on par with the results of [26] even though we only use the passive system, *i.e.* half of the available information. Figure 6.9 displays a close-up of the face of the subject. Our method allows to recover high-frequency facial details, such as the shape of the nostrils or the lips commissures, thus providing highly faithfull reconstructions.

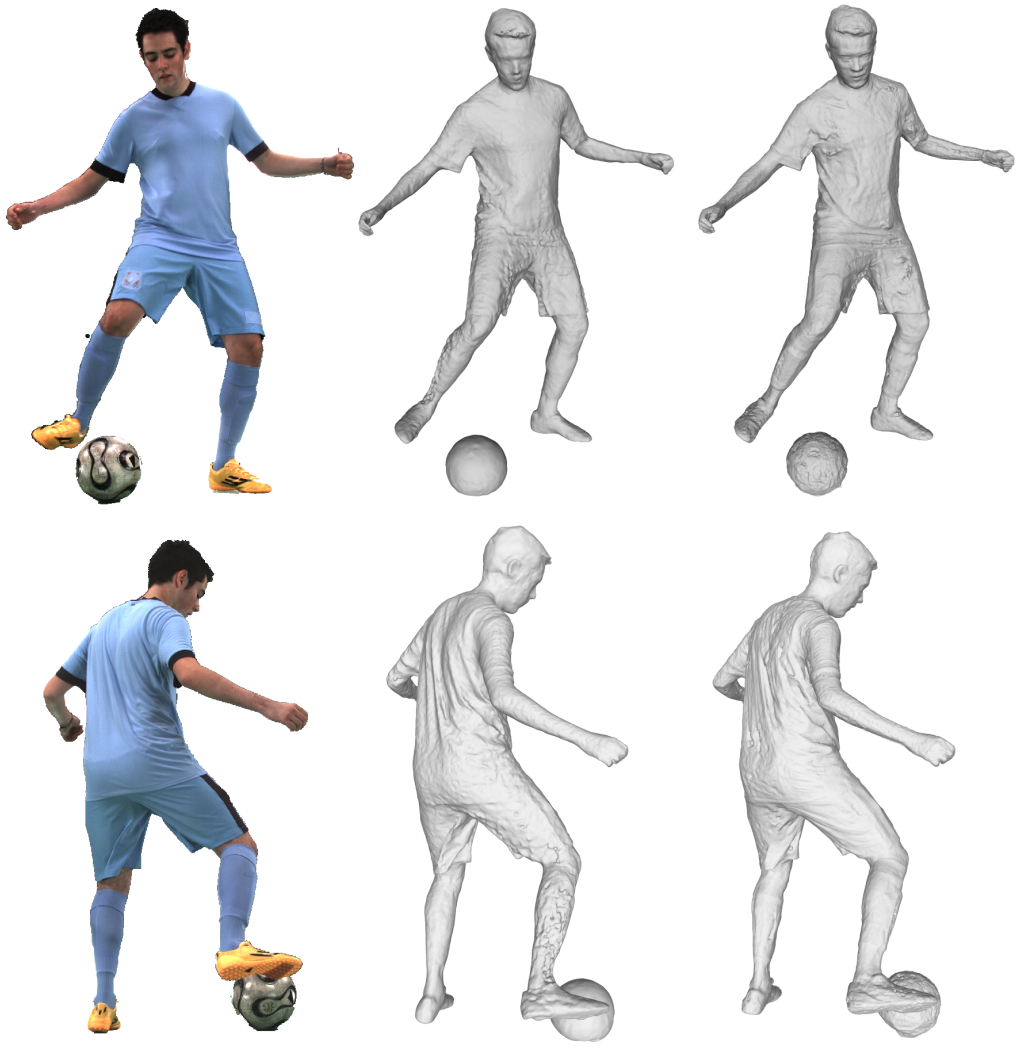

Figure 6.8: Two points of view of a subject from [26] (*left*). (*middle*) Reconstruction provided by the authors. (*right*) Results using our learning strategy.

