Saarland University
Faculty of Mathematics and Computer Science
Department of Computer Science

Master's Thesis

ϕ-SfT: Shape-from-Template with a Physics-Based Deformation Model

Submitted by

Navami Kairanda

Submitted on

February 27, 2022

Reviewers

Prof. Christian Theobalt

Dr. Vladislav Golyanik

Dean:

Univ.-Prof. Dr. Thomas Schuster
Saarland University,
Saarbrücken, Germany

Supervisors:

Edith Tretschk

Dr. Mohamed Elgharib

Dr. Vladislav Golyanik

Prof. Christian Theobalt

Erklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, 27.02.2022 

(Datum/Date) (Unterschrift / Signature)

Erklärung

Ich erkläre hiermit, dass die vorliegende Arbeit mit der elektronischen Version übereinstimmt.

Statement

I hereby confirm the congruence of the contents of the printed data and the electronic version of the thesis.

Saarbrücken, 27.02.2022 

(Datum/Date) (Unterschrift / Signature)

To Jnanachandra Kairanda

Abstract

Reconstructing the 3D shape of objects from multiple images is a fundamental problem in computer vision. It has been extensively studied for both rigid and deformable objects. While there are accurate and stable solutions such as Structure-from-Motion (SfM) for reconstructing rigid objects, the deformable case remains an open research problem. Shape-from-Template (SfT) and Non-Rigid Structure-from-Motion (NRSfM) and the more recent neural 3D mesh regression methods address this problem by taking into account the effect of deformation on the geometry of the object.

In particular, SfT methods estimate the deformations of an examined surface from a single RGB camera while assuming one of its 3D states (a template) is known in advance. This is an important yet challenging problem due to the under-constrained nature of monocular 3D reconstruction. Existing SfT approaches use approximate deformation models rather than more sophisticated models of physical deformation behavior; this limits their reconstruction abilities.

This work proposes a new SfT approach explaining the observations through simulation of a physically-based surface deformation model representing forces and material properties. In contrast to previous works, we utilise a physics-based simulator to implicitly regularise the surface evolution. This has been made possible with the advance of differentiable physics simulators (for *e.g.,* Liang *et al.* (2019)) that enable gradient-based optimisation for inverse problems. In addition to geometry, we estimate the material properties of the deformable surface such as its bending coefficients, elasticity, stiffness, and material density. We use a differentiable renderer to minimise the dense reprojection error between the estimated 3D states and the input images, and recover the deformation parameters using an adaptive gradient-based optimisation. For the evaluation, we record with an RGB-D camera challenging real surfaces with various material properties and texture, exposed to physical forces. In addition, we generate a new synthetic dataset of naturalistically deforming surfaces using physics-based simulation. On both datasets, our approach reconstructs the underlying deformations much more accurately than related state-of-the-art methods. As our reconstruction method estimates the material properties and forces that generate the deformations, we show applications for intuitively controlling deformations by editing underlying surface material and acting forces.

Acknowledgements

I would like to express my sincere gratitude to all my supervisors for guiding me through this project with absolute passion. Specifically, I wish to thank my thesis reviewers Dr. Vladislav Golyanik and Prof. Dr. Christian Theobalt for providing me the opportunity, guiding with profound insights and most importantly believing in me and my abilities. I wish to thank Dr. Mohamed Elgharib for always being supportive, insightful and additionally guiding me in capturing the dataset. I am very grateful to Edith Tretschk for the close supervision, mentoring and showing me the way of meticulous research.

I would like to thank all my fellow colleagues at MPI Informatics, particularly from the Visual Computing and Artificial Intelligence department, with whom I could have valuable discussions about the project and the field. I would like to further thank the professors and tutors at Saarland University, for helping me to build the basic knowledge of Computer Vision and Graphics which motivated and empowered me to take up this challenging project.

Finally, I am grateful to Ayush, my parents, brother, family, and friends for encouraging, supporting and for always being there by my side.

Contents

List of Figures

List of Tables

Acronyms

DR Differentiable Rendering.

ICP Iterative Closest Point.

NRSfM Non-Rigid Structure from Motion.

SfM Structure from Motion.

SfT Shape from Template.

Acronyms

Chapter 1

Introduction

1.1 Motivation

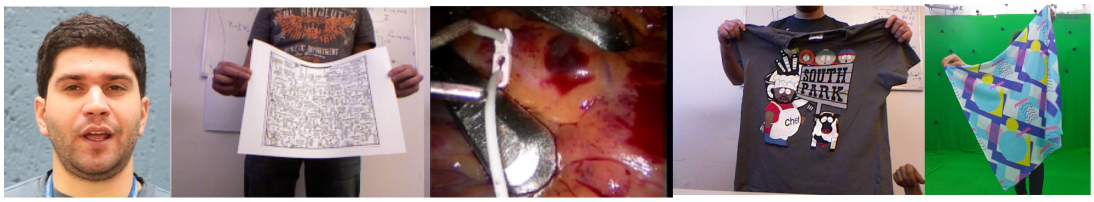

Figure 1.1: We show several examples of non-rigid object sequences addressed in literature, for *e.g., face* (Garg *et al.*, 2013a), *paper* (Varol *et al.*, 2012), *heart* (Stoyanov, 2012) and *cloth simple* (Varol *et al.*, 2012). Cloths belong to the most challenging category due to higher degrees of freedom for movement. However, none of the existing methods reconstruct cloth sequences that include local folds. *cloth challenging* shows a frame from one of our newly captured sequences.

Reconstructing general deformable, temporally-coherent surfaces in 3D from monocular videos is a long-standing challenging and ill-posed problem. It was studied under different assumptions, and methods addressing it can be roughly classified into (template-free) *non-rigid structure from motion* (NRSfM) (Bregler *et al.*, 2000; Garg *et al.*, 2013a), (template-based) *shape-from-template* (SfT) (Ngo *et al.*, 2015; Perriollat *et al.*, 2011), and neural 3D mesh regression (Li *et al.*, 2020). The objective of SfT is: Given a known initial 3D state of an observed deformable scene or an object, reconstruct all its later states observed in a monocular image sequence (Salzmann *et al.*, 2007). Recent SfT methods are learning-based (Fuentes-Jimenez *et al.*, 2021; Shimada *et al.*, 2019), *i.e.,* they encode prior knowledge about the deformation models and templates in neural network weights. This offers multiple advantages over a vast body of previous, non-learning-based works (Ngo *et al.*, 2015; Östlund *et al.*, 2012; Parashar *et al.*,

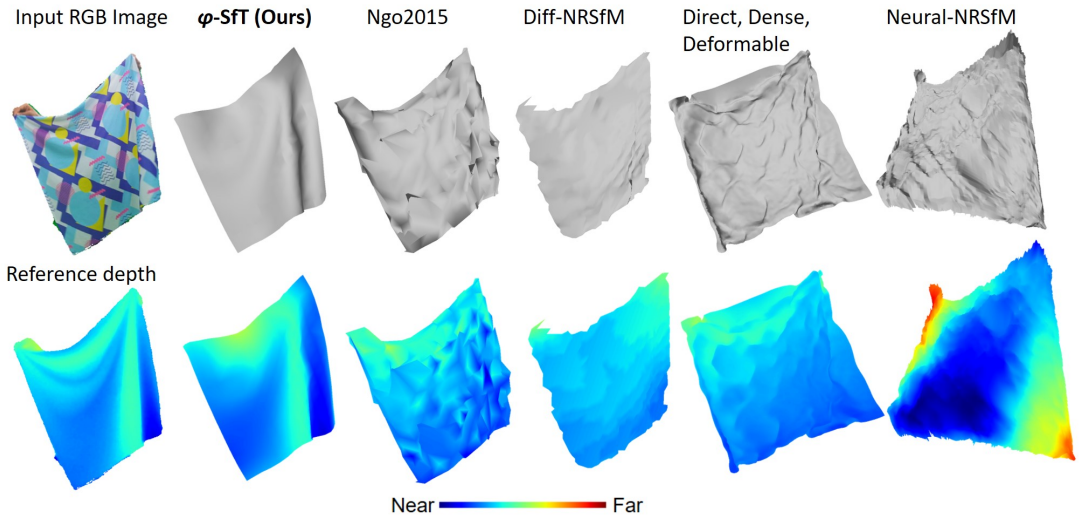

Figure 1.2: Our ϕ-SfT approach uses a physics simulator to reconstruct challenging deforming 3D surfaces observed in a monocular video. In contrast to existing methods, our estimates are significantly more accurate and evince increased physical plausibility.

2015; Perriollat *et al.*, 2011; Salzmann *et al.*, 2007; Salzmann *et al.*, 2009; Yu *et al.*, 2015), such as the ability to handle larger deformations, a broader spectrum of supported types of motions and deformations (including highly non-linear ones), and real-time operation.

Fig. 1.1 shows different types of non-rigid objects including the most challenging category of deforming clothes. Cloths deform freely and often form fine local surface deformations such as folds and wrinkles. One of the pivotal limitations of both classical and neural SfT methods is that they capture general 3D states well but not fine local surface deformations. This is due to non-awareness of the physical fold formation process attributable to the elastic properties of the materials and forces acting on them. As a result, existing methods can only reconstruct predominantly global deformations.

1.2 ϕ-SfT

This paper proposes ϕ-SfT (from Greek $\phi\upsilon\sigma\iota\kappa\eta$ meaning *physics*): A new analysis-by-synthesis SfT method which addresses several limitations of the current state of the art and improves the accuracy of monocular non-rigid 3D reconstruction by a significant margin, see Fig. 1.2 for comparison. Our approach explicitly models the physical fold formation process, and its parameters are physically meaningful. ϕ-SfT is a non-learning-based analysis-by-synthesis approach that

does not require training data. We enable gradient-based optimisation by employing two components: a differentiable renderer and a differentiable physics simulator. Our core idea is to use the latter as a regulariser during the optimisation of our objective function. Furthermore, the differentiable renderer ensures that the reprojections of the recovered 3D states accurately match the observed images. In contrast to earlier photometric terms used for SfT (Yu *et al.*, 2015), using differentiable rendering allows us for the first time to define the reprojection error densely *per pixel* and not only *per vertex*. We can thus exploit the information present in the texture regardless of the mesh resolution. Our approach is significantly more accurate than related methods and supports finer-scale local folds, which is demonstrated on a wider spectrum of deformations in extensive experiments (Sec. 6).

1.3 Scope

The scope of this thesis is restricted to reconstruction and tracking of deformable 3D surfaces from monocular RGB sequence and 3D template. We show that our method, ϕ-SfT can accurately estimate the geometry including fine folds and it outperforms related methods on the newly collected real and synthetic datasets.

Since we optimise for geometry in the *classical SfT* setting, note that accurate estimation of material elasticity, or forces, etc., are out of scope for this project.

1.4 Outline

In the subsequent chapters, we first introduce related works in Chapter 2, followed by a description of pre-requisites that form the main ingredients for our method in Chapter 3. Next, we introduce the ϕ-SfT real and synthetic datasets in Chapter 4.

Then, we present the ϕ-SfT deformation model and the optimisation techniques in Chapter 5, followed by evaluations of the model in Chapter 6, and applications in Chapter 7. Finally, we present the limitations of our model and discuss possible future work in Chapter 8, and conclude in Chapter 9.

1.4 Outline

Chapter 2

Related Work

Related work on dense monocular non-rigid 3D reconstruction is vast. The methods in the literature differ in the assumptions they make about the input, available prior knowledge and how they model the deformations. This chapter reviews methods that can be classified into non-rigid structure from motion (NRSfM), shape from template (SfT), monocular 3D mesh reconstruction and monocular volume reconstruction. In Tab. 2.1, we provide a short summary of the related methods that are later used for comparison with ϕ-SfT.

2.1 NRSfM

NRSfM operates on point tracks over the input monocular views. Earlier NRSfM methods were designed for sparse 2D point tracks and modelled deformations with linear subspaces along with various priors (Akhter *et al.*, 2009; Bregler *et al.*, 2000; Dai *et al.*, 2014; Paladini *et al.*, 2009; Torresani *et al.*, 2008). More recent NRSfM techniques (Agudo & Moreno-Noguer, 2018; Garg *et al.*, 2013a; Kumar *et al.*, 2018) allow reconstructing dense image points observed in a reference frame. They impose constraints on spatial point locations to infer smooth and continuous deforming surfaces. Agudo & Moreno-Noguer (2015) propose a sparse

Method	Template	Point tracks	Supervision	Deformation constraint
DDD Yu *et al.* (2015)	✓	✗	per vertex	isometry
Ngo2015 Ngo *et al.* (2015)	✓	✗	per feature	isometry
IsMo-GAN Shimada *et al.* (2019)	✓	✗	pre trained	isometry
N-NRSfM Sidhu *et al.* (2020)	✗	✓	per pixel	subspace constraint on point tracks
Diff-NRSfM Parashar *et al.* (2020)	✗	✓	per feature/pixel	differential structure preservation
ϕ-SfT (ours)	✓	✗	per pixel	anisotropic elastic model

Table 2.1: Comparison of our approach with other monocular non-rigid 3D reconstruction methods. The methods in the literature differ in the assumptions they make about the input, available prior knowledge (whether 2D point tracks or 3D template is required), how they model the deformations and the source of supervision signal. Note that all the NRSfM methods here are dense.

NRSfM method relying on principles of continuum mechanics, *i.e.,* it represents a deformable object using an estimated (not a simulated) elastic model and a low-rank force field acting on it. Even though the force prior has a direct physical interpretation, this model still shares most limitations with other NRSfM methods. Diff-NRSfM (Parashar *et al.*, 2020) assumes the observed structure preserves its differentiable structure and infinitesimal planarity. This method produces impressive results for smooth surfaces but struggles to reconstruct fine-scale folds, unlike our ϕ-SfT. Recently, neural NRSfM approaches both for sparse (Novotny *et al.*, 2019; Wang & Lucey, 2021) and dense (Sahasrabudhe *et al.*, 2019; Sidhu *et al.*, 2020) cases were proposed in the literature. Some of them need to be trained for each object category (Novotny *et al.*, 2019; Sahasrabudhe *et al.*, 2019), whereas N-NRSfM (Sidhu *et al.*, 2020) and PAUL (Wang & Lucey, 2021) run on unknown data. Some 2D keypoint lifting approaches for 3D human pose estimation, such as Chen *et al.* (2019), require only 2D data for supervision and share similarities with neural sparse NRSfM.

