Pattern-Based Cloth Registration and Sparse-View Animation

OSHRI HALIMI, Technion – Israel Institute of Technology, Israel and Meta Reality Labs, USA
TUUR STUYCK, Meta Reality Labs, USA
DONGLAI XIANG, Carnegie Mellon University, USA and Meta Reality Labs, USA
TIMUR BAGAUTDINOV, Meta Reality Labs, USA
HE WEN, Meta Reality Labs, USA
RON KIMMEL, Technion – Israel Institute of Technology, Israel
TAKAAKI SHIRATORI, Meta Reality Labs, USA
CHENGLEI WU, Meta Reality Labs, USA
YASER SHEIKH, Meta Reality Labs, USA
FABIAN PRADA, Meta Reality Labs, USA

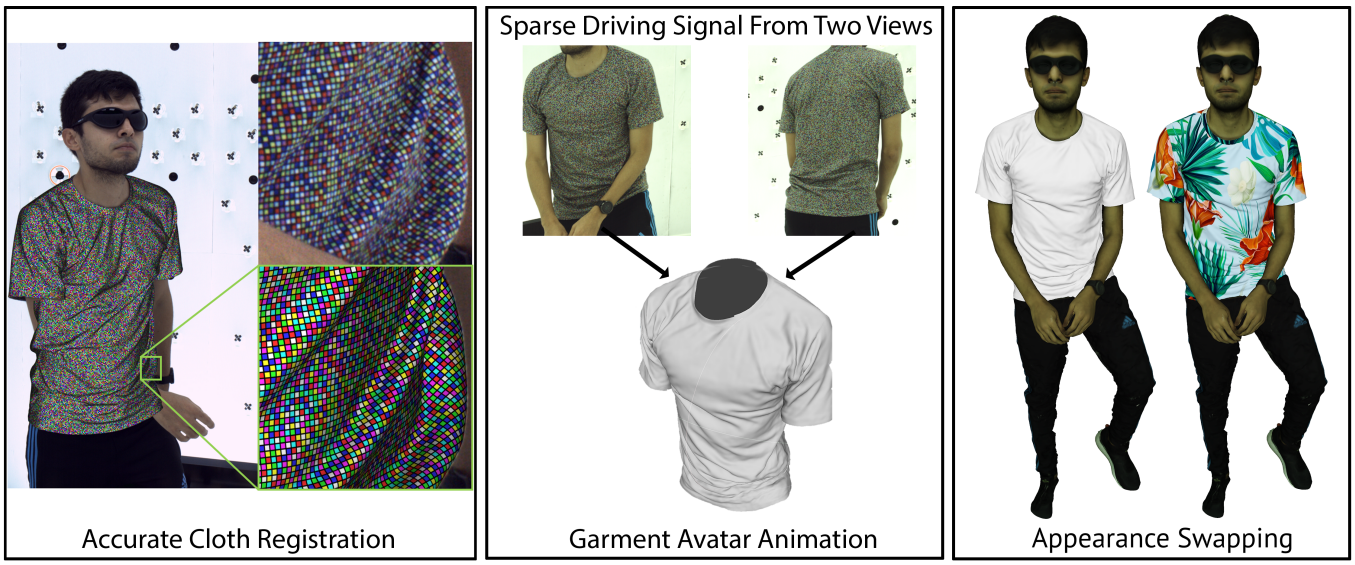


Fig. 1. **Left:** Comparison of our cloth registration and capture image. **Middle:** Garment Avatar animated from two cameras. **Right:** Appearance editing with our drivable garment avatar.

Authors' addresses: Oshri Halimi, Technion – Israel Institute of Technology, Israel and Meta Reality Labs, USA, oshri.halimi@gmail.com; Tuur Stuyck, Meta Reality Labs, USA, tuur@fb.com; Donglai Xiang, Carnegie Mellon University, USA and Meta Reality Labs, USA, donglaix@cs.cmu.edu; Timur Bagautdinov, Meta Reality Labs, USA, timurb@fb.com; He Wen, Meta Reality Labs, USA, hewen@fb.com; Ron Kimmel, Technion – Israel Institute of Technology, Israel, ron@cs.technion.ac.il; Takaaki Shiratori, Meta Reality Labs, USA, tshiratori@fb.com; Chenglei Wu, Meta Reality Labs, USA, chenglei@fb.com; Yaser Sheikh, Meta Reality Labs, USA, yasers@fb.com; Fabian Prada, Meta Reality Labs, USA, fabianprada@fb.com.