Figure 6.9: Close up of the face of the subject from [26] (*left*). The reconstruction provided by the authors (*middle*) is very smooth compared to our result (*right*).

6.3 Conclusions

In this chapter, we presented a strategy to extract a surface mesh from the TSDF previously computed 5. We provided experiments to verify the claims of our approach, and compared our results to state-of-the-art MVS strategies. We want to emphasize that these qualitative results demonstrate the necessity of a specifically designed pipeline for the performance capture scenario. All general purpose out-of-the-shelf methods that were tested on kinovis data gave similar results in terms of noise and level of details that makes them less suited for this application. In contrast, our approach allows highly detailed reconstructions outperforming these methods, even though quantitative evaluations in chapter 4 show that we achieve similar quality on standard datasets. We also compared to [26] and showed that it was possible to achieve a similar level of details while only using passive information, *e.g.* 52 cameras. In contrast, their results were obtained using 52 supplementary IR cameras coupled with random patterns IR projectors. This highlights the importance of this work direction, that could possibly decrease the acquisition cost and/or allow for larger capture areas.

Chapter 7

Summary and Extensions

7.1 Summary

This manuscript described our contributions to the MVS pipeline and associated experiments, especially in the performance capture scenario. Our first contribution (3) lies in the surface detection step. We proposed two strategies to robustly detect surface points in space. The first one is a handcrafted method based on image gradient histogram descriptors. Second, we proposed to improve it with a learning based strategy. Trained on a standard static MVS database, our network displayed strong generalization properties, allowing for detailed and faithful reconstructions for the performance capture scenario, that strongly differs from the training data. Then, we presented a depth map estimation procedure (4) that allows us to filter the previous surface detection signal in a local manner. Next, we explained (5) how to transform the depth maps into an interior/exterior indicator with the TSDF and how to apply an iterative temporal filter over it to refine the underlying shapes. We finally explained how to extract a 3D mesh from this indicator at every time step (6), and compared our reconstructions to state-of-the-art methods both handcrafted and learning based.

The source code used in this thesis is made available to the community as an OpenSource project at the following address: `http://deep4dcvtr.gforge.inria.fr`. As stated in 1, the proposed pipeline was used for AR and VR applications for the fashion industry (see Fig. 1.2) and is fully integrated in the reconstruction pipeline for the Kinovis platform [5].

We want to emphasize that all our contributions are robust alternatives to standard bricks in the full MVS pipeline. The following paragraphs detail the different contributions for every brick.

Learning based photoconsistency In our first contribution we showed that, indeed, MVS for performance capture could benefit from the recent advances in machine learning. By designing a binary classifier trained to learn local properties of incident rays, our network is able to generalize correctly to unseen scenarios that differ a lot from the training database and for which no ground truth is available. We outperform state of the art methods on various datasets, and show a finer level of details in complex areas, allowing for more faithful reconstructions. The main conclusion of our work is that learning can help improve MVS in other scenarios when designed accordingly, *i.e.* focus on local properties with no high level semantics. Moreover, the network only focuses on low level configurations and implicitely takes scene geometry into account. Its asymetric design allows for a robust per-camera depth map computation, correctly considering visibility and allowing to gather information from any given number of neighbors, regardless of the input order, thanks to the average pooling step.

Sweeping Strategy We then presented a self contained sweeping pipeline for depth search in the performance capture scenario, that allows for robust and faithfull reconstructions by restricting search in a confidence volume, and accumulating surface presence probabilities along the rays. The confidence volume allows to perform complete reconstructions whith noisy silhouette information, preventing hole formations in the output shape when segmentation information is noisy and unreliable. The accumulation term helps dealing with the noisy photoconsistency information typically arising in the performance capture scenario as demonstrated in our experiments.

Temporal Integration Our iterative filtering strategy coupled with local assumptions on motion allows for a temporal smoothing, reducing noise and jitter. The proposed filtering strategy was shown to improve completeness and smoothness of the shapes while preserving fine details on both synthetic and real data.