2.2 SfT

SfT algorithms operate directly on images and assume a known 3D surface prior as input (Ngo *et al.*, 2015; Perriollat *et al.*, 2011; Salzmann *et al.*, 2007; Yu *et al.*, 2015). These methods minimise the 3D-2D reprojection error and impose geometric constraints such as surface inextensibility (Perriollat *et al.*, 2011; Salzmann *et al.*, 2007) or isometry (Bartoli *et al.*, 2015; Ngo *et al.*, 2015; Yu *et al.*, 2015). Recent neural SfT methods (Fuentes-Jimenez *et al.*, 2021; Pumarola *et al.*, 2018; Shimada *et al.*, 2019) predict 3D surfaces from monocular images relying on datasets with different template states. Our approach contrasts with other SfT methods in that it uses temporal information and a differentiable physics simulator as a regulariser for high-fidelity 3D surface tracking instead of approximating the underlying physical properties via geometric constraints. Moreover, none of these methods uses a *per pixel* differentiable photometric loss which ensures that 3D estimates accurately reproject into the 2D images.

2.3 Physics-Based Priors

Physics-based priors in 3D human performance capture is an emerging field, although there is some early work on it (Stoll *et al.*, 2010). Rempe *et al.* (2020) method and PhysCap (Shimada *et al.*, 2020) show that integrating physics laws into an objective for sparse 3D human motion capture improves the accuracy and quality of the 3D estimates. The proposed constraints reduce the artifacts arising from the monocular setting, such as unnatural jitter of the recovered structure,

unnatural body leaning, foot sliding, and foot-floor penetration. Several methods for 3D human performance capture include *clothes deformations*, such as Guo *et al.* (2021) and Li *et al.* (2021). The method of Guo *et al.* operates on point clouds and optimises the states of the simulated clothes so that they match the inputs. The cloth motion is expressed through a combination of skin friction, gravity and forces attributed to the material (elasticity). Thus, their focus is cloth state recovery from sparse point cloud measurements, which provide a strong 3D shape cue, whereas we assume a single provided 3D template and operate on monocular videos; this is a much more ill-posed inverse problem. Li *et al.* generate training data with a physics-based simulator on-the-fly and use it to train a neural network for 3D human performance capture, including clothes deformations. Thus, they do not impose hard physics-based constraints as we do with the differentiable physics simulator. Work by Liang *et al.* (2019) uses 3D supervision for physics-based cloth reconstruction. The work by Weiss *et al.* (2020) recovers material parameters of a physics simulator in an analysis-by-synthesis policy to solve an inverse elasticity problem. In contrast to our method, they additionally require depth inputs for a strong 3D cue, and they do not recover local surface deformations. The broad idea of using a combination of a differentiable physics engine and a differentiable graphics engine has previously been explored in the works of (Jaques *et al.*, 2020; Kandukuri *et al.*, 2020; Murthy *et al.*, 2021). However, these methods estimate physical parameters in controlled setting and for rigid bodies whereas we aim to accurately reconstruct the surface deformations of more challenging and high-dimensional (DoF) surfaces.

2.4 Monocular 3D Mesh Reconstruction

Monocular 3D mesh reconstruction approaches can be trained on extensive collections of unstructured views in the desired object category. Some works (Choy *et al.*, 2016; Wang *et al.*, 2018) require 3D supervision, while others, similar to ours, do not: In an early work, Cashman & Fitzgibbon (2013) show that sufficiently rigid object categories, like dolphins, can be reconstructed from image collections. Kanazawa *et al.* (2018) relax the need for input annotations. Li *et al.* (2020) extend Kanazawa *et al.* (2018) to video input and estimate a temporally consistent coarse mesh reconstruction for weakly articulated objects. LASR (Wu *et al.*, 2021; Yang *et al.*, 2021a) further relax the need for an initial coarse template. ViSER (Yang *et al.*, 2021b) extend LASR to reason about long-range correspondences and is robust to moderate shape variations and appearance changes. We differ from these by the usage of a physics-based deformation model, and we focus on recovering local surface deformations.

2.5 Monocular Volume Reconstruction

Monocular Volume Reconstruction methods learn a continuous scene function for novel view synthesis given a set of monocular images. Earlier works (Lombardi *et al.*, 2019; Mildenhall *et al.*, 2020; Sitzmann *et al.*, 2019) assume the scene is rigid and the camera poses are accurately registered. To extend them to dynamic scenes, recent works (Park *et al.*, 2021; Tretschk *et al.*, 2021) introduce additional functions to deform observed points to a canonical space over time. Yang *et al.* (2021c) support deformable scenes when the motion between objects and background is large and reconstruct animatable 3D models.

Chapter 3

Preliminaries

In this chapter, we introduce the important concepts needed to understand our method described in Chapter 5.

3.1 Material Elastic Model

(a) Gray Interlock (b) Pink Ribbon Brown (c) Navy Sparkle Sweat (d) 11oz Black Denim (e) White Dots on Black

Figure 3.1: An elastic model defines the strain/displacement generated due to stress/force for a given surface material. Surfaces with varying underlying material (shown with different textures) deform differently when subjected to the same external forces (contact forces with mannequin and gravity). Image credit Wang *et al.* (2011)

ϕ-SfT models the deformation field as a function of forces acting on the given surface as well underlying elastic properties of the surface material. In this section, we provide a brief introduction to material elasticity. An elastic model defines the strain/displacement generated due to stress/force for a given surface material. As shown in Fig. 3.1, surfaces with varying underlying elasticity deform differently when subjected to identical external forces. Given ϕ-SfT's aim to recover time-varying surface deformations from monocular sequence, modelling the elasticity enables to recover the distinctive fold and wrinkle patterns for a range of different materials. In ϕ-SfT, we use the the elasticity measurements of *The Data-Driven Elastic Model* (Wang *et al.*, 2011) for describing the material properties.

The Data-Driven Elastic Model by Wang *et al.* (2011) is a piecewise linear elastic model that provides a good approximation to nonlinear, anisotropic stretching and bending behaviors of various materials. This material model consists of three parts: density d, stretching stiffness $\mathcal{S}$, and bending stiffness $\mathcal{B}$. The stretching stiffness quantifies how large the reaction force will be when the cloth is stretched out. The bending stiffness models how easily the cloth can be bent and folded. Wang *et al.* (2011) record a real-world dataset consisting of 10 different cloth materials, a few of them are visualised in Fig. 3.1. We use these measurements in the physics simulator (see Sec. 3.2) to create natural and realistic clothing folds and shapes, for a range of different materials. Specifically, we initialise the material parameters in our optimisation to the average of these ten measurements.

3.2 Physics Simulator

At the heart of ϕ-SfT deformation model lies differentiable physics simulation. In this section, we provide background on physics simulation for clothes.

Suppose, we have a surface parameterised as a triangular mesh $\mathbf{S}_t = \{\mathbf{V}_t, \mathbf{E}\}$ where the state of the i-th vertex in $\mathbf{V}_t$ comprises its 3D position $\mathbf{x}_t^i \in \mathbb{R}^3$ and its velocity $\mathbf{v}_t^i \in \mathbb{R}^3$. In the continuous domain, physics-based simulation can be formulated as a time-varying partial differential equation (Baraff & Witkin, 1998):

$$\frac{\partial^2 \mathbf{x}}{\partial t} = \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}, \mathbf{v}), \tag{3.1}$$

where $(\mathbf{x}, \mathbf{v})$ is the vertex state, and $\mathbf{M}$ is a diagonal matrix of the mass distribution derived from the material density d and surface area. $\mathbf{f}(\cdot)$ are the forces, *i.e.,* internal forces which are a function of cloth elastic properties $\mathcal{S}$ and $\mathcal{B}$ as well as external forces such as wind or gravity. We follow the elastic model of cloth materials by Wang *et al.* (2011) for describing the effects of d, $\mathcal{S}$ and $\mathcal{B}$ (see Sec. 3.1).

In practice, we are given the known position $\mathbf{x}_{t-1}$ and velocity $\mathbf{v}_{t-1}$ of the system at time $t-1$. Our goal is to determine the new position $\mathbf{x}_t = \mathbf{x}_{t-1} + \Delta\mathbf{x}$ and velocity $\mathbf{v}_t = \mathbf{v}_{t-1} + \Delta\mathbf{v}$ at time t with a time step size h=1. To that end, Eq. (3.1) can be transformed into a first-order differential equation, and can then be solved for $\Delta\mathbf{x}$ and $\Delta\mathbf{v}$ with the implicit, backward Euler method (Baraff & Witkin, 1998):

$$\begin{pmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{v} \end{pmatrix} = h \begin{pmatrix} \mathbf{v}_t \\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}_t, \mathbf{v}_t) \end{pmatrix}, \tag{3.2}$$

which is non-linear due to $\mathbf{f}$. For efficiently solving Eq. (3.2), $\mathbf{f}$ can be linearised via first-order Taylor series approximation:

$$\mathbf{f}(\mathbf{x}_t, \mathbf{v}_t) = \mathbf{f}_{t-1} + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} h(\mathbf{v}_{t-1} + \Delta\mathbf{v}) + \frac{\partial \mathbf{f}}{\partial \mathbf{v}} \Delta\mathbf{v}, \tag{3.3}$$

where the Jacobians $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ and $\frac{\partial \mathbf{f}}{\partial \mathbf{v}}$ are evaluated at $\mathbf{f}_{t-1}$. Thus, a simple cloth simulation process involves solving for $\mathbf{v}_t$ using Eq. (3.2) and Eq. (3.3), and computing the subsequent simulation state as $\mathbf{x}_t = \mathbf{x}_{t-1} + h\mathbf{v}_t$.

However, there can additionally be self-collisions and collisions with obstacles during simulation. Harmon *et al.* (2008) determine the collision response at the impact zones to update the vertex positions appropriately:

$$\mathbf{x}_t = \mathbf{x}_t + \text{collision\_response}(\mathbf{x}_t, \mathbf{v}_t, \mathbf{x}_t^{obs}, \mathbf{v}_t^{obs}). \tag{3.4}$$

Fig. 3.2 shows an overview of cloth simulation pipeline. Since we want to use end-

Algorithm 1 Cloth simulation

$\mathbf{v}_0 \leftarrow \mathbf{0}$
for $t = 1$ **to** n **do**
  $\mathbf{M}, \mathbf{f} \leftarrow \text{compute\_forces}(\mathbf{x}, \mathbf{v})$
  $\mathbf{a}_t \leftarrow \mathbf{M}^{-1}\mathbf{f}$
  $\mathbf{v}_t \leftarrow \mathbf{v}_{t-1} + \mathbf{a}_t \Delta t$
  $\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} + \mathbf{v}_t \Delta t$
  $\mathbf{x}_t \leftarrow \mathbf{x}_t + \text{collision\_response}(\mathbf{x}_t, \mathbf{v}_t, \mathbf{x}_t^{obs}, \mathbf{v}_t^{obs})$
  $\mathbf{v}_t \leftarrow (\mathbf{x}_t - \mathbf{x}_{t-1})/\Delta t$
end for

Figure 3.2: Cloth is parameterised as a surface mesh with vertex states comprising of geometry $\mathbf{x}_t$ and velocity $\mathbf{v}_t$. Then, cloth simulation involves determining subsequent simulation states by computing the acceleration $\mathbf{a}_t$ due to acting forces $\mathbf{f}$ and surface mass $\mathbf{M}$. Image credit Liang *et al.* (2019)

to-end gradient-based optimisation, we need to backpropagate gradients through these steps. Due to the high dimensionality of the dynamical system when modelling cloth, a naïve gradient computation for the general system (Eq. (3.2)) and the collision response (Eq. (3.4)) can become impractical. Liang *et al.* (2019) propose a solution for this problem, and we proceed with their approach. Specifically, they use implicit differentiation for Eq. (3.2) and Eq. (3.4), where the gradient of the latter is approximated via QR decomposition of a much smaller constraint matrix. For more details on the backward pass, please refer to Liang *et al.* (2019).

3.3 Differentiable Renderer

Given ϕ-SfT's goal to recover time-varying surface deformations from monocular sequence, we aim to optimise for 3D geometry using analysis-by-synthesis. Differentiable rendering enables such an end-to-end optimisation by obtaining useful gradients of the image-based losses.

In computer graphics, rendering refers to the forward process of synthesising images of 3D scenes defined by geometry, materials, scene lights and camera parameters. Rendering is a complex task consisting of many operations, however

3.3 Differentiable Renderer

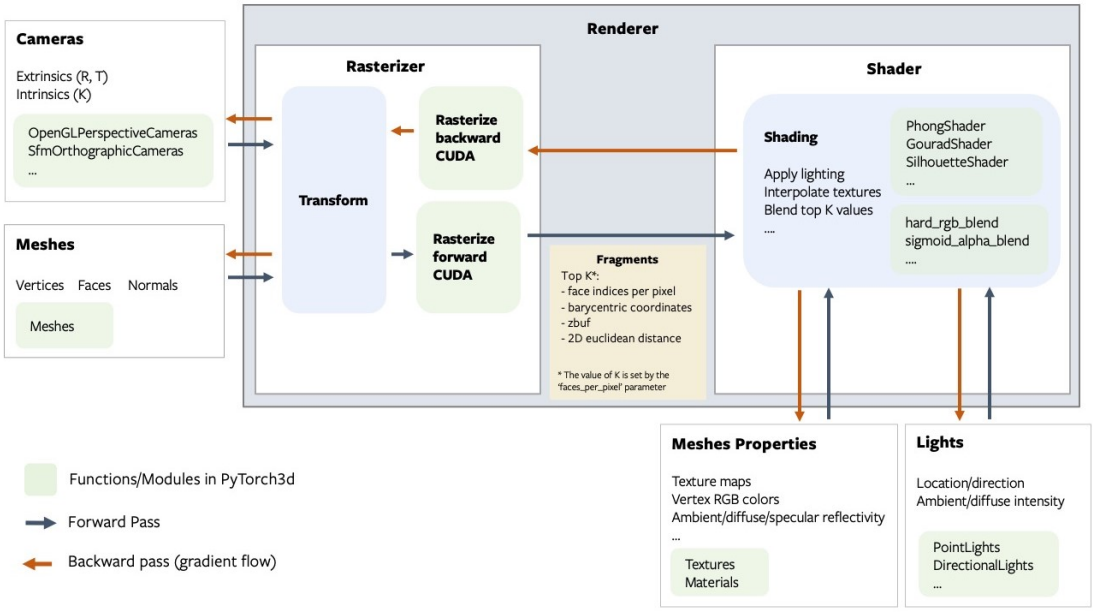

Figure 3.3: A differentiable renderer provides gradients of 3D scene properties with respect to rendered image. PyTorch3D comprises of a rasterizer and shader components, with the former efficiently implemented in CUDA. Image credit Ravi *et al.* (2020)

not all operations are analytically differentiable. Differentiable rendering (DR) constitutes a family of techniques that enable optimisation of the 3D scene parameters by backpropagating the gradients with respect to the rendered image. We refer the reader to a recent survey Kato *et al.* (2020) for further details on various DR methods. We use PyTorch3D (Ravi *et al.*, 2020) as a layer for optimising the objective function of ϕ-SfT.