0730-0301/2022/12-ART196
https://doi.org/10.1145/3550454.3555448

We propose a novel multi-view camera pipeline for the reconstruction and registration of dynamic clothing. Our proposed method relies on a specifically designed pattern that allows for precise video tracking in each camera view. We triangulate the tracked points and register the cloth surface in a fine-grained geometric resolution and low localization error. Compared to state-of-the-art methods, our registration exhibits stable correspondence, tracking the same points on the deforming cloth surface along the temporal sequence. As an application, we demonstrate how the use of our registration pipeline greatly improves state-of-the-art pose-based drivable cloth models. Furthermore, we propose a novel model, *Garment Avatar*, for driving cloth from a dense tracking signal which is obtained from two opposing camera views. The method produces realistic reconstructions which are faithful to the actual geometry of the deforming cloth. In this setting, the user wears a garment with our custom pattern which enables our driving model to reconstruct the geometry. Our code and data are available at https://github.com/HalimiOshri/Pattern-Based-Cloth-Registration-and-Sparse-View-Animation. The released data includes our pattern and registered mesh sequences containing four different subjects and 15k frames in total.

CCS Concepts: • **Computing methodologies** → **Motion capture**.

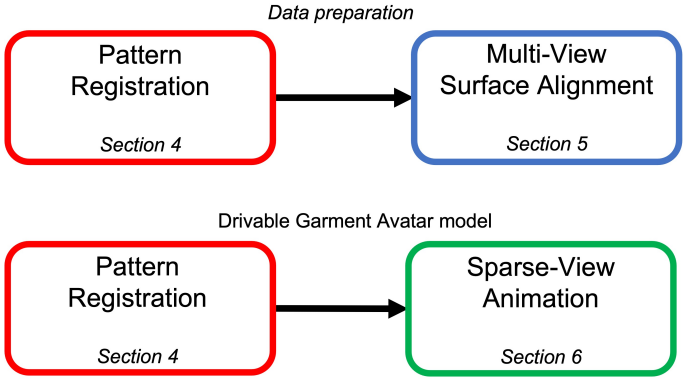
Additional Key Words and Phrases: Garment Capture, Telepresence, Virtual Clothing, Machine Learning, Computer Vision, Registration, Cloth Animation

ACM Reference Format:
Oshri Halimi, Tuur Stuyck, Donglai Xiang, Timur Bagautdinov, He Wen, Ron Kimmel, Takaaki Shiratori, Chenglei Wu, Yaser Sheikh, and Fabian Prada. 2022. Pattern-Based Cloth Registration and Sparse-View Animation. *ACM Trans. Graph.* 41, 6, Article 196 (December 2022), 17 pages. https://doi.org/10.1145/3550454.3555448