Surface Mesh Extraction Instead of relying on a regular voxel discretization, we propose to keep a continuous TSDF and sample it with a ray casting variation of Clipped Centroïdal Voronoï Tesselations. This allows for a better scalability to larger scenes thanks to a discretization of the shape and not of the capture volume, with denser samples where more observations are available. We qualitatively outperform multiple state-of-the-art

MVS methods on performance capture data, demonstrating the relevance of every brick in our reconstruction pipeline.

7.2 Extensions

Unification of MVS and silhouette based reconstructions We believe that our volume sweeping approach is a first step towards a data-driven method to unify short baseline stereo and shape from silhouette, as our network is able to learn surface presence probability by gathering information from other points of view, implicitely encoding the setup geometry thus managing any kind of baselines. The provided synthetic experiments show that our approach is able to gather information from almost orthogonal cameras, considering both photoconsistency and contour information for surface detection. Investigating more into that direction could either help obtain better details in the reconstructions or decrease capture costs by lowering the number of cameras needed for a realistic result.

Shape Representation The main goal of this work was to reconstruct a surface observed by a multi-camera system. This geometry is then used to generate an appearance, in the form of a 2D texture map. The reason we make use of such 3D surface models is that it is an efficient and reliable way of encoding and compression of a shape and its appearance. This pipeline is well known and efficiently integrated in almost all existing graphics applications. However, recent advances in free-viewpoint rendering [52] or realistic re-rendering of captured scenes [84] showed that deep representations could provide rather low dimensional spaces to represent captured data aswell with less handcrafted constraints and better learned priors. We could easily imagine that it is possible to learn other latent spaces, that would enable us to compress information in a more efficient way and render captured scenes from any viewpoint in real-time in a lighter and more realistic fashion.

GPU parallelization A possible bottleneck of our pipeline lies in the computation time. A single frame reconstruction lasts between 20 to 45 minutes depending on the observed scene. Even though the inference is done in parallel on possibly multiple GPUs, most of the parallelization in the reconstruction process is done on CPUs. Similarly to [43] parallel computations, we could easily implement a fully parallel version leveraging the capacities of such GPUs and decreasing the computation times by a significant margin.

Continuous ray representation Nevertheless, the main weakness of the presented strategy lies in the discretization of the volume around a query point. This procedure involves a lot of redundancy and is a computationally expensive step for both training and inference. Moreover, even when optimized to process several neighboring depths in parallel, it remains memory inefficient. A possible future work could be to find a continous representation for rays crossing the volume of interest, that could be used to infer surface presence probability in a similar manner with a much lighter computational cost.

Dynamic Datasets Finally, supervised training in this scenario remains unfortunately limited by the available ground truth data. In fact, obtaining *ground-truth* information for supervised training in the performance capture scenario is still an open problem in the community. Two possible directions emerge from this: one may want to find new strategies to create such ground truth database. The other strategy consists in finding way to transfer properties learned on standard datasets equipped with ground truth, or design new unsupervised strategies to solve the problem using only the available data.

Bibliography

[1] Accute3d. https://www.acute3d.com/

[2] Dense multiview stereo evaluation dataset. http://cvlabwww.epfl.ch/data/multiview/

[3] Filmic blender. https://sobotka.github.io/filmic-blender/

[4] Intel capture platform. https://youtu.be/nd6vrSL7i1s

[5] Kinovis inria platform. https://kinovis.inria.fr/inria-platform/

[6] Middleburry multi-view stereo evaluation dataset. vision.middlebury.edu/mview/

[7] Middleburry multi-view stereo evaluation dataset. https://cvlab.epfl.ch/data

[8] Tum dataset. https://vision.in.tum.de/data/datasets/3dreconstruction

[9] Performance capture from multi-view video. In: Image and Geometry Processing for 3D-Cinematography (2010)

[10] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Ssstrunk, S.: SLIC Superpixels. Tech. rep., EPFL (2010)

[11] Aganj, E., Pons, J., Ségonne, F., Keriven, R.: Spatio-temporal shape from silhouette using four-dimensional delaunay meshing. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007 (2007)