PyTorch3D by Ravi *et al.* (2020) is a library for differentiable rendering of meshes and pointclouds. It is an efficient and modular implementation of *Soft Rasterizer* (Liu *et al.*, 2019), which introduces useful gradients by composing probability maps of rendered triangles into the final image. It performs fast 3D operations, supports batching of meshes and uses autograd functionality of PyTorch for automatically computing the gradients. Fig. 3.3 shows overview of the PyTorch3D architecture.

Chapter 4

ϕ-SfT Dataset

In this chapter, we describe the datasets used for evaluating ϕ-SfT, including the data acquisition setup and preprocessing.

Existing methods on monocular 3D reconstruction focus predominantly on large and global deformations. Therefore, the community lacks datasets of freely deforming clothes which are arguably the most challenging category of non-rigid objects. As ϕ-SfT aims to recover fine local surface deformations such as folds, we create new real and synthetic datasets to serve this need.

4.1 Real Sequences

Figure 4.1: We record a new real "ϕ-SfT dataset" of nine sequences with reference depth data to facilitate quantitative comparisons of monocular 3D surface reconstruction methods. Our setup consists of a synchronised RGBD camera. The depth camera is used to extract point clouds that serve as pseudo-ground-truth for evaluation.

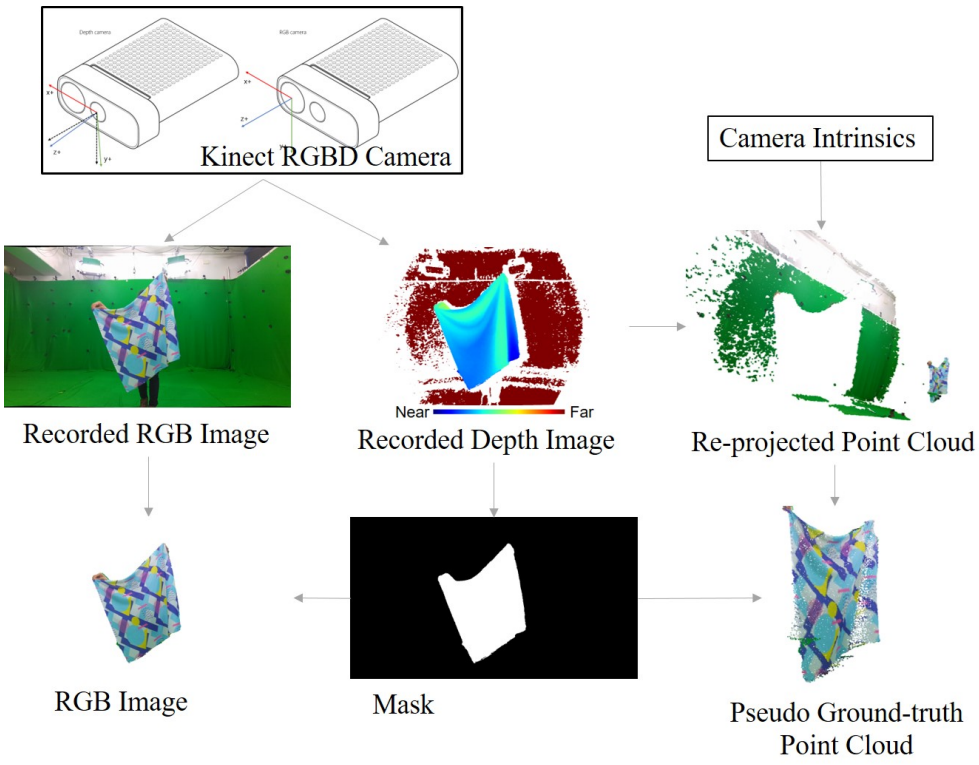

Figure 4.2: Overview of the capture setup: We expose cloth surfaces to external forces, such as gravity, wind, and hand contacts for generating challenging deformations. Each sequence is simultaneously recorded using a monocular RGB and depth camera (Azure Kinect). Then, pseudo-ground-truth deformations for each frame are reconstructed as a point cloud through backprojection, using the depth images and known camera intrinsics. We segment out the background from the captured images and point clouds by depth thresholding.

We have recorded a new dataset of deforming surfaces to allow quantitative evaluations of reconstruction methods on real data against pseudo ground truth. Fig. 4.1 shows an overview of the recorded sequences. The dataset has a total of nine sequences of various surface shapes and textures, including differing material properties due to differences in the cloths' fabric and weaving. There are more and less elastic, and more and less dense materials. The texture pattern varies from fine-grained and regular to more global and irregular patterns. The cloth size ranges from 55×55 to 95×95cm. Fig. 4.2 shows an overview of the capture setup and preprocessing. The surfaces are exposed to external forces, *i.e.,* gravity, wind, and hand contacts. Each sequence is simultaneously recorded using a monocular RGB and depth camera (Azure Kinect) and has a length of about 40 frames, such that they focus on challenging folds. The images in our real scenes have resolution 1920×1080 pixels. We segment out the background from the captured images and point clouds by depth thresholding. Then, pseudo-ground-truth deformations for each frame are reconstructed as a point cloud through backprojection, using the depth images and known camera intrinsics. Note that the correspondences between ground-truth pointclouds across all frames is therefore not available. The coordinate system of the point clouds is in meters.

4.2 Synthetic Sequences

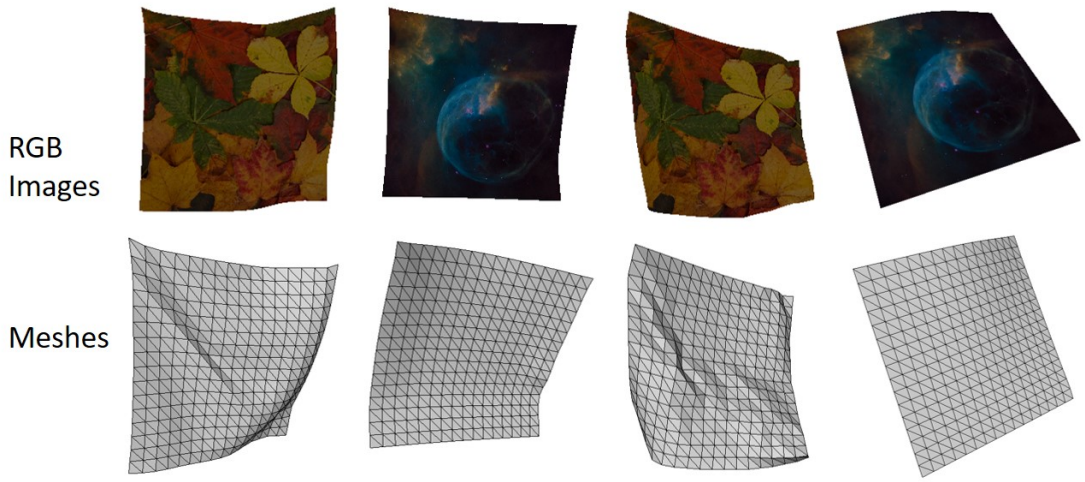

Figure 4.3: We generate a new synthetic "ϕ-SfT dataset" of four sequences with reference ground truth meshes to facilitate quantitative comparisons of monocular 3D surface reconstruction methods.

In addition to the real dataset described in Sec. 4.1, we create a synthetic dataset to enable more fair and thorough evaluation. Synthetic dataset provides ground truth meshes and the vertex correspondence for the surface across all

frames, unlike the real data sequences. This facilities better quantitative comparison with other monocular 3D surface reconstruction methods.

We generate a new synthetic dataset of four monocular RGB sequences of naturalistically deforming surfaces with different textures. We use the physics simulator by Liang *et al.* (2019) as described in Sec. 3.2 for generating deforming surface meshes. This is the same simulator used as part of our reconstruction pipeline later (see Cha. 5). Note that using the same simulator for both data generation and reconstruction can have small inductive bias. A flat square cloth of dimensions $1 \times 1m$ is provided in the form of a textured mesh to the simulator at the beginning of the simulation. The deformations at subsequent time points are caused by the varying gravity and wind forces acting on the cloth. Moreover, we vary elastic material properties of the cloth across the sequences, following Wang *et al.* (2011). Each sequence contains 50 frames, and the mesh contains 289 regularly-sampled vertices. Finally, the simulated cloth states are rendered as virtual images using PyTorch3D (Ravi *et al.*, 2020). The rendered images serve as inputs to the evaluated methods, and the obtained meshes are 3D ground truth. Fig. 4.3 shows an overview of the generated synthetic sequences.

Chapter 5

Method: ϕ-SfT

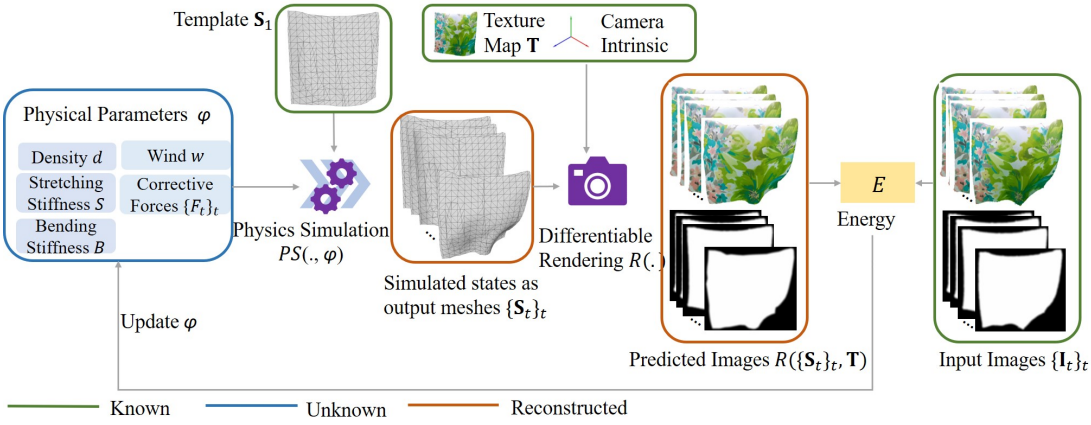

Figure 5.1: High-level method overview: Given a sequence of monocular input images $\{\mathbf{I}_t\}_t$, a template at the rest position $\mathbf{S}_1$ and the corresponding texture map $\mathbf{T}$, our technique solves for the unknown physical parameters ϕ that describe the deforming 3D surface $\{\mathbf{S}_t\}_t$. We optimise for the per-sequence physical parameters of $\{d, \mathcal{S}, \mathcal{B}, w\}$ as well as the per-frame corrective forces $\{\mathcal{F}_t\}_t$ in a gradient-based manner. We utilise (1) a physics-based differentiable simulator PS for reconstructing meshes with a physical deformation model and (2) a differentiable renderer R for projecting the reconstructions into image space, which allows us to define a reprojection error *over all pixels* (instead of vertices) during optimisation. The differentiable nature of both components enables us to back-propagate the gradients of the total energy E all the way back to the unknown physics parameters.

We propose ϕ-SfT: a new method for the 3D reconstruction of a deforming surface (such as cloth) from a monocular RGB video $\{\mathbf{I}_t\}_{t\in[1,\dots,T]}$ with known intrinsics. As is common for SfT methods (Ngo *et al.*, 2015; Yu *et al.*, 2015), we assume that the camera is static and take as input a flat rest shape of the target deformable surface $\mathbf{S}_1$ for $t = 1$ with a corresponding texture map $\mathbf{T}$. We also assume that a segmentation mask separating a foreground object and background is available. To encourage physically plausible deformations, we use a *full* physical model, described in Sec. 5.1, that explicitly models forces acting on the surface as

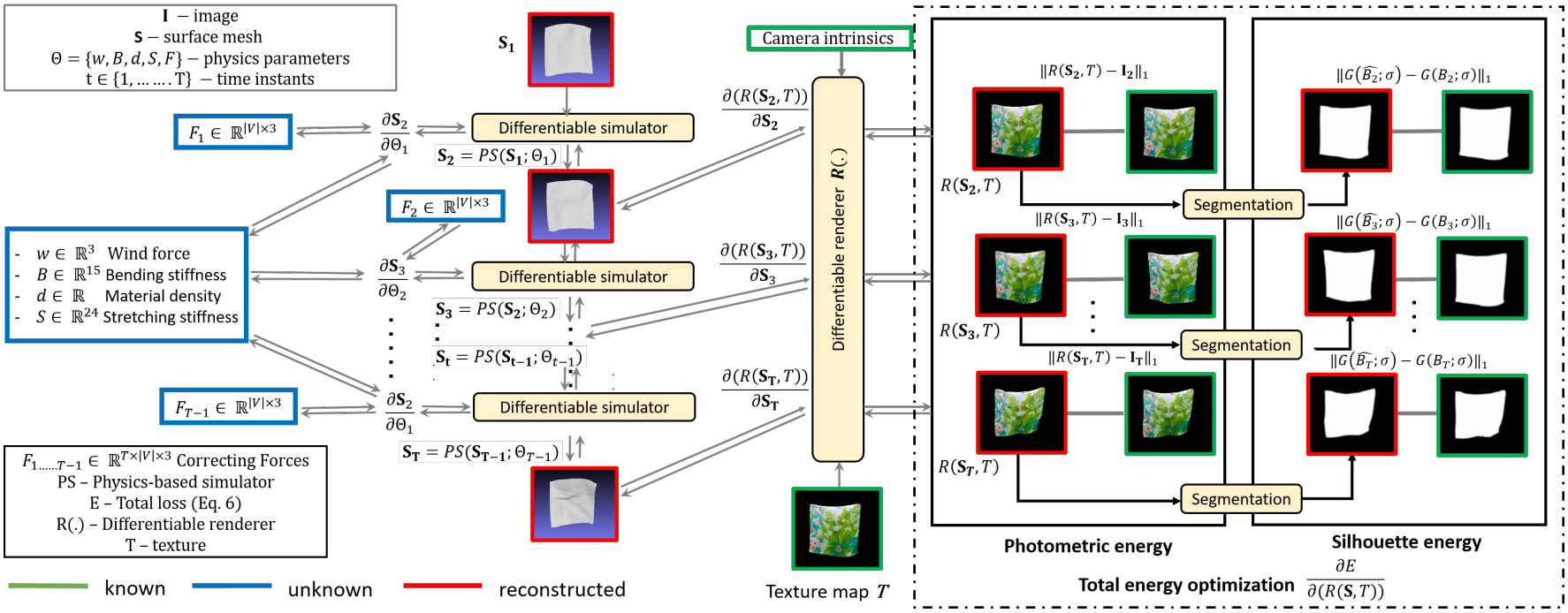

Figure 5.2: Detailed method overview: Given a sequence of monocular input images $\{\mathbf{I}_t\}_t$, a template at the rest position $\mathbf{S}_1$ and the corresponding texture map $\mathbf{T}$, our technique solves for the unknown physical parameters ϕ that describe the deforming 3D surface $\{\mathbf{S}_t\}_t$. We optimise for the per-sequence physical parameters of $\{d, \mathcal{S}, \mathcal{B}, w\}$ as well as the per-frame corrective forces $\{\mathcal{F}_t\}_t$ in a gradient-based manner. We utilise (1) a physics-based differentiable simulator PS for reconstructing meshes with a physical deformation model and (2) a differentiable renderer R for projecting the reconstructions into image space, which allows us to define a reprojection error *over all pixels* (instead of vertices) during optimisation. The differentiable nature of both components enables us to back-propagate the gradients of the total energy E all the way back to the unknown physics parameters. Note that the gradients are calculated automatically and provided here for completeness.

well as the underlying elastic properties of the material. A brief introduction to the basic concepts necessary to understand the ideas presented here can be found in Chapter 3. Sec. 5.2 presents the objective function we employ to relate the 2D observations to the estimated reconstructions. We describe how we optimise the objective function for the physical parameters in Sec. 5.3. See Figs. 5.1 and 5.2 for a high-level and detailed overview of our method.