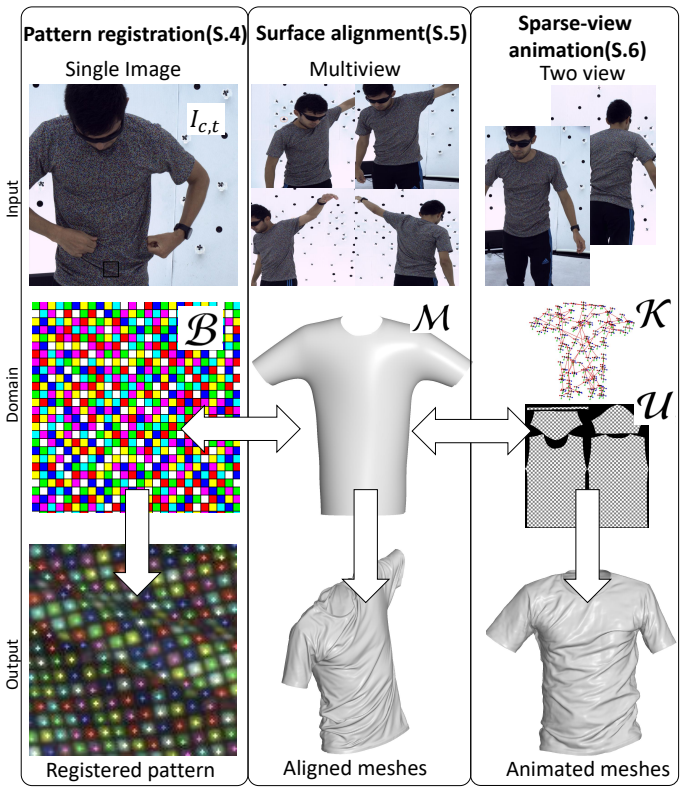
1 INTRODUCTION

We introduce a novel approach for capturing full garments in motion with high accuracy. Our cloth registration pipeline operates in a multi-view camera setting and produces a dynamic sequence of registered cloth meshes with a registration resolution of 2.67 millimeters and triangulation localization error of 1 millimeter, greatly improving upon previous work. Most notably, compared to state-of-the-art cloth registration methods [Pons-Moll et al. 2017; Xiang et al. 2021], our approach produces temporally stable texture coordinates allowing to track a specific surface point with negligible drift in the point's identity between different frames. This accurate registration is achieved using a novel pattern printed on the cloth, designed to optimize the ratio of pixel area to the number of uniquely registered cloth surface points. Dense pattern primitives are localized using a specialized image-based detector and each pattern keypoint is identified using a graph processing algorithm that robustly handles the combination of non-rigid and projective transformations, self-occlusions, and detection noise. Note that our method does not assume any human body model and treats the cloth as a generic non-rigid surface. On the other hand, ClothCap [Pons-Moll et al. 2017] is built on SMPL [Loper et al. 2015a], and [Xiang et al. 2021] uses proprietary LBS-based body model, which generally leads to decreased tracking performance when the cloth deviates significantly from the body.

Our motivation for developing an accurate cloth registration method is that many cloth-related computer vision tasks that target realistic cloth appearance could benefit from precisely registered mesh sequences. One major challenge in representing clothing comes from the lack of high-quality cloth registration of the stretching and shearing of the fabric on moving bodies, which is notoriously difficult due to the numerous self-occlusions and the lack of visual features for alignment. Previous work have made efforts in capturing simple cloth swatches under external forces in a controlled environment [Bhat et al. 2003; Clyde et al. 2017; Miguel et al. 2012; Rasheed et al. 2020; Wang et al. 2011]. However, these captures only consider isolated suspended fabrics without capturing the combined effects of friction, air drag, external forces or the interaction of garments being worn with the underlying body. While synthetic thin shell simulations have made drastic progress in recent decades [Stuyck 2018], a simulation-to-real gap remains. Our high-quality capturing of the fabric's dynamics could help bridge this gap. Applications that could benefit from the proposed registration method include enhancing cloth physical simulations [Jin et al. 2020; Runia et al. 2020], facilitating neural models for cloth dynamics [Holden et al. 2019; Lahner et al. 2018; Liang et al. 2019; Patel



(a) **Method Overview.** The data preparation process (top) reconstructs the registered cloth surface in a multi-view setting. We use the high-quality cloth registration data to train our drivable Garment Avatar model that animates the cloth, utilizing a limited number of cameras.



(b) Illustration of the three modules in the above diagram. For a given camera c, and time step t: the **Pattern Registration** module registers the visible pattern cells in a single image $I_{c,t}$; Each registration has two attributes 1) pixel coordinates in the image space 2) grid coordinates in the pattern board domain $\mathcal{B}$. The **Multi-View Surface Alignment** stage aligns a template mesh $\mathcal{M}$ according to the multi-view image registrations. The **Sparse-View Animation** module animate the cloth from the image registrations of two cameras by mapping the registrations to the UV domain, $\mathcal{U}$, and inpainting the partial signals. A crucial step in the animation fits a coarse surface to the driving camera registrations, represented by a surface kinematic model $\mathcal{K}$.

Fig. 2. Overview of the three main components in our drivable Garment Avatar model

et al. 2020; Santesteban et al. 2019, 2022, 2021] and non-rigid shape correspondence [Attaiki et al. 2021; Bracha et al. 2020; Eisenberger et al. 2020a,b; Halimi et al. 2019a; Litany et al. 2017], designing precise interfaces for cloth manipulation by robots [Bersch et al. 2011; Miller et al. 2012; Strazzeri and Torras 2021], generating synthetic data for optical-flow learning of non-rigid surfaces similar to [Butler et al. 2012; Dosovitskiy et al. 2015], VR telepresence enabling high-precision geometric modeling of garments [Bagautdinov et al. 2021; Habermann et al. 2021a], ground-truth data to generative clothing models [Bertiche et al. 2020; Ma et al. 2021a, 2020a; Saito et al. 2021], shape completion [Bednarik et al. 2020; Chi and Song 2021; Halimi et al. 2020], and interpolation [Cosmo et al. 2020; Eisenberger et al. 2019; Trappolini et al. 2021], to name just a few.