[12] Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building rome in a day. In: IEEE 12th International Conference on Computer Vision, ICCV. pp. 72–79 (2009)

[13] de Aguiar, E., Stoll, C., Theobalt, C., Ahmed, N., Seidel, H.P., Thrun, S.: Performance capture from sparse multi-view video. In: ACM SIGGRAPH 2008 Papers (2008)

[14] Alexa, M., Cohen-Or, D., Levin, D.: As-rigid-as-possible shape interpolation. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, New Orleans, LA, USA, July 23-28, 2000 (2000)

[15] Bay, H., Tuytelaars, T., Gool, L.J.V.: SURF: speeded up robust features. In: ECCV (2006)

[16] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In: ECCV 2016 (2016)

[17] Bogo, F., Romero, J., Loper, M., Black, M.J.: FAUST: Dataset and evaluation for 3D mesh registration. In: Computer Vision and Pattern Recognition (CVPR) (2014)

[18] Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)

[19] Brown, M.Z., Burschka, D., Hager, G.D.: Advances in computational stereo. IEEE Trans. Pattern Anal. Mach. Intell. (2003)

[20] Buades, A., Coll, B., Morel, J.: A non-local algorithm for image denoising. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA. pp. 60–65 (2005)

[21] Cagniart, C., Boyer, E., Ilic, S.: Probabilistic Deformable Surface Tracking From Multiple Videos. In: ECCV 2010 - 11th European Conference on Computer Vision (2010)

[22] Campbell, N.D.F., Vogiatzis, G., Hernández, C., Cipolla, R.: Using multiple hypotheses to improve depth-maps for multi-view stereo. In: ECCV (2008)

[23] Chen, J., Bautembach, D., Izadi, S.: Scalable real-time volumetric surface reconstruction. ACM Trans. Graph. (2013)

[24] Cheung, G.K.M., Baker, S., Kanade, T.: Visual hull alignment and refinement across time: A 3d reconstruction algorithm combining shape-from-silhouette with stereo. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), 16-22 June 2003, Madison, WI, USA (2003)

[25] Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: ECCV (2016)

[26] Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A.G., Sullivan, S.: High-quality streamable free-viewpoint video. ACM Trans. Graph. (2015)

[27] Collins, R.T.: A space-sweep approach to true multi-image matching. In: CVPR (1996)

[28] Cremers, D., Kolev, K.: Multiview stereo and silhouette consistency via convex functionals over convex domains. IEEE Trans. Pattern Anal. Mach. Intell. (2011)

[29] Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: SIGGRAPH (1996)

[30] Delaunoy, A., Prados, E., Gargallo, P., Pons, J., Sturm, P.F.: Minimizing the multi-view stereo reprojection error for triangular surface meshes. In: Proceedings of the British Machine Vision Conference 2008, Leeds, UK, September 2008 (2008)

[31] Dou, M., Davidson, P.L., Fanello, S.R., Khamis, S., Kowdle, A., Rhemann, C., Tankovich, V., Izadi, S.: Motion2fusion: real-time volumetric performance capture. ACM Trans. Graph. (2017)

[32] Dou, M., Khamis, S., Degtyarev, Y., Davidson, P., Fanello, S.R., Kowdle, A., Escolano, S.O., Rhemann, C., Kim, D., Taylor, J., Kohli, P., Tankovich, V., Izadi, S.: Fusion4d: Real-time performance capture of challenging scenes. ACM Trans. Graph. (2016)

[33] Douze, M., Franco, J.S., Raffin, B.: QuickCSG: Arbitrary and Faster Boolean Combinations of N Solids. Research report (2015)

[34] Durou, J.D., Charvillat, V., Daramy, M., Gurdjos, P.: Résolution du shape-from-shading par apprentissage. In: ORASIS - Congrès des jeunes chercheurs en vision par ordinateur (2011)

[35] Esteban, C.H., Schmitt, F.: Silhouette and stereo fusion for 3d object modeling. Computer Vision and Image Understanding (2004)