Physics-based simulators are widely used in computer graphics for 3D simulations (Baraff & Witkin, 1998; Narain *et al.*, 2012), and differentiable versions exist (Liang *et al.*, 2019). Our idea is to use a differentiable physics simulator as a regulariser (deformation model prior) in monocular non-rigid 3D reconstruction. Its usage in SfT is, unfortunately, not straightforward. To integrate the physics-based simulator into our framework, we have to make several crucial improvements to it. First, the idealised assumptions of simulated environments cannot account for the variety of forces and effects causing surface deformations in the real world, such as wind turbulence. We take inspiration from the recent work on physically plausible 3D human motion capture (Shimada *et al.*, 2020), which uses a virtual force acting on the root joint of a human skeleton to account for the effects which are not explicitly considered by the physics model. We, therefore, introduce corrective forces accounting for mismatched assumptions about the natural scene (Sec. 5.1.1). Second, while most simulators used in computer graphics are designed to create simulations following 3D reference motions, it now has to be driven by the 2D input images, and the gradients need to be backpropagated from the image-based losses. In particular, our energy function includes a dense photometric loss and a silhouette loss (Sec. 5.2). Lastly, the optimisation strategy suitable for 3D simulations is not the best choice for our analysis-by-synthesis ϕ-SfT approach—optimising for deformed surface states, material properties and forces—and we propose an adaptive training strategy instead (Sec. 5.3).

5.1 Deformation Model

We seek to reconstruct a sequence of deforming surfaces as 2D manifold meshes in 3D space with fixed topology (edges $\mathbf{E}$), thereby providing temporal correspondences. A surface in this sequence can be parameterised as a triangular mesh $\mathbf{S}_t = \{\mathbf{V}_t, \mathbf{E}\}$ where the state of the i-th vertex in $\mathbf{V}_t$ comprises its 3D position $\mathbf{x}_t^i \in \mathbb{R}^3$ and its velocity $\mathbf{v}_t^i \in \mathbb{R}^3$.

5.1.1 Surface Parametrisation

At the core of our method, we model deformations of the non-rigid surface as a physical process, *i.e.,* as elastic deformations resulting from internal stretching

and bending forces as well as external forces acting on the surface. We thus do not treat the mesh states $\mathbf{S}_t$ as parameters but instead, use a physical parametrisation.

We initialise the differentiable cloth simulator from the template $\mathbf{S}_1$ and generate $\mathbf{S}_t$ for $t > 1$ with physics simulation PS (Sec. 3.2) according to the material parameters and external forces:

$$\mathbf{S}_t = PS(\mathbf{S}_{t-1}; \phi_{t-1}), \tag{5.1}$$

where ϕ_{t-1} are the estimated physics parameters, *i.e.,*

$$\phi_{t-1} = \{d, \mathcal{S}, \mathcal{B}, w, \mathcal{F}_{t-1}\}. \tag{5.2}$$

Here, material density $d \in \mathbb{R}$, stretching stiffness $\mathcal{S} \in \mathbb{R}^{24}$ (resistance to stretching), and bending stiffness $\mathcal{B} \in \mathbb{R}^{15}$ (resistance to bending and folding) all together describe the elastic properties of the material and, hence, determine the cloth's internal forces. We also optimise for the wind force $w \in \mathbb{R}^3$. However, the wind model is not sufficient to fully describe all the external forces in the scene, such as hand contacts and wind turbulence. We seek to correct for these model insufficiencies by additionally defining a set of corrective forces $\mathcal{F} = \{\mathcal{F}_t \in \mathbb{R}^{|\mathbf{V}|\times 3}\}_{t\in[1,\dots,T-1]}$. Note that these vary across vertices and across time. They can, in principle, account for any physical force that the simulator does not explicitly model. In the following, we use the shorthand $\phi = \{d, \mathcal{S}, \mathcal{B}, w, \mathcal{F}\}$. We next describe the objective function we use to optimise for the parameters ϕ of the differentiable simulator.

5.2 Objective Function

We now have a physical deformation model that is parametrised by ϕ and that outputs a 3D mesh $\mathbf{S}_t$ for time t. We solve for the optimal parameters ϕ^* by minimising the objective function $E = E_p + \lambda E_s$ (with $\lambda \in \mathbb{R}$):

$$\phi^* = \underset{\phi}{\operatorname{argmin}}\, E(\phi). \tag{5.3}$$

Since we are only given RGB images for time t>1, we define a photometric energy term E_p in the image space. Specifically, E_p is an ℓ_1 RGB data term to encourage photometric consistency between the reconstructed surface rendered into 2D and the input frames:

$$E_p = \sum_{t=2}^{T} \|R(\mathbf{S}_t, \mathbf{T}) - \mathbf{I}_t\|_1, \tag{5.4}$$

where $R(\cdot)$ is a differentiable renderer (Sec. 3.3) outputting a perspective projection of the input mesh (textured with $\mathbf{T}$) onto the image plane with known

intrinsics. We implement R with *Soft Rasterizer* (Liu *et al.*, 2019), which introduces useful gradients by composing probability maps of rendered triangles into the final image. This allows us 1) to define E_p densely *over all pixels*, instead of just the vertices; 2) use the information in the high-resolution $\mathbf{T}$ that would have been ignored if we had used a photometric term only at the vertices.

While the photometric energy term works well for local corrections, it does not provide a signal for mismatches that are farther apart in image space. To get a signal even for larger, coarser errors, we add a silhouette energy term that encourages consistency between the foreground segmentation mask of the input frames and the rendered 2D surface:

$$E_s = \sum_{t=2}^{T} \|G(B_t;\sigma) - G(\hat{B}_t;\sigma)\|_1. \tag{5.5}$$

Here, B and $\hat{B}$ are the foreground binary segmentation masks of the reconstructed and the input images, respectively. $G(\cdot,\sigma)$ represents a Gaussian filter of standard deviation σ. The Gaussian filter smooths the binary masks, extending the spatial area where informative gradients are obtainable. Without this Gaussian filter, non-zero gradients would be obtained only at pixels located immediately next to the silhouettes of both binary masks. Thus, if the silhouettes do not match almost perfectly at a pixel, the gradient would be zero there, providing no signal to the network as to the target direction to move the mesh's triangles. Importantly, both the ground-truth mask B and the rendered mask $\hat{B}$ are processed in the same way, ensuring that E_s is well-behaved.

Given our model and the objective function, we next look at how we solve the optimisation problem.

5.3 Optimisation

Our goal is to obtain the optimal physical parameters ϕ^* via Eq. (5.3). We use iterative, gradient-based optimisation to that end. Since both the simulator PS and the renderer R are differentiable, we can back-propagate gradients from the objective function E through the rendering to the meshes $\mathbf{S}_t$ and from there further through the physics simulation to the physical parameters ϕ (see Fig. 5.2).

5.3.1 Initialisation

To obtain a sufficiently accurate initial guess for the elastic properties $d, \mathcal{S}$, and $\mathcal{B}$, we set them to the average values of ten different real fabrics from Wang *et al.* (2011). The wind and corrective forces $\mathcal{F}$ are initialised to $\mathbf{0}$, *i.e.,* a zero vector. Note that this initialisation leads PS to initially generate surfaces $\{\mathbf{S}_t\}_t$ that are identical to the template $\mathbf{S}_1$.

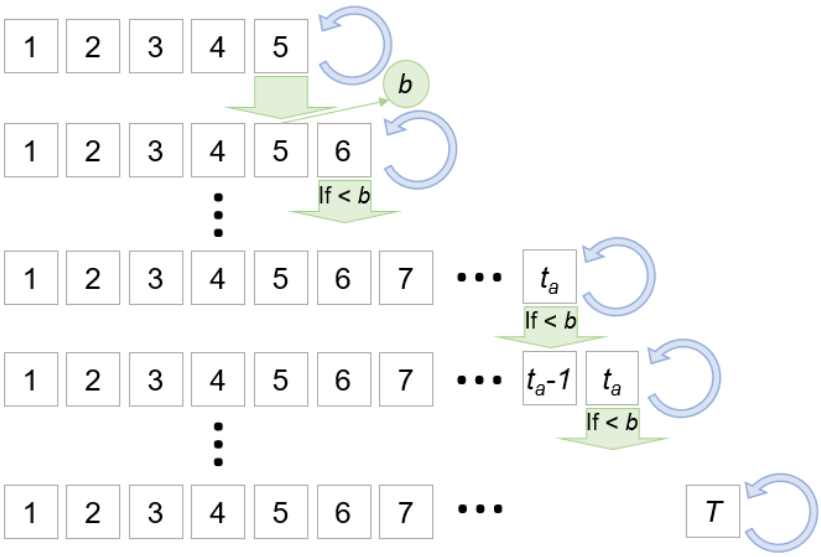

Figure 5.3: We use an adaptive scheme that initially optimises for the first five frames, then keeps the energy of frame 5 as the threshold b, and then incrementally adds the next frame whenever the energy of the latest active frame t_a decreases below b (or a maximum number of steps is reached).

5.3.2 Adaptive Optimisation Scheme

Since simulation is a temporal process, the physics parameters for the early frames directly influence the reconstructions of the later frames. In addition, later frames are initially reconstructed at lower fidelity than earlier ones. Therefore, similar to tracking a surface, we exploit the temporal order of the frames and do not optimise for all $t \geq 1$ from the start. Instead, we adaptively grow the temporal window that is *active*, *i.e.*, the latest time t_a up to which all earlier frames $t \leq t_a$ participate in the optimisation. We start with the first five frames as active and optimise E for them. Once the energy of the latest frame decreases below a threshold b, we add the next frame to the optimisation, see Fig. 5.3. b is set to the energy that the fifth frame has when the sixth frame is added. We assume the fifth frame to be well-reconstructed since it is early in the sequence. (In case the optimisation cannot reach the threshold, we set a maximum number of iterations, after which we progress regardless.) This adaptive scheme speeds up optimisation by converging to a reasonable guess for ϕ before later frames become active. Moreover, it allocates more iterations to frames with more challenging deformations. We evaluate this scheme experimentally in Sec. 6.3.

5.4 Implementation Details

We implement our approach in Pytorch (Ravi *et al.*, 2020). Eq. (5.3) is solved with optimiser Adam (Kingma & Ba, 2017) with learning rate 10^{-3}. The adaptive

optimisation scheme leads to several hundred iterations in most cases, which takes between 16 and 24 hours on an Nvidia RTX 8000 GPU. Due to the sequential nature of the simulator, we compute the objective function for all active times t for each optimisation iteration. We set $\sigma = 7px$ for E_s. We apply the corrective forces $\mathcal{F}$ by modifying the velocity of vertex i at time t: $v_t^i = v_t^i + \mathcal{F}_t^i$ (which is a valid implementation because both mass and time step size are constant). We keep the wind air density fixed at 1kg/m^3 and optimise only for the wind velocity.

The images in our real scenes have resolution 1920×1080 pixels. For real scenes (recorded with an RGB-D camera), pre-processing is more involved: We first segment out the background from the captured images and point clouds by depth thresholding. We next use Poisson surface reconstruction (Kazhdan & Hoppe, 2013) on the template point cloud (at $t = 1$), which yields $\sim$300 vertices on average. We then determine the initial rigid pose relative to a flat sheet (which is required by the simulator) using iterative closest point (ICP) (Cignoni *et al.*, 2008), and initialise the simulator with it. The first image $\mathbf{I}_1$ is used as texture map $\mathbf{T}$. We obtain the corresponding texture parameterisation by projecting the vertices of the template mesh $\mathbf{S}_1$ onto image space with known camera intrinsics.

Chapter 6

Evaluations

In this chapter, we present evaluations of our method in different settings.

We evaluate our technique on real and synthetic data qualitatively and quantitatively. We recorded the dataset (Sec. 4.1) of natural sequences using a monocular RGB camera together with the depth camera Azure Kinect. The latter is used to obtain pseudo-ground-truth segmentations and deformations. Moreover, we generated a synthetic dataset (Sec. 4.2) of challenging deformations with physics-based simulator (Liang *et al.*, 2019). As ground truth meshes are known and vertex correspondences are available across all frames, per-vertex geometry and normal errors can be evaluated for synthetic data sequences. Qualitative and quantitative results on real dataset (Sec. 6.1) and synthetic dataset (Sec. 6.2) show our technique clearly outperforms state of the art by capturing a wider variety of deformations and local folds. We also perform an ablation study in Sec. 6.3 that demonstrates the importance of corrective forces and other design choices.