We show how the proposed cloth registration method is beneficial to telepresence applications. Due to the highly complicated non-linear motion of garments, it still remains elusive to teleport the clothing faithfully on the moving body of an avatar. State-of-the-art cloth driving methods, i.e., predicting the cloth's state from a signal containing partial information, model the garment with a separate mesh layer [Xiang et al. 2021] and apply a registration stage when training those garment models. However, ground truth data of registered clothes, describing the complex dynamics, such as the bending, stretching and shearing at a fine-grained level is still unavailable at high resolution and accuracy, limiting the quality of drivable garment models. In this context, we tried to answer two questions. The first is: do existing cloth-driving methods using the body pose as the driving signal improve using our registration method in the training stage and, as a result, produce more plausible geometry at inference time when driven from the body pose. Our observations confirms this statement. We show in Section 7.2.1 that the proposed novel cloth registration method significantly boosts the performance of state-of-the-art pose-driven animatable garment models.

The second explored question is: can the dense-tracking signal, obtained in our registration pipeline per camera view, serve as a novel driving signal. Specifically, we use the surface points' pixel coordinates, which we track per camera using our pattern, as described in Section 4. This newly explored driving signal is interesting as an alternative to the pose signal, which generally serves for driving and comes with limitations. In practice, pose-driven models tend to either significantly smooth out details or introduce high-frequency deformations which are not faithful to the actual underlying clothing state. On the other end, a dense tracking signal from a sparse set of views has much more spatial correlations with the garment embedding, and therefore, we expect better fidelity.

To this end, we use the tracking signal obtained from two opposing camera views, selected to provide wide optical coverage. This is a challenging setting that stereo-vision-based approaches are unable to reconstruct. Every point on the cloth surface is visible in one view at most and therefore cannot be directly triangulated. In our approach, we design a UNet-type network that receives multiple camera channels; each contains a partial pixel coordinate signal in UV space, as described in Section 6.1. The network predicts the inpainted 3D coordinates in the same UV domain. To allow generalization, we normalize both inputs and outputs of the model with respect to an estimation of the coarse surface, described in Section 6.2. We produce this coarse geometry using a surface-based kinematic model that fits the partial observations. Importantly, our kinematic model is agnostic to the wearer's body state and ultimately can be applied to an arbitrary type of garment. We achieve this by modeling the cloth deformation using a hierarchical deformation graph with independent degrees of freedom, constructed automatically for a given triangulated surface. Our experiments, in Section 7.2.2, demonstrate that our novel driving model delivers significantly more realistic reconstructions than state-of-the-art pose-driving baselines.

Our motivation to introduce a proof of concept for this novel driving mechanism is twofold. Firstly, this driving paradigm can serve applications that target the realistic and faithful geometry driven by the patterned cloth. For example, in the early stages of cloth teleportation, patterned clothes could be distributed that allow the user to pick and swap garment appearance on-the-fly, supporting even dynamic textures, like Narita et al. [2016]. The second motivation is to provide an essential baseline for the performance of cloth driving from a dense tracking signal, for a tracking signal of ground-truth quality. In summary, our main contributions are:

- We develop a carefully designed cloth registration pipeline that captures cloth at high accuracy with dense correspondences.
- We develop a method for accurate and realistic cloth animation from pixel registration obtained from two cameras video streams.
- We demonstrate that the training data obtained from our registration method can significantly improve the output quality when applied to pose-driven animation.
- We release a dataset of captured cloth motion from 4 different subjects and 15k frames in total.

2 RELATED WORK

2.1 Multi-View Clothing Capture

Multi-view clothing capture has been explored as a source of geometry for garment modeling. A typical multi-view clothing capture pipeline consists of three steps, geometry reconstruction, clothing region segmentation, and registration.