[36] Faugeras, O.D., Keriven, R.: Complete dense stereovision using level set methods. In: ECCV'98 (1998)

[37] Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: Learning to predict new views from the world's imagery. In: CVPR (2016)

[38] Franco, J., Boyer, E.: Efficient polyhedral modeling from silhouettes. IEEE Trans. Pattern Anal. Mach. Intell. (2009)

[39] Fua, P.: From multiple stereo views to multiple 3-d surfaces. International Journal of Computer Vision (1997)

[40] Fua, P., Leclerc, Y.G.: Object-centered surface reconstruction: Combining multi-image stereo and shading. International Journal of Computer Vision (1995)

[41] Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi-view stereopsis. In: CVPR (2007)

[42] Gall, J., Stoll, C., Aguiar, E.D., Theobalt, C., Rosenhahn, B., peter Seidel, H.: Motion capture using joint skeleton tracking and surface estimation. In: CVPR (2009)

[43] Galliani, S., Lasinger, K., Schindler, K.: Massively parallel multiview stereopsis by surface normal diffusion. In: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. pp. 873–881 (2015)

[44] Galliani, S., Schindler, K.: Just look at the image: Viewpoint-specific surface normal prediction for improved multi-view reconstruction. In: CVPR (2016)

[45] Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2012)

[46] Gilbert, A., Volino, M., Collomosse, J., Hilton, A.: Volumetric performance capture from minimal camera viewpoints. In: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI (2018)

[47] Goldlücke, B., Magnor, M.A.: Space-time isosurface evolution for temporally coherent 3d reconstruction. In: CVPR (2004)

[48] Guillemaut, J.Y., Kilner, J., Hilton, A.: Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes. In: ICCV (2009)

[49] Guo, K., Xu, F., Wang, Y., Liu, Y., Dai, Q.: Robust non-rigid motion tracking and surface reconstruction using l_0 regularization. IEEE Trans. Vis. Comput. Graph. **24**(5), 1770–1783 (2018)

[50] Habermann, M., Xu, W., Zollhöfer, M., Pons-Moll, G., Theobalt, C.: Reticam: Real-time human performance capture from monocular video (2018)

[51] Hartmann, W., Galliani, S., Havlena, M., Van Gool, L., Schindler, K.: Learned multi-patch similarity. In: ICCV (2017)

[52] Hedman, P., Philip, J., Price, T., Frahm, J.M., Drettakis, G., Brostow, G.: Deep blending for free-viewpoint image-based rendering. ACM Transactions on Graphics (SIGGRAPH Asia Conference Proceedings) **37**(6) (November 2018)

[53] Hirschmüller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. IEEE Trans. Pattern Anal. Mach. Intell. (2009)

[54] Horn, B.K.P.: Understanding image intensities. Artif. Intell. (1977)

[55] Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision. IEEE Trans. Pattern Anal. Mach. Intell. (2012)

[56] Hu, Y., Zhou, Q., Gao, X., Jacobson, A., Zorin, D., Panozzo, D.: Tetrahedral meshing in the wild. ACM Trans. Graph. **37**(4), 60:1–60:14 (2018)

[57] Huang, P., Matzen, K., Kopf, J., Ahuja, N., Huang, J.: Deepmvs: Learning multi-view stereopsis. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 2821–2830 (2018)

[58] Huang, Q., Wang, H., Koltun, V.: Single-view reconstruction via joint analysis of image and shape collections. ACM Trans. Graph. (2015)

[59] Huang, Z., Li, T., Chen, W., Zhao, Y., Xing, J., LeGendre, C., Luo, L., Ma, C., Li, H.: Deep volumetric video from very sparse multi-view performance capture. In: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XVI (2018)

[60] Innmann, M., Zollhöfer, M., Nießner, M., Theobalt, C., Stamminger, M.: Volumedeform: Real-time volumetric non-rigid reconstruction. In: ECCV (2016)

[61] Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R.A., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A.J., Fitzgibbon, A.W.: Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (2011)

[62] Jensen, R.R., Dahl, A.L., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: CVPR (2014)