6.1 Real Sequences

6.1.1 Compared Methods

We compare our technique to SfT methods, namely Yu *et al.* (2015)'s **Direct, Dense, Deformable (DDD)**, Ngo *et al.* (2015)'s **Ngo2015** and Shimada *et al.* (2019)'s **IsMo-GAN**, and NRSfM methods, Sidhu *et al.* (2020)'s **Neural NRSfM (N-NRSfM)** and Parashar *et al.* (2020)'s **Diff-NRSfM**. Since NRSfM methods accept 2D point correspondences, we track 2D points densely across the input images with multi-frame subspace flow (Garg *et al.*, 2013b; MFSF), as suggested in Sidhu *et al.* (2020). The first frame of the sequence is selected as a keyframe for tracking. We provide DDD with the required hierarchy of coarse-to-fine templates and Ngo2015 with the same template used by ϕ-SfT. Additionally, to support the need for the physical simulation based on the parameters $\{d, \mathcal{S}, \mathcal{B}, w\}$, we show the result of a baseline ("Only $\mathcal{F}$") where the only optimisation parameters are the corrective parameters $\{\mathcal{F}_t\}_t$. As other physics parameters are completely

omitted, we implement this baseline as an optimisation of per-vertex offsets over time. Note that this does not use physics simulator PS and therefore not physics-aware.

6.1.2 Metric

Due to the possible scale and depth ambiguity, we globally align reconstructions of all methods to ground truth in a rigid body fashion. We first determine the transformation matrix using Procrustes alignment, which is further refined with per-frame ICP.

For quantitative evaluation, we compute the Chamfer distance between the pseudo-ground-truth point cloud from Kinect G and points M sampled from the reconstructed mesh:

$$Ch_{G,M} = \frac{1}{|G|} \sum_{\mathbf{g} \in G} \min_{\mathbf{m} \in M} \|\mathbf{g} - \mathbf{m}\|_2^2 + \frac{1}{|M|} \sum_{\mathbf{m} \in M} \min_{\mathbf{g} \in G} \|\mathbf{m} - \mathbf{g}\|_2^2, \tag{6.1}$$

We perform quantitative comparison with two evaluation settings: (a) using Procrustes alignment on the reference frame, and (b) using per-frame ICP after Procrustes initialisation. While Procrustes-only evaluation is sufficient to globally align reconstructions of all methods to the ground truth, per-frame ICP improves this alignment. We also note that using per-frame alignment introduces temporal noise to the reconstructed sequences.

6.1.3 Results

We show qualitative reconstruction results for arbitrary frames on all real sequences in Fig. 6.1. ϕ-SfT deformation model reconstructs challenging surface deformations well by capturing both coarse shape and local folds. Our physics-based approach provides a reasonable prior for self-occluded surface parts (S3 and S4 in Fig. 6.1). See Fig. 6.3 for depth maps reconstructed by our approach. This alternative way to visualise the results allows us to study and perceive even smaller details on the reconstructed surfaces.

Figs. 1.2 and 6.2 show that ϕ-SfT outperforms related methods qualitatively. Quantitatively, the 3D reconstruction error $Ch_{G,M}$ of ϕ-SfT is an order of magnitude lower compared to the tested methods when using single global alignment (Tab. 6.1) and on average lower when using per-frame global alignment (Tab. 6.2). Moreover, the reduction in $Ch_{G,M}$ is lower for ϕ-SfT when moving from global to per-frame alignment in comparison to all other methods (compare Tabs. 6.1 and 6.2). This suggests that our method is the most temporally coherent.

The results confirm that SfT and NRSfM, both relying on simple geometric prior assumptions (such as surface smoothness, isometry or small local deformations), cannot cope with such elaborate fold patterns as those present in our

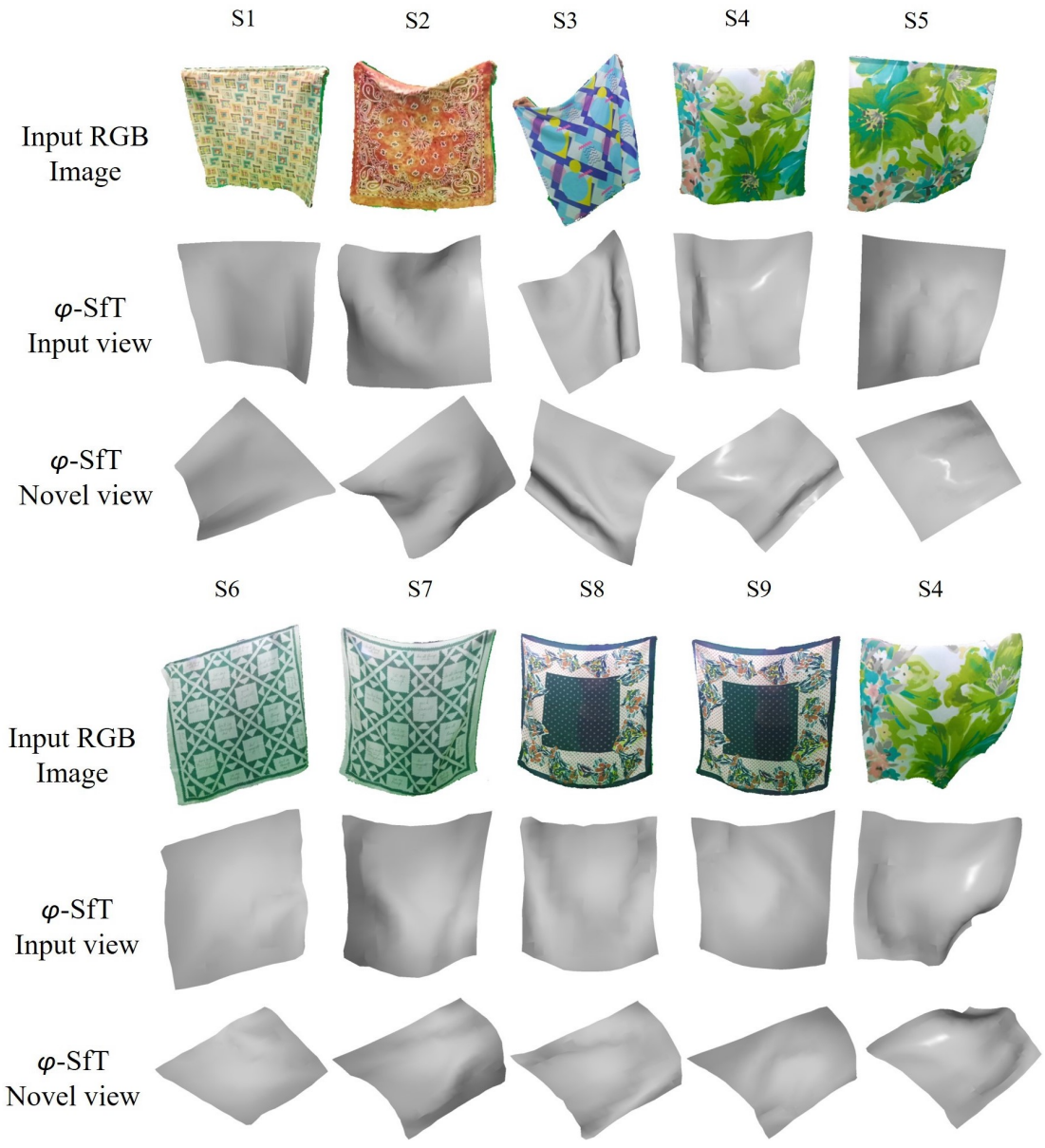

Figure 6.1: We show qualitative results on all real sequences. For the given RGB image, the reconstructed mesh is visualised in the input camera view as well as novel camera view. ϕ-SfT accurately reconstructs the coarse shape and local folds.

6.1 Real Sequences

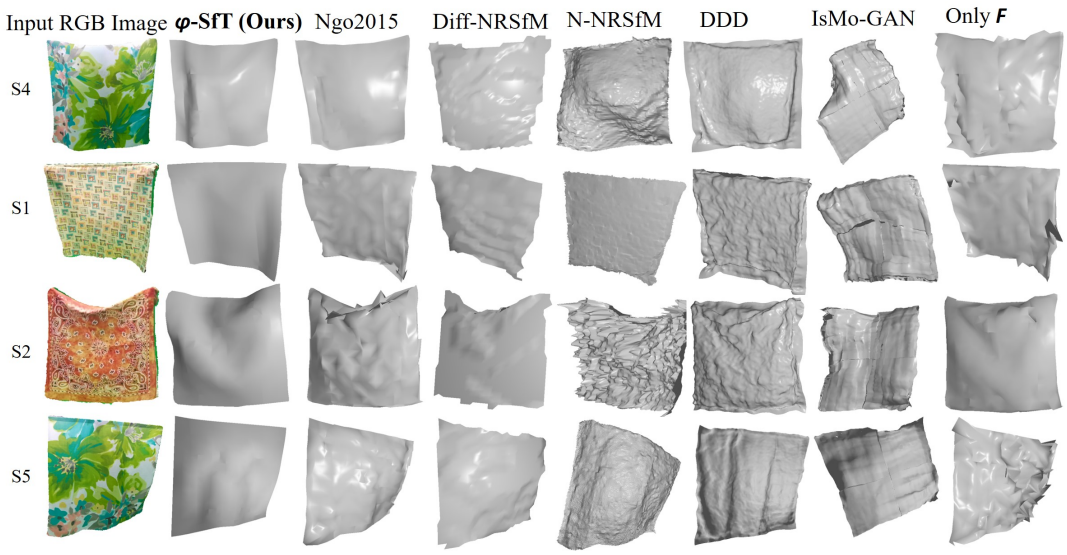

Figure 6.2: Qualitative comparisons of several tested methods Ngo *et al.* (2015); Parashar *et al.* (2020); Shimada *et al.* (2019); Sidhu *et al.* (2020); Yu *et al.* (2015), including ϕ-SfT, for an arbitrary frame of *real* S1, S2, S4 and S5 sequences. Our results are significantly more accurate and, unlike the other methods, capture the folds well.

Seq.	IsMo-GAN	N-NRSfM	DDD	Diff-NRSfM	ϕ-SfT
S1	91.90	101.19	52.31	155.43	**16.64**
S2	32.93	310.62	**3.76**	9.84	11.54
S3	47.88	183.23	10.15	71.94	**6.99**
S4	283.47	177.29	64.29	139.51	**7.80**
S5	267.86	446.54	110.28	153.34	**9.85**
S6	137.03	103.76	12.95	28.78	**11.00**
S7	113.11	195.16	65.18	88.25	**9.22**
S8	68.96	111.19	24.21	21.67	**3.31**
S9	76.88	37.06	36.36	48.56	**2.86**
Average	124.45	185.12	42.17	79.70	**8.80**

Table 6.1: We quantitatively compare our method to the state of the art on the ϕ-SfT *real* dataset after *Procrustes alignment on the reference frame*. We measure the Chamfer distance between the Kinect point cloud and points sampled from the reconstructed meshes (and multiply by 10^4 for readability). By an order of magnitude on average, our technique significantly outperforms all related methods on all sequences, except for S2.

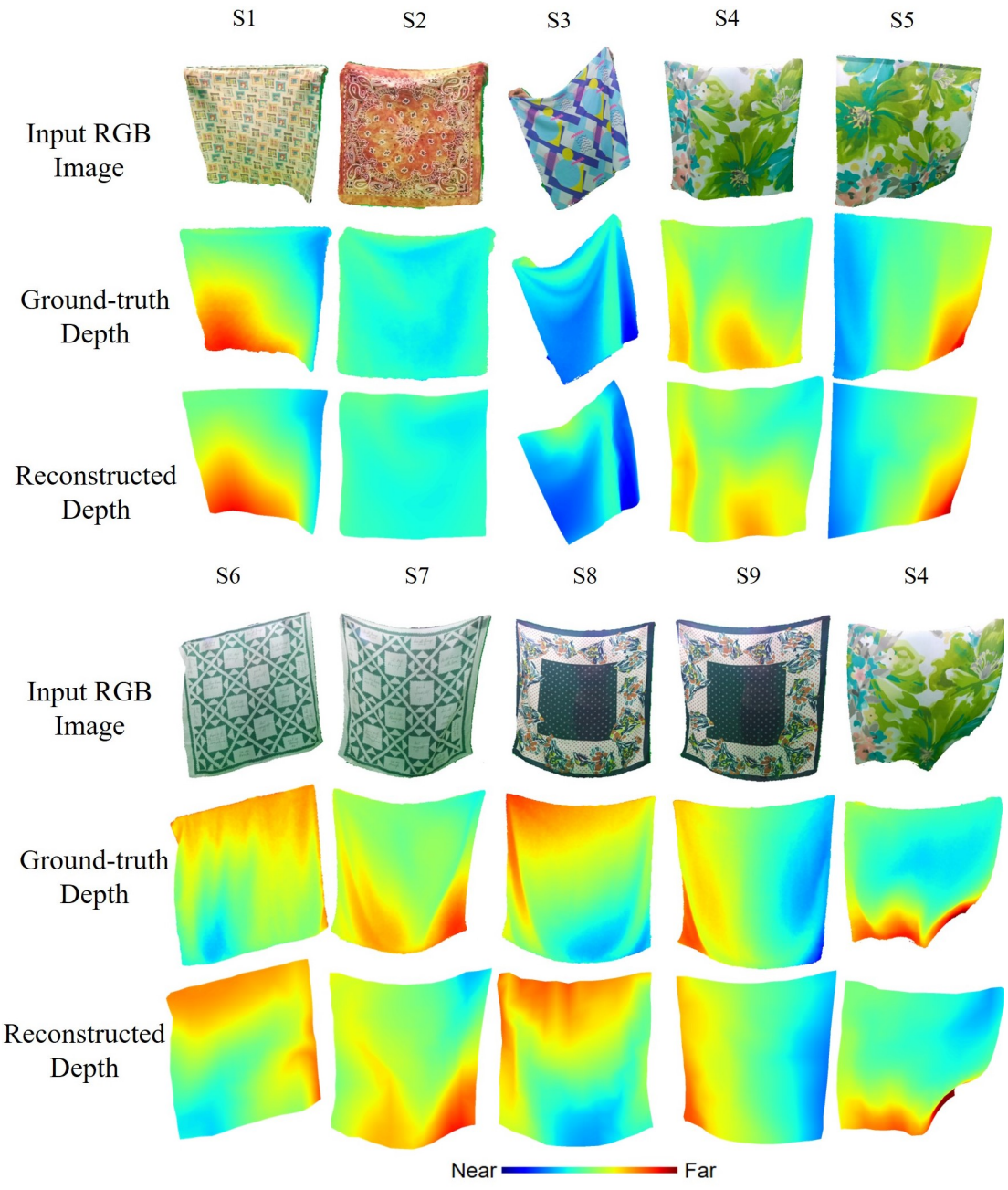

Figure 6.3: We show qualitative results as colour-coded depth maps on all real sequences. For the given RGB image, the ground-truth depth map exhibits similar features as our reconstructed depth. Both the coarse shape and local folds are well captured.