In the geometry reconstruction step, the raw surface of the clothed human body is typically reconstructed from multi-view RGB input images using Multi-View Stereo (MVS) or from 3D scanners using Photometric Stereo. In the segmentation step, the garment region is identified and segmented out. Early work [Bradley et al. 2008; Pons-Moll et al. 2017] uses the difference of color between the garment and the skin as the primary source of information for segmentation, while more recent work [Bhatnagar et al. 2019; Xiang et al. 2021] aggregates clothing parsing results from multi-view RGB images to perform the segmentation more robustly. Bang et al. [2021] introduce boundary-based segmentation scheme to further improve segmentation accuracy.

Compared with the reconstruction and segmentation steps, the registration of garment is a more open question and also the focus of this work. The goal of registration is to represent the complete

geometry of the clothing in different frames with a fixed mesh topology and encode the correspondences by vertices. Previous literature of clothing registration falls into the following two categories: **correspondence-free** methods and **correspondence-based** methods.

Correspondence-free methods start from a pre-defined template topology and fit the template to each instance of the captured garment according to the geometry. Early work [Bradley et al. 2008] attempts to find consistent cross-parameterization among different frames of the garment from a common base mesh by minimizing the stretching distortion, and then explicitly represent the correspondences by re-triangulation.

The majority of work in this category [Bhatnagar et al. 2019; Ma et al. 2020b; Pons-Moll et al. 2017; Tiwari et al. 2020; Xiang et al. 2021, 2020; Zhang et al. 2017] uses non-rigid Iterative Closest Point (ICP) to fit a template mesh to the target clothing geometry. The problem is formulated as a minimization of surface distance between the free-form template and the target, with an additional regularization term that preserves the quality of mesh triangulation. To provide better initialization for the optimization, some work [Ma et al. 2020b; Xiang et al. 2020; Zhang et al. 2017] first uses a kinematic model such as SMPL [Loper et al. 2015b] to help estimate a coarse human surface, which is further aligned with the reconstructed clothing shape by allowing free-form deformation in the SMPL body topology. Some recent work [Bhatnagar et al. 2020a,b] replaces the explicit optimization with the prediction from a neural network to avoid the difficulty of robustly initializing and regularizing the optimization. The above SMPL+D formulation assumes a one-to-one fixed correspondence between the body template and the clothing, which is often violated due to tangential relative movement between the fabric and the body, as well as the invisible clothing regions in the wrinkle folds.

Therefore, some work [Bhatnagar et al. 2019; Pons-Moll et al. 2017; Tiwari et al. 2020; Xiang et al. 2021] further separate the representation of the clothing from the underlying body, and use the segmented boundary to guide the deformation. While these methods can produce visually appealing registered clothing sequences, they suffer from the fundamental limitation of inferring correspondences purely from geometry. There are no explicit clues for the correspondences between frames except the regularization of mesh triangulation or vertex distances. Therefore, the registration output generally suffer from correspondence errors since there is no mechanism to ensure that each vertex coherently tracks the same physical point.

By comparison, **Correspondence-based** methods, to which our approach belongs, do not suffer from the ambiguity in correspondences as the **correspondence-free** methods and often take advantage of visual cues by using a designed pattern. The key idea is to use identifiable patterns to explicitly encode correspondences on the captured surface. Similar concepts have been widely explored in the use of checker boards for camera calibration [Dao and Sugimoto 2010], and the application of fiducial markers, like ARTag [Fiala 2005], AprilTag [Olson 2011; Wang and Olson 2016] and ArUco [Garrido-Jurado et al. 2014].

Specific to the area of garment capture, early work [Pritchard and Heidrich 2003; Scholz et al. 2005; White et al. 2007] utilizes classical computer vision techniques such as corner detection and multi-view geometry to reconstruct and identify printed markers on the garments. With the help of the pattern, the correspondences on the garments can be robustly tracked in visible sections and reliably estimated in regions occluded by folds and wrinkles. Our work extends the color-coded pattern approach [Scholz et al. 2005] to achieve denser detection.

In recent years, the constantly evolving frontier of learning-based computer vision algorithms enables revisiting this research problem with a plethora of enhanced image processing capabilities. The focus has shifted to the pattern design question and its resulting information theory properties. Specifically, to allow high-resolution capture, one should design the pattern to detect as many as possible points per surface area. For example, Yaldiz and colleagues [2021] generate learnable fiducial markers optimized for robust detection under surface deformations, using a differentiable renderer for end-to-end training. To make the marker detection resilient to surface deformations, they use geometric augmentations such as radial and perspective distortions and TPS (thin-plate-spline) deformation. Those kinds of deformation are applied in the 2D image space and cannot simulate self-occlusions resulting from the 3D folding of the surface. Currently, fiducial-markers-based tracking methods cope with relatively mild deformations and still do not address complex deformations containing folds and wrinkles. Chong et al. [2021] uses a colored patterned cloth with an actuated mannequin to collect loose ground truth correspondences and supervise an image translation network, but the method is not capable of faithfully reproducing complete geometry of the garment.