[63] Ji, M., Gall, J., Zheng, H., Liu, Y., Fang, L.: Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In: ICCV (2017)

[64] Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. In: NIPS (2017)

[65] Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A.: End-to-end learning of geometry and context for deep stereo regression. In: ICCV (2017)

[66] Khanian, M., Boroujerdi, A.S., Breuß, M.: Photometric stereo for strong specular highlights. Computational Visual Media (2018)

[67] Klette, R., Krüger, N., Vaudrey, T., Pauwels, K., Hulle, M.M.V., Morales, S., Kandil, F.I., Haeusler, R., Pugeault, N., Rabe, C., Lappe, M.: Performance of correspondence algorithms in vision-based driver assistance using an online image sequence database. IEEE Trans. Vehicular Technology (2011)

[68] Knapitsch, A., Park, J., Zhou, Q., Koltun, V.: Tanks and temples: benchmarking large-scale scene reconstruction. ACM Trans. Graph. (2017)

[69] Kolmogorov, V., Zabih, R.: Multi-camera scene reconstruction via graph cuts. In: Computer Vision - ECCV 2002, 7th European Conference on Computer Vision, Copenhagen, Denmark, May 28-31, 2002, Proceedings, Part III. pp. 82–96 (2002)

[70] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Conference on Neural Information Processing Systems (NIPS) (2012)

[71] Kutulakos, K.N., Seitz, S.M.: A theory of shape by space carving (2000)

[72] Labatut, P., Pons, J., Keriven, R.: Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In: ICCV (2007)

[73] Larsen, E.S., Mordohai, P., Pollefeys, M., Fuchs, H.: Temporally consistent reconstruction from multiple video streams using enhanced belief propagation. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007 (2007)

[74] Leroy, V., Franco, J.S., Boyer, E.: Multi-View Dynamic Shape Refinement Using Local Temporal Integration. In: ICCV (2017)

[75] Leroy, V., Franco, J., Boyer, E.: Shape reconstruction using volume sweeping and learned photoconsistency. In: Computer Vision - ECCV (2018)

[76] Liao, Y., Donné, S., Geiger, A.: Deep marching cubes: Learning explicit surface representations. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 2916–2925 (2018)

[77] Liu, Y., Dai, Q., Xu, W.: A point-cloud-based multiview stereo algorithm for free-viewpoint video. IEEE Trans. Vis. Comput. Graph. (2010)

[78] Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. In: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1987, Anaheim, California, USA, July 27-31, 1987 (1987)

[79] Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV (1999)

[80] Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)

[81] Luo, W., Schwing, A.G., Urtasun, R.: Efficient deep learning for stereo matching. In: CVPR (2016)

[82] Marr, D., Poggio, T.: A theory of human stereo vision. In: M.I.T. A.I. Lab. Memo 451 (1977)

[83] Marr, D., Poggio, T.: A computational theory of human stereo vision. In: Proceedings of the Royal Society of London (1979)

[84] Martin-Brualla, R., Pandey, R., Yang, S., Pidlypenskyi, P., Taylor, J., Valentin, J.P.C., Khamis, S., Davidson, P.L., Tkach, A., Lincoln, P., Kowdle, A., Rhemann, C., Goldman, D.B., Keskin, C., Seitz, S.M., Izadi, S., Fanello, S.R.: Lookingood: Enhancing performance capture with real-time neural re-rendering. CoRR (2018)

[85] Merrell, P., Akbarzadeh, A., Wang, L., michael Frahm, J., Nistr, R.Y.D.: Real-time visibility-based fusion of depth maps. In: CVPR (2007)

[86] Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. In: CVPR (2003)

[87] Mustafa, A., Kim, H., Guillemaut, J., Hilton, A.: Temporally coherent 4d reconstruction of complex dynamic scenes. In: CVPR (2016)

[88] Narayanan, P.J., Rander, P., Kanade, T.: Constructing virtual worlds using dense stereo. In: ICCV (1998)

[89] Neumann, J., Aloimonos, Y.: Spatio-temporal stereo using multi-resolution subdivision surfaces. International Journal of Computer Vision (2002)