6.1 Real Sequences

Seq.	IsMo-GAN	N-NRSfM	DDD	Diff-NRSfM	Ngo2015	Only $\mathcal{F}$	ϕ-SfT
S1	19.69	8.25	2.95	17.14	2.19	2.59	**0.89**
S2	22.18	33.62	1.69	4.46	**1.51**	1.60	2.75
S3	33.54	104.60	3.80	4.40	**2.17**	3.23	3.54
S4	90.30	77.02	25.73	41.37	15.90	14.95	**7.80**
S5	92.78	72.66	10.46	26.92	10.72	21.32	**7.53**
S6	57.62	8.73	6.97	14.02	**3.01**	3.08	6.20
S7	49.27	129.44	15.64	12.49	7.95*	6.03	**4.73**
S8	24.45	38.06	7.61	9.91	fail	3.78	**2.52**
S9	53.12	19.81	11.77	5.29	fail	4.39	**2.36**
Avg.	49.22	54.69	10.87	15.11	5.92*	6.77	**4.26**

Table 6.2: We quantitatively compare our method to the state of the art on the ϕ-SfT *real* dataset after *per-frame ICP alignment*. We measure the Chamfer distance between the Kinect point cloud and points sampled from the reconstructed meshes (and multiply by 10^4 for readability). Our technique outperforms all related methods on *average*. '*' indicates that the method failed on few challenging frames. We exclude these frames during error computation.

dataset. The results of N-NRSfM follow the silhouettes of the input images better than DDD and IsMO-GAN, thanks to 2D point tracking, even though its $Ch_{G,M}$ is the highest, due to bad shape initialisation obtained under rigidity assumption using Tomasi-Kanade approach (Sidhu *et al.*, 2020; Tomasi & Kanade, 1992). Its 3D surfaces are somewhat rugged, and the surface pattern allows to recognise the observed texture (third row in Fig. 6.2), again, due to the specifics of dense 2D point tracking. Moreover, we see that, as expected, more fine-grained textures result in more accurate 2D point tracking by the MFSF approach. It is known that DDD cannot follow large deformations, and we observe that it does not manage to track the silhouettes in our tests. IsMo-GAN, trained on rather smooth surfaces, cannot reproduce local folds and barely captures the contours. Diff-NRSfM produces reasonable reconstructions in smooth regions owing to its differentiable structure preserving formulation. However, the method is sensitive to noise in point correspondences and leads to visual artifacts in the case of challenging folds. Ngo2015 failed fully on two scenes and partially on S7, we thus exclude the last few (challenging) frames from Ngo2015's error on S7 in Tab. 6.2. Ngo2015 struggles to faithfully reconstruct surfaces and often leads to noisy and physically implausible results. Ngo2015 and DDD has better numbers on a few scenes (S2, S3 and S6), but even on these we obtain better qualitative results (Figs. 1.2 and 6.2 compares S3 and S2 respectively). This suggests that an isometry prior (as in Ngo *et al.* (2015)) is insufficient compared to our physics-based elastic model, which can even express non-isometric deformations (depending on the parameters). When removing all forces from the model except for correctives ($\mathcal{F}$), the results degrade in quality and average error increases >50%, see "Only

Seq.	IsMo-GAN		N-NRSfM		DDD		Diff-NRSfM		Only $\mathcal{F}$		ϕ-SfT	
	e_{3D}	e_n	e_{3D}	e_n	e_{3D}	e_n	e_{3D}	e_n	e_{3D}	e_n	e_{3D}	e_n
S1	0.066	34.27	0.167	34.34	0.043	33.86	0.053	**11.30**	0.054	12.73	**0.042**	11.86
S2	0.077	45.11	n/a	n/a	0.036	25.20	0.055	11.35	0.069	14.92	**0.023**	**10.62**
S3	0.096	36.72	0.113	26.36	0.066	42.16	0.077	17.59	0.059	15.83	**0.033**	**9.12**
S4	0.078	41.16	0.077	24.36	0.023	19.86	0.063	5.69	0.043	9.33	**0.005**	**2.61**
Avg.	0.079	39.32	0.119	28.35	0.042	30.27	0.062	11.48	0.056	13.20	**0.026**	**8.55**

Table 6.3: e_{3D} and e_n after rigid alignment with *per-frame Procrustes* on our *synthetic* scenes. Ours gives most accurate results.

$\mathcal{F}$" in Fig. 6.2 & Tab. 6.2. The failure of this baseline demonstrates that accuracy of the ϕ-SfT model, in the extreme case, does not lie solely on the corrective forces. In contrast to other methods and the baseline, ϕ-SfT estimates temporally coherent surfaces and captures all significant folds while missing only small nuances.

6.2 Synthetic Sequences

6.2.1 Compared Methods

Similar to ϕ-SfT's real dataset, we compare our synthetic dataset results to SfT methods, namely Yu *et al.* (2015)'s **Direct, Dense, Deformable (DDD)** and Shimada *et al.* (2019)'s **IsMo-GAN**, and the NRSfM methods Sidhu *et al.* (2020)'s **Neural NRSfM (N-NRSfM)** and Parashar *et al.* (2020)'s **Diff-NRSfM**. Since ground truth meshes are available in the case of synthetic dataset, we compute ground truth 2D point correspondences as input to Diff-NRSfM and Neural NRSfM (N-NRSfM). We provide DDD with the required hierarchy of coarse-to-fine templates and Ngo2015 with the same template used by ϕ-SfT. Additionally, we show the result of a baseline (Only $\mathcal{F}$) where the only optimisation parameters are the corrective parameters $\{\mathcal{F}_t\}_t$.

6.2.2 Metrics

Since vertex correspondences are known across all surface states in the synthetic dataset, we align reconstructions of all methods to ground truth in a rigid body fashion using *per-frame Procrustes*.

As in previous methods (Ngo *et al.*, 2015; Parashar *et al.*, 2020; Shimada *et al.*, 2019; Sidhu *et al.*, 2020), we use 3D error assuming known correspondences to express the reconstruction accuracy on the new dataset:

$$e_{3D} = \frac{1}{|T|} \sum_{t \in [1,\dots,T]} \frac{\|\mathbf{G}_t - \mathbf{M}_t\|_F}{\|\mathbf{G}_t\|_F} \tag{6.2}$$

where $\mathbf{G}_t$ and $\mathbf{M}_t$ are the vertices of ground truth and reconstructed mesh, respectively, and $\|\cdot\|_F$ denotes the Frobenius norm. To better capture the error in local deformations, we additionally compute the per-vertex angular error in degrees as

$$e_n = \frac{180}{\pi |T||N|} \sum_{t\in[1,\ldots,T]} \sum_{i\in[1,\ldots,N]} \cos^{-1}(\mathbf{g}_t^i \cdot \mathbf{m}_t^i) \tag{6.3}$$

where $\mathbf{g}_t^i$ and $\mathbf{m}_t^i$ are the unit normals at the ith vertex in frame t of the ground truth and reconstructed mesh, respectively.

6.2.3 Results

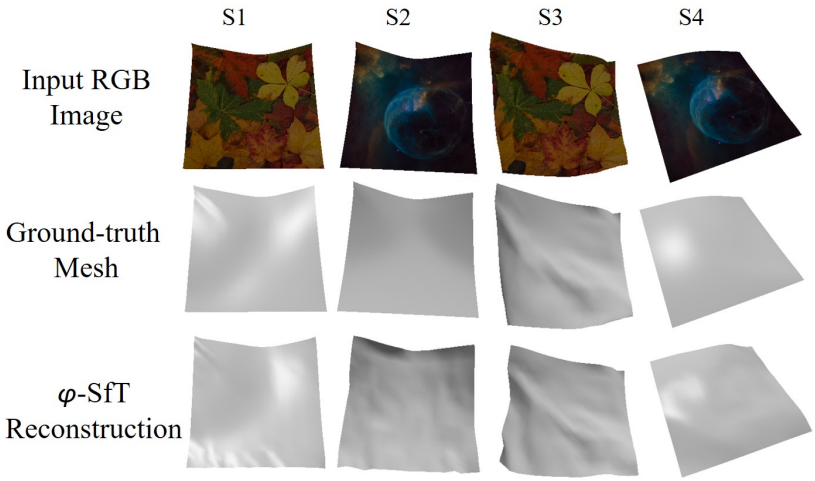

Figure 6.4: We show qualitative results on all *synthetic* sequences. For the given RGB image, the ground truth mesh and reconstructed mesh are visualised in the input camera view. ϕ-SfT accurately reconstructs physically plausible and accurate surfaces.

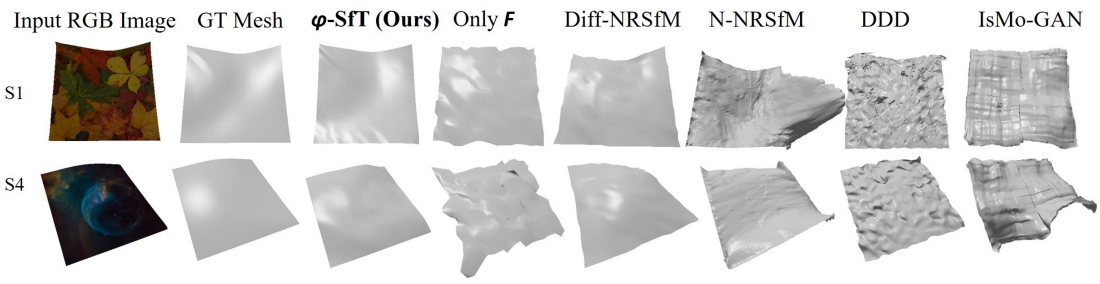

Figure 6.5: Qualitative comparisons of several tested methods Parashar *et al.* (2020); Shimada *et al.* (2019); Sidhu *et al.* (2020); Yu *et al.* (2015), including ϕ-SfT, for an arbitrary frame of *synthetic* S1 and S4 sequences. Our results are significantly more accurate and, unlike the other methods, is physically plausible.

In Fig. 6.4, we show qualitative results on all synthetic sequences. ϕ-SfT accurately reconstructs physically plausible and accurate surfaces. Fig. 6.5 shows that ϕ-SfT outperforms related methods qualitatively, similar to the observations on real dataset. This demonstrates that SfT and NRSfM, both relying on simple geometric prior assumptions, struggle to estimate physically plausible surfaces. We also note that Diff-NRSfM performs better on our synthetic dataset as opposed to real sequences, as the deformations here are global and smooth (see Fig. 6.5).

We refer to Tab. 6.3 for mean vertex error, e_{3D}, and mean angular normal error in degrees, e_n, on our synthetic data with per-frame Procrustes using ground truth correspondences. We outperform others on all synthetic sequences, except for Diff-NRSfM with e_n on S1. Since vertex correspondences are available in the case of synthetic dataset, this allows for more faithful alignment as well as better metric, *i.e.,* per-frame Procrustes over per-frame ICP and e_{3D}, e_n over $Ch_{G,M}$. As shown in Tab. 6.3, our method has lowest average e_{3D} suggesting it reconstructs global deformations and lowest average e_n suggesting we also better capture local folds.

6.3 Ablative Study

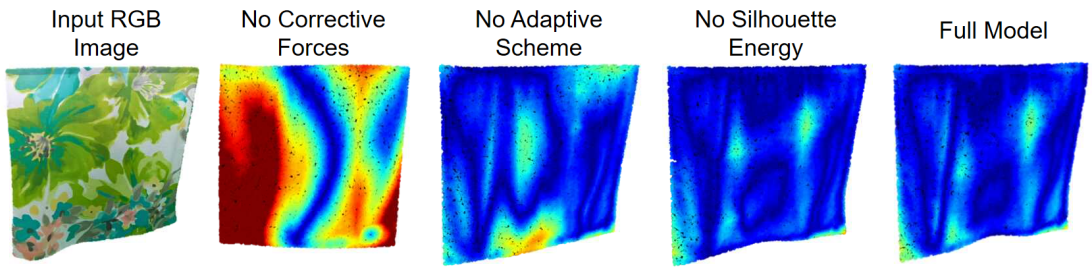

Figure 6.6: In the ablative study, we remove corrective forces, the adaptive scheme, or the silhouette energy term. The corrective forces are the most crucial component to make our method work.

We conduct an ablative study on the various design choices we made to integrate the physics simulator into the SfT setting and test the following modes: 1) Operation without corrective forces $\mathcal{F}$, 2) Influence of the adaptive training by considering all frames from the start (Sec. 5.3), and 3) Disabling the silhouette term (5.5). We evaluate several sequences on the real dataset and report the Chamfer distance against pseudo ground truth in Tab. 6.4. We observe that omitting $\mathcal{F}$ always leads to a significant increase in the error, and abandoning our adaptive training policy increases it by a factor of two. Fig. 6.6 shows qualitative results. Qualitative visualisations confirm the statistics over all sequences, *i.e.,*

6.3 Ablative Study

Sequence	w/o E_s	w/o $\mathcal{F}$	w/o adaptive	Full
S1	16.64	59.10	18.01	**16.41**
S2	**11.54**	27.76	22.08	12.93
S3	**6.99**	28.21	14.76	10.59
S4	9.05	84.30	16.75	**7.80**
S5	9.85	118.65	12.83	**9.09**
S6	11.00	26.30	13.58	**11.00**
S7	9.22	63.76	9.1	**7.82**
S8	4.74	13.92	4.11	**3.31**
S9	3.23	28.87	5.76	**2.86**
Average	9.14	50.10	15.12	**9.09**

Table 6.4: We evaluate the various design choices of our method by removing the silhouette energy term, the external forces, and the adaptive optimisation scheme. Here, we compute the Chamfer distance between the Kinect point clouds and points sampled from the reconstructed meshes after *Procrustes alignment on the reference frame* (and multiply by 10^4 for readability). We use sequences from the new ϕ-SfT *real* dataset.

the largest errors are present in the colour-coded error maps when $\mathcal{F}$ is disabled. The second most crucial component of ϕ-SfT is the adaptive training strategy which is vital when addressing the inverse problem of SfT (but which might not be as useful in 3D simulations, *i.e.,* when solving a *direct* problem). Note that E_s helps when the structure deforms and changes its shape in the re-projection significantly. This is the case for most sequences, however sequence S3 for instance, (Fig. 6.2, Tab. 6.4) has less global deformations and significant local fold, in which case E_s isn't very helpful. Also, E_s is susceptible to error in input segmentation whereas the photometric energy E_p is robust to the same. We note that our method is not sensitive to initialisation of physical parameters. We empirically find no issue with always initialising with our default material.