[90] Newcombe, R.A., Fox, D., Seitz, S.M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: CVPR (2015)

[91] Oswald, M.R., Cremers, D.: A convex relaxation approach to space time multi-view 3d reconstruction. In: ICCV Workshop on Dynamic Shape Capture and Analysis (4DMOD) (2013)

[92] Oswald, M.R., Stühmer, J., Cremers, D.: Generalized connectivity constraints for spatio-temporal 3d reconstruction. In: European Conference on Computer Vision (ECCV). pp. 32–46 (2014)

[93] Oswald, M.R., Cremers, D.: Surface normal integration for convex space-time multi-view reconstruction. In: British Machine Vision Conference, BMVC 2014, Nottingham, UK, September 1-5, 2014 (2014)

[94] Petit, B., Dupeux, T., Bossavit, B., Legaux, J., Raffin, B., Melin, E., Franco, J., Assenmacher, I., Boyer, E.: A 3d data intensive tele-immersive grid. In: Proceedings of the 18th International Conference on Multimedia 2010 (2010)

[95] Petit, B., Lesage, J., Ménier, C., Allard, J., Franco, J., Raffin, B., Boyer, E., Faure, F.: Multicamera real-time 3d modeling for telepresence and remote collaboration. Int. J. Digital Multimedia Broadcasting (2010)

[96] Pfeiffer, D., Gehrig, S., Schneider, N.: Exploiting the power of stereo confidences. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013 (2013)

[97] Pons, J.P., Keriven, R., Faugeras, O.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. IJCV (2007)

[98] Pribanic, T., Mrvos, S., Salvi, J.: Efficient multiple phase shift patterns for dense 3d acquisition in structured light scanning. Image Vision Comput. (2010)

[99] Saxena, A., Schulte, J., Ng, A.Y.: Depth estimation using monocular and stereo cues. In: IJCAI 2007, International Joint Conference on Artificial Intelligence (2007)

[100] Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision (1-3) (2002)

[101] Schlüns, K.: Photometric stereo for non-lambertian surfaces using color information. In: Computer Analysis of Images and Patterns, 5th International Conference, CAIP'93, Budapest, Hungary, September 13-15, 1993, Proceedings (1993)

[102] Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: CVPR (2017)

[103] Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: CVPR 2017, Honolulu,USA (2017)

[104] Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: CVPR (2006)

[105] Seitz, S.M., Dyer, C.R.: Photorealistic scene reconstruction by voxel coloring. In: Conference on Computer Vision and Pattern Recognition (1997)

[106] Shi, B., Wu, Z., Mo, Z., Duan, D., Yeung, S., Tan, P.: A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016 (2016)

[107] Sinha, S.N., Mordohai, P., Pollefeys, M.: Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007. pp. 1–8 (2007)

[108] Sipiran, I., Bustos, B.: A robust 3d interest points detector based on harris operator. In: Eurographics Workshop on 3D Object Retrieval, Norrköping, Sweden, May 2, 2010, Proceedings (2010)

[109] Starck, J., Hilton, A.: Surface capture for performance-based animation. IEEE Comput. Graph. Appl. (2007)

[110] Strecha, C., von Hansen, W., Gool, L.V., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. CVPR (2008)

[111] Sun, J., Ovsjanikov, M., Guibas, L.J.: A concise and provably informative multi-scale signature based on heat diffusion. Comput. Graph. Forum (2009)

[112] Tewari, A., Zollhöfer, M., Garrido, P., Bernard, F., Kim, H., Pérez, P., Theobalt, C.: Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. CVPR (2018)

[113] Thiery, J., Tierny, J., Boubekeur, T.: Cager: from 3d performance capture to cage-based representation. In: International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2012, Los Angeles, CA, USA, August 5-9, 2012, Talks Proceedings. p. 16 (2012)

[114] Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: CVPR (2008)

[115] Tola, E., Lepetit, V., Fua, P.: DAISY: an efficient dense descriptor applied to wide-baseline stereo. IEEE Trans. Pattern Anal. Mach. Intell. (2010)

[116] Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Mach. Vis. Appl. (2012)

[117] Treece, G.M., Prager, R.W., Gee, A.H.: Regularised marching tetrahedra: improved iso-surface extraction. Computers & Graphics **23**(4), 583–598 (1999)

[118] Tung, T., Nobuhara, S., Matsuyama, T.: Complete multi-view reconstruction of dynamic scenes from probabilistic fusion of narrow and wide baseline stereo. In: ICCV (2009)

[119] Ulusoy, A.O., Geiger, A., Black, M.J.: Towards probabilistic volumetric reconstruction using ray potentials. In: 3DV (2015)

[120] Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: CVPR (2017)

[121] Valkenburg, R.J., McIvor, A.M.: Accurate 3d measurement using a structured light system. Image Vision Comput. (1998)

[122] Vedula, S., Baker, S., Rander, P., Collins, R.T., Kanade, T.: Three-dimensional scene flow. In: ICCV (1999)

[123] Vedula, S., Baker, S., Seitz, S.M., Kanade, T.: Shape and motion carving in 6d. In: Conference on Computer Vision and Pattern Recognition (CVPR 2000), 13-15 June 2000, Hilton Head, SC, USA (2000)

[124] Vlasic, D., Baran, I., Matusik, W., Popovic, J.: Articulated mesh animation from multi-view silhouettes. ACM Trans. Graph. (2008)

[125] Vlasic, D., Peers, P., Baran, I., Debevec, P.E., Popovic, J., Rusinkiewicz, S., Matusik, W.: Dynamic shape capture using multi-view photometric stereo. ACM Trans. Graph. (2009)

[126] Vogiatzis, G., Torr, P.H.S., Cipolla, R.: Multi-view stereo via volumetric graph-cuts. In: CVPR (2005)

[127] Vogiatzis, G., Esteban, C.H., Torr, P.H.S., Cipolla, R.: Multi-view stereo via volumetric graph-cuts and occlusion robust photo-consistency. IEEE Trans. Pattern Anal. Mach. Intell. (2007)

[128] Vollmer, J., Mencl, R., Müller, H.: Improved laplacian smoothing of noisy surface meshes. Comput. Graph. Forum **18**(3), 131–138 (1999)

[129] Vu, H., Keriven, R., Labatut, P., Pons, J.: Towards high-resolution large-scale multi-view stereo. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA (2009)

[130] Žbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. (2016)

[131] Wang, L., Hétroy-Wheeler, F., Boyer, E.: On volumetric shape reconstruction from implicit forms. In: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III (2016)

[132] Waschbüsch, M., Würmlin, S., Cotting, D., Sadlo, F., Gross, M.H.: Scalable 3d video of dynamic scenes. The Visual Computer **21**(8-10), 629–638 (2005)

[133] Wu, C., Frahm, J., Pollefeys, M.: Repetition-based dense single-view reconstruction. In: CVPR (2011)

[134] Xu, W., Chatterjee, A., Zollhöfer, M., Rhodin, H., Mehta, D., Seidel, H., Theobalt, C.: Monoperfcap: Human performance capture from monocular video. ACM Trans. Graph. **37**(2), 27:1–27:15 (2018)

[135] Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. ECCV 2018 Munich (2018)

[136] Zach, C., Pock, T., Bischof, H.: A globally optimal algorithm for robust tv-l^1 range image integration. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007 (2007)

[137] Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: CVPR (2015)

[138] Zaharescu, A., Boyer, E., Horaud, R.: Keypoints and local descriptors of scalar functions on 2d manifolds. International Journal of Computer Vision (2012)

[139] Zeng, G., Paris, S., Quan, L.: Robust carving for non-lambertian objects. In: 17th International Conference on Pattern Recognition, ICPR. (2004)

[140] Zeng, M., Zheng, J., Cheng, X., Liu, X.: Templateless quasi-rigid shape modeling with implicit loop-closure. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013. pp. 145–152 (2013)