Chapter 7

Applications

In this chapter, we show a few applications of our method, that highlight the advantages of ϕ-SfT's surface deformation formulation. We demonstrate that the optimised physical parameters are meaningful and inferred well enough to enable intuitive editing.

7.1 Semantic Material Editing

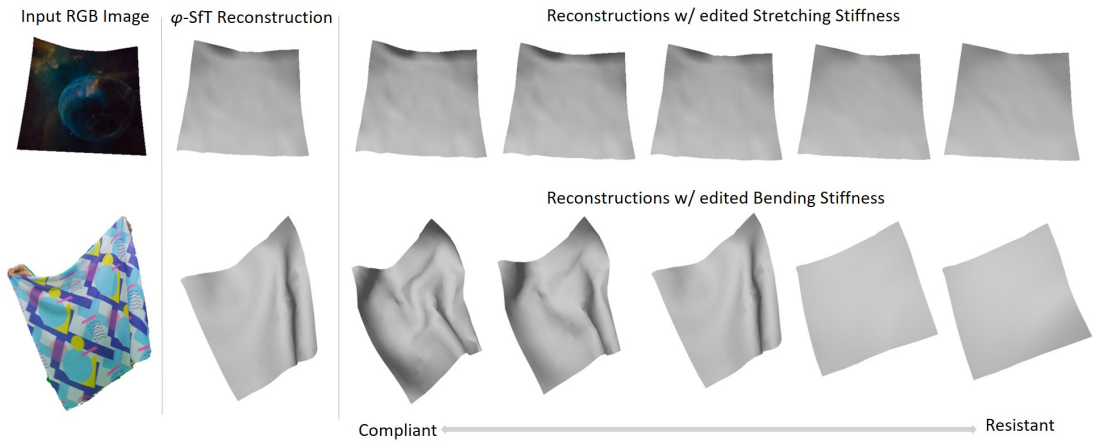

Figure 7.1: We show semantic controllability of surface material by scaling optimised stretching stiffness $\mathcal{S}$ (top row) and bending stiffness $\mathcal{B}$ (bottom row). The deformations introduced in coarse shape and local folds suggest that intuitive editing is possible.

ϕ-SfT parameterises the material elasticity of non-rigidly deforming surface with density d, stretching stiffness $\mathcal{S}$ (resistance to stretching) and bending stiffness $\mathcal{B}$ (resistance to bending). These parameters describe the behaviors characteristic of real cloth materials as explained in Sec. 3.1. Given a monocular sequence of deforming surface, ϕ-SfT allows optimising for material $\{d, \mathcal{S}, \mathcal{B}\}$ which uniquely describes the deformations along with optimised forces $\{w, \mathcal{F}\}$.

Given optimised physical parameters ϕ^*, we aim to generate new deformations at test-time by modifying the inferred material parameters. To this end, we run physics simulation by scaling stretching stiffness $\mathcal{S}$ while re-using the other physical parameters $\{d^*, \mathcal{B}^*, w^*, \mathcal{F}^*\}$. In Fig. 7.1, we visualise an example where $\mathcal{S}$ is scaled by factors of $\frac{1}{20}$(most compliant), $\frac{1}{10}$, 1, 10 and 20 (most resistant). Similarly, we scale bending stiffness $\mathcal{B}$ by factors of $\frac{1}{10}$(most compliant), $\frac{1}{5}$, 1, 5 and 10 (most resistant) as shown Fig. 7.1. The results demonstrate that $\boldsymbol{\phi}$-SfT's estimation of material parameters not only describes the underlying surface, it additionally allows for semantic control over the reconstruction.

7.2 Intuitive Surface Animation

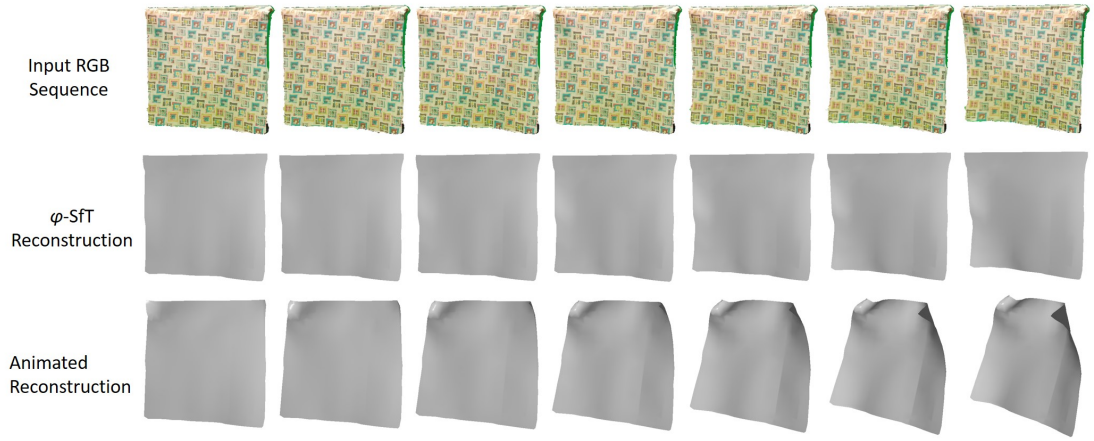

Figure 7.2: We generate new animation at test-time by modifying the inferred corrective forces. By specifying external forces $\mathcal{F}_{ext}$ on the top corner vertices, we obtain an animated reconstruction (bottom row) that seamlessly integrates the original deformation (middle row) with the new forces. The animation result is physically plausible, smooth and intuitive.

Alongside elastic model for cloth, $\boldsymbol{\phi}$-SfT models the deformations using wind and corrective forces $\{w, \mathcal{F}\}$. Given optimised physical parameters ϕ^*, we can generate new deformations at test-time by modifying the inferred corrective forces. As an example, we create an animated version of the reconstruction by applying external forces on the two upper corners as shown in Fig. 7.2. This is achieved by running a physics simulation with optimal physical parameters $\{d^*, \mathcal{S}^*, \mathcal{B}^*, w^*\}$ and $\mathcal{F} = \mathcal{F}^* + \mathcal{F}_{ext}$ where $\mathcal{F}_{ext}$ is the force applied on the two upper corner vertices over all frames. For instance, this could be the additional hand motion. The animation result is physically plausible, smooth and intuitive.

Chapter 8

Discussion and Future Work

In this chapter, we discuss the limitations of our method and possible ways to mitigate them. We also briefly discuss the high-level observations across multiple experiments and the possible future directions for this project.

8.1 Limitations

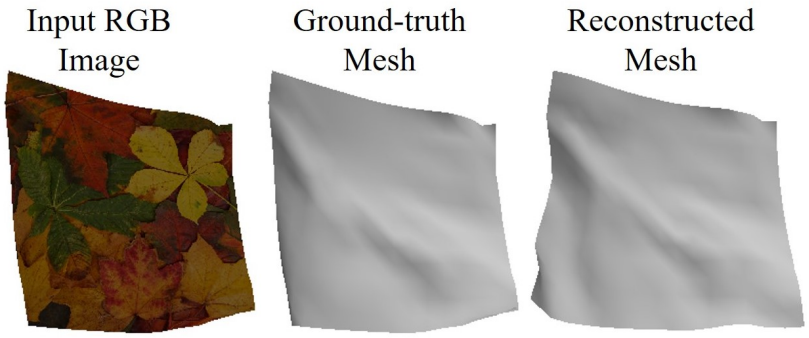

Figure 8.1: We find that few synthetic sequences incorrectly fold at surface corners due to the inherent problem of depth ambiguity in monocular reconstruction. Notice that reconstructed mesh folds outwards in the lower left corner, whereas the ground truth mesh folds inwards in the same region.

Depth Ambiguity

Depth ambiguity is an inherent problem in monocular 4D surface reconstruction. Our physics-aware model provides reasonable prior for plausible and accurate reconstruction of deforming surface. However, in some cases, the method may incorrectly reconstruct the bending corners/edges as shown in Fig. 8.1 when the optimisation process gets stuck in local minima. We notice this limitation in the case of synthetic data sequences as it uses perfectly flat template. In the case of

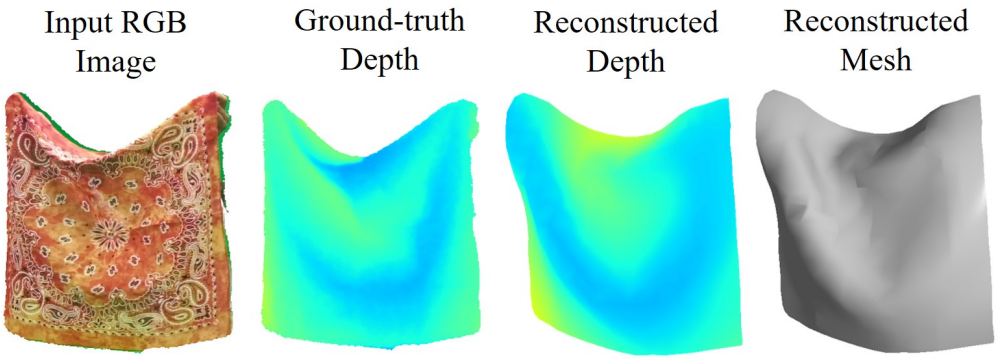

Figure 8.2: As the runtime of physical simulator is high (16 hours for optimising a single sequence), we use lower resolution for reconstructed mesh. This limits the reconstruction accuracy as we cannot capture very fine wrinkles (top region of input RGB image) even though we capture local folds.

real dataset sequences, slight deformation in the template is already helpful in convergence.

Longer Optimisation Times

The runtime of our approach is comparably high, for *e.g.,* we require twice as long compared to N-NRSfM (Sidhu *et al.*, 2020). This is due to single-threading when resolving collisions in the physics simulator; this can be improved with engineering and parallelisation in the future. We use lower resolution template meshes for all sequences, with 300 vertices as it is not feasible to optimise model parameters with higher resolution meshes. This limits the reconstruction accuracy as we cannot capture very fine wrinkles, as shown in 8.2. However, note that our photometric energy is defined densely over all pixels, instead of just the vertices, and thus uses the information in high-resolution texture map. Therefore, the lower number of vertices do not hurt the ϕ-SfT's ability to capture folds.

Missing Information about Environment Map/Light Sources

In the case of real dataset, the method has no prior knowledge of lights in the recording environment. PyTorch3d only supports a single light source, which differs from the environment where the cloths were recorded. We use the RGB image of the first frame as texture map for rendering during optimisation. This can neither account for shadows and nor the transparency of the real clothes. There is, therefore, a systematic mismatch between the rendered images and the input images during training. ϕ-SfT nevertheless achieves high accuracy, as we empirically find our differentiable rendering loss is robust to moderate photometric discrepancies.

Modelling Limitation of the Physics Simulator

Moreover, in theory, even more sophisticated physics simulators could be implemented, which could take into account more physical laws (*e.g.*, wind turbulence or electrostatic forces). Such requirements depend on downstream applications such as game design, movie production or industrial quality control.

8.2 Discussion

Existing methods on monocular 3D reconstruction focus predominantly on large and global deformations. As ϕ-SfT aims to recover fine local surface deformations in the challenging non-rigid category of clothes, we create new real and synthetic datasets to serve this need.

In the qualitative results on both the datasets, we observe that ϕ-SfT model reconstructs challenging surface deformations and leads to physically plausible results. Our approach is significantly more accurate than related methods (Ngo *et al.*, 2015; Parashar *et al.*, 2020; Shimada *et al.*, 2019; Sidhu *et al.*, 2020; Yu *et al.*, 2015) and supports finer-scale local folds, which is demonstrated on a wider spectrum of deformations in extensive experiments (see Sec. 6). Quantitatively on *real* dataset, the 3D reconstruction error $Ch_{G,M}$ of ϕ-SfT is an order of magnitude lower compared to the tested methods when using single global alignment and on average lower when using per-frame global alignment. We outperform others on *synthetic* dataset as well, when evaluating mean vertex error, e_{3D}, and mean angular normal error in degrees, e_n, after per-frame Procrustes using ground truth correspondences. We observe that our method has lowest average e_{3D} suggesting it reconstructs global deformations and lowest average e_n suggesting we also better capture local folds. The results confirm that SfT and NRSfM, both relying on simple geometric prior assumptions (such as surface smoothness, isometry or small local deformations), cannot cope with such elaborate fold patterns as those present in our dataset. When removing all forces from the model except for correctives ($\mathcal{F}$), the results degrade in quality and average error increases >50%. The failure of this baseline demonstrates that accuracy of the ϕ-SfT model, does not lie solely on the corrective forces, but rather on the internal forces due to the material elastic modelling. In contrast to other methods and the baseline, ϕ-SfT estimates temporally coherent surfaces and captures all significant folds while missing only small nuances.

The reasons for the better performance of our method are manifold. First, our approach explicitly models the physical fold formation process, and its parameters are hence physically meaningful. Our differentable physics simulation approach acts as a regulariser, provides a reasonable prior for self-occluded surface parts and can even express non-isometric deformations. Second, differentiable renderer ensures that the reprojections of the recovered 3D states accurately match the

observed images. In contrast to earlier photometric terms used for SfT (Yu *et al.*, 2015), using differentiable rendering allows us for the first time to define the reprojection error densely *per pixel* and not only *per vertex*. Thus, we exploit the rich information present in the texture regardless of the mesh resolution.

Limitations We do not address accurate inference of deformation forces and cloth materials due to the inherent force-elasticity ambiguity, however they are inferred well enough to enable intuitive editing (see Sec. 7). Also, note that our method takes significantly longer to optimise compared to the competing methods, with differentiable physics simulation being the computation bottleneck.

8.3 Future Work

Given the novelty of our work and the demonstrated improvements over SOTA, we see significant potential to provoke further research.

Physically-Aware 3D Reconstruction of Humans in Clothing

While we do not target complex objects or new scenarios, such as the separate field of cloth simulations for virtual dresses, this can be an interesting future avenue. As this being more challenging setup than ours, we can use for additional supervision from depth signals and/or optical flow estimated by off-the-shelf methods. Owing to ϕ-SfT's analysis-by-synthesis approach that uses differentiable rendering, incorporating these additional energies as soft constraint is straightforward.

Monocular 3D Reconstruction of Volumetric Non-Rigid Objects

In this work, we demonstrated using physics simulation as a regulariser for *classical SfT*. We can extend ϕ-SfT model to the reconstruction of deformable solids. Similar to triangular mesh for surfaces, we can use tetrahedral parameterisation for volumetric objects. It is possible to use physical models such as the Neo-Hookean model of Smith *et al.* (2018) that provides parameters to control the tetrahedral element's resistance to shearing and volumetric strains. These may be specified on a per-element basis, further allowing to represent heterogeneous materials.

Supervised Learning

As we propose an end-to-end differentiable framework for optimising physical parameters, we can use this in supervised setting for learning-based tasks. Monocular 4D cloth reconstruction can be coupled with deep learning to solve problems

such as detail refinement, garment retargeting, and material estimation. For instance, in an extension to Li *et al.* (2021)'s human performance capture with cloth deformation, hard physics-based constraints can be imposed to refine geometric details at test time.

Chapter 9

Conclusion

We introduced ϕ-SfT, a new optimisation-based SfT method that models deformations with a physical simulator and uses differentiable rendering to define a reprojection energy term over all pixels, exploiting texture information independent of the mesh resolution. As existing methods reconstruct predominantly global deformations, there are no datasets of scenes with local folds. We therefore create new real and synthetic datasets of the clothes, since they belong to the most challenging class of non-rigid objects. Experiments on the new real and synthetic datasets demonstrate that our approach improves the reconstructions qualitatively and quantitatively by a significant margin over competing techniques of several method classes. Especially remarkable is ϕ-SfT's accuracy in folded surface regions. This is due to awareness of the physical fold formation process attributable to the elastic properties of the materials and forces acting on them. Additionally, we showed that our physical modelling enables intuitive scene editing by modifying optimised material elasticity and acting forces. One of the limitation of ϕ-SfT is it's high runtime for optimisation attributable to the back-propagation through physics simulation. We believe that the proposed technique has a high potential for future research, and we hope to see more improvements on physically principled methods for monocular non-rigid 3D reconstruction. Moreover, we plan to generalise our model to complex objects or new scenarios, such as physically plausible and accurate reconstructions of humans in clothing.

References

Agudo, A. & Moreno-Noguer, F. (2015). Learning shape, motion and elastic models in force space. In *International Conference on Computer Vision (ICCV)*. 5

Agudo, A. & Moreno-Noguer, F. (2018). A scalable, efficient, and accurate solution to non-rigid structure from motion. *Computer Vision and Image Understanding (CVIU)*, **167**, 121–133. 5

Akhter, I., Sheikh, Y., Khan, S. & Kanade, T. (2009). Nonrigid structure from motion in trajectory space. In *Advances in Neural Information Processing Systems (NeurIPS)*. 5

Baraff, D. & Witkin, A. (1998). Large steps in cloth simulation. *ACM Transactions on Graphics*, 43–54. 10, 19

Bartoli, A., Gérard, Y., Chadebecq, F., Collins, T. & Pizarro, D. (2015). Shape-from-template. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **37**, 2099–2118. 6

Bregler, C., Hertzmann, A. & Biermann, H. (2000). Recovering non-rigid 3d shape from image streams. In *Computer Vision and Pattern Recognition (CVPR)*. 1, 5

Cashman, T.J. & Fitzgibbon, A.W. (2013). What shape are dolphins? building 3d morphable models from 2d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **35**, 232–244. 7

Chen, C.H., Tyagi, A., Agrawal, A., Drover, D., MV, R., Stojanov, S. & Rehg, J.M. (2019). Unsupervised 3d pose estimation with geometric self-supervision. In *Computer Vision and Pattern Recognition (CVPR)*. 6

Choy, C.B., Xu, D., Gwak, J., Chen, K. & Savarese, S. (2016). 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European Conference on Computer Vision (ECCV)*. 7

REFERENCES

Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F. & Ranzuglia, G. (2008). MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference*. 23

Dai, Y., Li, H. & He, M. (2014). A simple prior-free method for non-rigid structure-from-motion factorization. In *Computer Vision and Pattern Recognition (CVPR)*. 5

Fuentes-Jimenez, D., Pizarro, D., Casillas-Perez, D., Collins, T. & Bartoli, A. (2021). Texture-generic deep shape-from-template. *IEEE Access*, **9**, 75211–75230. 1, 6

Garg, R., Roussos, A. & Agapito, L. (2013a). Dense variational reconstruction of non-rigid surfaces from monocular video. In *Computer Vision and Pattern Recognition*. 1, 5

Garg, R., Roussos, A. & Agapito, L. (2013b). A variational approach to video registration with subspace constraints. *International Journal of Computer Vision (IJCV)*, **104**, 286–314. 25

Guo, J., Li, J., Narain, R. & Park, H.S. (2021). Inverse simulation: Reconstructing dynamic geometry of clothed humans via optimal control. In *Computer Vision and Pattern Recognition (CVPR)*, 14698–14707. 7

Harmon, D., Vouga, E., Tamstorf, R. & Grinspun, E. (2008). Robust treatment of simultaneous collisions. *SIGGRAPH (ACM Transactions on Graphics)*, **27**, 1–4. 11

Jaques, M., Burke, M. & Hospedales, T. (2020). Physics-as-inverse-graphics: Unsupervised physical parameter estimation from video. In *International Conference on Learning Representations*. 7

Kanazawa, A., Tulsiani, S., Efros, A.A. & Malik, J. (2018). Learning category-specific mesh reconstruction from image collections. In *European Conference on Computer Vision (ECCV)*. 7

Kandukuri, R., Achterhold, J., Moeller, M. & Stueckler, J. (2020). Learning to identify physical parameters from video using differentiable physics. In *Proc. of the 42th German Conference on Pattern Recognition (GCPR)*, gCPR 2020 Honorable Mention, preprint https://arxiv.org/abs/2009.08292. 7

Kato, H., Beker, D., Morariu, M., Ando, T., Matsuoka, T., Kehl, W. & Gaidon, A. (2020). Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*. 12

Kazhdan, M. & Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Trans. Graph.*, **32**. 23

Kingma, D.P. & Ba, J. (2017). Adam: A method for stochastic optimization. *arXiv e-prints*. 22

Kumar, S., Cherian, A., Dai, Y. & Li, H. (2018). Scalable dense non-rigid structure-from-motion: A grassmannian perspective. In *Computer Vision and Pattern Recognition (CVPR)*. 5

Li, X., Liu, S., De Mello, S., Kim, K., Wang, X., Yang, M.H. & Kautz, J. (2020). Online adaptation for consistent mesh reconstruction in the wild. In *NeurIPS*. 1, 7

Li, Y., Habermann, M., Thomaszewski, B., Coros, S., Beeler, T. & Theobalt, C. (2021). Deep Physics-aware Inference of Cloth Deformation for Monocular Human Performance Capture. In *3D Vision (3DV)*. 7, 41

Liang, J., Lin, M. & Koltun, V. (2019). Differentiable cloth simulation for inverse problems. In *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32. ix, 7, 11, 16, 19, 25

Liu, S., Li, T., Chen, W. & Li, H. (2019). Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *International Conference on Computer Vision (ICCV)*. 12, 21

Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A. & Sheikh, Y. (2019). Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*. 8

MFSF (2015). http://www0.cs.ucl.ac.uk/staff/lagapito/subspace_flow/. 25

Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R. & Ng, R. (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, 405–421, Springer. 8

Murthy, J.K., Macklin, M., Golemo, F., Voleti, V., Petrini, L., Weiss, M., Considine, B., Parent-Lévesque, J., Xie, K., Erleben, K., Paull, L., Shkurti, F., Nowrouzezahrai, D. & Fidler, S. (2021). gradsim: Differentiable simulation for system identification and visuomotor control. In *International Conference on Learning Representations*. 7

Narain, R., Samii, A. & O'Brien, J.F. (2012). Adaptive anisotropic remeshing for cloth simulation. *ACM Transactions on Graphics*, 147:1–10. 19

REFERENCES

Ngo, D.T., Park, S., Jorstad, A., Crivellaro, A., Yoo, C.D. & Fua, P. (2015). Dense image registration and deformable surface reconstruction in presence of occlusions and minimal texture. In *International Conference on Computer Vision (ICCV)*. 1, 5, 6, 17, 25, 28, 30, 31, 39

Novotny, D., Ravi, N., Graham, B., Neverova, N. & Vedaldi, A. (2019). C3dpo: Canonical 3d pose networks for non-rigid structure from motion. In *International Conference on Computer Vision (ICCV)*. 6

Östlund, J., Varol, A., Ngo, D.T. & Fua, P. (2012). Laplacian meshes for monocular 3d shape recovery. In *European Conference on Computer Vision (ECCV)*. 1

Paladini, M., Bue, A.D., Stošić, M., Dodig, M., ao Xavier, J. & Agapito, L. (2009). Factorization for non-rigid and articulated structure using metric projections. In *Computer Vision and Pattern Recognition (CVPR)*. 5

Parashar, S., Pizarro, D., Bartoli, A. & Collins, T. (2015). As-rigid-as-possible volumetric shape-from-template. In *International Conference on Computer Vision (ICCV)*. 1

Parashar, S., Salzmann, M. & Fua, P. (2020). Local non-rigid structure-from-motion from diffeomorphic mappings. In *Computer Vision and Pattern Recognition (CVPR)*. 5, 6, 25, 28, 31, 32, 39

Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M. & Martin-Brualla, R. (2021). Nerfies: Deformable neural radiance fields. *ICCV*. 8

Perriollat, M., Hartley, R. & Bartoli, A. (2011). Monocular template-based reconstruction of inextensible surfaces. *International Journal of Computer Vision (IJCV)*, **95**, 124–137. 1, 2, 6

Pumarola, A., Agudo, A., Porzi, L., Sanfeliu, A., Lepetit, V. & Moreno-Noguer, F. (2018). Geometry-Aware Network for Non-Rigid Shape Prediction from a Single View. In *Computer Vision and Pattern Recognition (CVPR)*. 6

Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.Y., Johnson, J. & Gkioxari, G. (2020). Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*. 12, 16, 22

Rempe, D., Guibas, L.J., Hertzmann, A., Russell, B., Villegas, R. & Yang, J. (2020). Contact and human dynamics from monocular video. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 6

Sahasrabudhe, M., Shu, Z., Bartrum, E., Güler, R.A., Samaras, D. & Kokkinos, I. (2019). Lifting AutoEncoders: Unsupervised Learning of a Fully-Disentangled 3D Morphable Model Using Deep Non-Rigid Structure From Motion. Computer Vision (ICCV) Workshops. 6

Salzmann, M., Pilet, J., Ilic, S. & Fua, P. (2007). Surface deformation models for nonrigid 3d shape recovery. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **29**, 1481–1487. 1, 2, 6

Salzmann, M., Urtasun, R. & Fua, P. (2009). Local deformation models for monocular 3d shape recovery. In *Computer Vision and Pattern Recognition (CVPR)*. 2

Shimada, S., Golyanik, V., Theobalt, C. & Stricker, D. (2019). IsMo-GAN: Adversarial learning for monocular non-rigid 3d reconstruction. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*. 1, 5, 6, 25, 28, 31, 32, 39

Shimada, S., Golyanik, V., Xu, W. & Theobalt, C. (2020). Physcap: Physically plausible monocular 3d motion capture in real time. *ACM Transactions on Graphics*, **39**. 6, 19

Sidhu, V., Tretschk, E., Golyanik, V., Agudo, A. & Theobalt, C. (2020). Neural dense non-rigid structure from motion with latent space constraints. In *European Conference on Computer Vision (ECCV)*. 5, 6, 25, 28, 30, 31, 32, 38, 39

Sitzmann, V., Zollhöfer, M. & Wetzstein, G. (2019). Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems*. 8

Smith, B., Goes, F.D. & Kim, T. (2018). Stable neo-hookean flesh simulation. *ACM Trans. Graph.*, **37**. 40

Stoll, C., Gall, J., de Aguiar, E., Thrun, S. & Theobalt, C. (2010). Video-based reconstruction of animatable human characters. In *ACM SIGGRAPH Asia*. 6

Stoyanov, D. (2012). Stereoscopic scene flow for robotic assisted minimally invasive surgery. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 479–486, Springer. 1

Tomasi, C. & Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization method. *Int. J. Comput. Vis.*, 137–154. 30

REFERENCES

Torresani, L., Hertzmann, A. & Bregler, C. (2008). Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **30**, 878–892. 5

Tretschk, E., Tewari, A., Golyanik, V., Zollhöfer, M., Lassner, C. & Theobalt, C. (2021). Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *IEEE International Conference on Computer Vision (ICCV)*, IEEE. 8

Varol, A., Salzmann, M., Fua, P. & Urtasun, R. (2012). A constrained latent variable model. In *2012 IEEE conference on computer vision and pattern recognition*, 2248–2255, Ieee. 1

Wang, C. & Lucey, S. (2021). Paul: Procrustean autoencoder for unsupervised lifting. In *Computer Vision and Pattern Recognition (CVPR)*, 434–443. 6

Wang, H., Ramamoorthi, R. & O'Brien, J.F. (2011). Data-driven elastic models for cloth: Modeling and measurement. *ACM Transactions on Graphics*, 71:1–11. 9, 10, 16, 21

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W. & Jiang, Y.G. (2018). Pixel2mesh: Generating 3d mesh models from single rgb images. In *European Conference on Computer Vision (ECCV)*. 7

Weiss, S., Maier, R., Cremers, D., Westermann, R. & Thuerey, N. (2020). Correspondence-free material reconstruction using sparse surface constraints. *Computer Vision and Pattern Recognition (CVPR)*, 4685–4694. 7

Wu, S., Jakab, T., Rupprecht, C. & Vedaldi, A. (2021). DOVE: Learning deformable 3d objects by watching videos. *arXiv preprint arXiv:2107.10844*. 7

Yang, G., Sun, D., Jampani, V., Vlasic, D., Cole, F., Chang, H., Ramanan, D., Freeman, W.T. & Liu, C. (2021a). Lasr: Learning articulated shape reconstruction from a monocular video. In *Computer Vision and Pattern Recognition (CVPR)*, 15980–15989. 7

Yang, G., Sun, D., Jampani, V., Vlasic, D., Cole, F., Liu, C. & Ramanan, D. (2021b). Viser: Video-specific surface embeddings for articulated 3d shape reconstruction. *Advances in Neural Information Processing Systems*, **34**. 7

Yang, G., Vo, M., Natalia, N., Ramanan, D., Andrea, V. & Hanbyul, J. (2021c). Banmo: Building animatable 3d neural models from many casual videos. *arXiv preprint arXiv:2112.12761*. 8

Yu, R., Russell, C., Campbell, N.D.F. & Agapito, L. (2015). Direct, dense, and deformable: Template-based non-rigid 3d reconstruction from rgb video. In *International Conference on Computer Vision (ICCV)*. 2, 3, 5, 6, 17, 25, 28, 31, 32, 39, 40