In the clothing registration problem, the pursuit of high resolution makes characters and symbols unsuitable for the printed pattern due to their low density of correspondences. Thus Chen and colleagues [2021] propose to identify a corner from its immediate surrounding squares printed on a tight suit worn by the subjects. In our setting, each center might cover no more than a few pixels so we must rely on simpler attributes that can be robustly detected from the low resolution observation of the board squares in the images.

2.2 Clothing Animation

Pose-driven clothing animation aims to produce realistic clothing animation from the input pose represented by 3D joint angles, or the underlying body skinned according to the joint angles by a kinematic model. *Physics-based simulation* of garments [Baraff and Witkin 1998; Narain et al. 2012; Stuyck 2018] has been studied for a long time and is an established approach for clothing animation in the movie and gaming industry. In recent years, there has been a lot of interest in using *data-driven* approaches, especially deep neural networks, to directly learn clothing animation from data paired with the input body poses [Bertiche et al. 2021; Habermann et al. 2021a; Lahner et al. 2018; Ma et al. 2021b, 2020b, 2021c; Saito et al. 2021; Santesteban et al. 2021; Xiang et al. 2021]. While these approaches can produce visually appealing animation, the input body motion alone does not contain enough information to guarantee the consistency between the animation output and the real clothing status of the teleported subject, as these pose-driven approaches do not utilize any visual cues of the current appearance.

Our binocular clothing animation setting is also related to the recent work in **performance capture** from monocular or sparse multi-view inputs, which can also serve the purpose of clothing animation for VR telepresence. This line of work can be further divided into two categories: *shape regression* approaches and *template deformation approaches*. The *shape regression* approaches train deep neural networks to regress per-frame clothed human shape from monocular [Alldieck et al. 2019; Li et al. 2020; Natsume et al. 2019; Saito et al. 2019, 2020] or sparse multi-view inputs [Bhatnagar et al. 2019; Huang et al. 2018]. These approaches enjoy the flexibility of being able to address different subjects and clothing with a single network, but are usually limited in output quality due to the fundamental ambiguity of inferring complicated clothing geometry without prior knowledge about the specific subject. More related to our work are the *template deformation* approaches. These approaches utilize a pre-scanned personalized template of a specific subject to track the clothing deformation from a single RGB video [Habermann et al. 2019, 2021b, 2020; Li et al. 2021; Xu et al. 2018]. The templates may also be built on-the-fly by fusing different frames of geometry from RGB-D input [Su et al. 2020; Yu et al. 2021, 2018]. The personalized templates can provide strong prior knowledge to alleviate the 3D garment shape ambiguity. In our work, patterned cloth serves as a special template whose correspondences can be easily inferred from the input driving signal, thus amenable to high-quality clothing teleportation.

3 OUTLINE

The core contribution of this work is a high-quality reconstruction and registration of the cloth surface in 3D. To register the surface precisely in 3D, we devised a unique pattern that allows precise tracking and registration of surface points in the image domain, allowing their later triangulation from multiple views to create the data term for the surface alignment. We also explored our registered pattern in the image domain as a dense driving signal to generate faithful and realistic cloth animations. Figure 2a displays the global relations between the different modules presented in this work, depicting the registration data generation and cloth driving stages. Figure 2b includes an overview for each of the three modules that appear in the general diagram.

The first stage is **Pattern Registration**, described in Section 4, see Figure 3, is the shared component between both model building and model animation. Given a single frame capturing a performer wearing a grid-like patterned garment, we register every visible pattern cell, with predefined grid coordinates in the pattern domain $\mathcal{B}$, to pixel coordinates in the image space.

In the *data preparation* branch, we use the registered frames obtained from a multi-view camera system as an input to our **Multi-View Surface Alignment** stage, which we describe in Section 5 and visualize in Figure 8. First, we triangulate all the registered frames to get registered point clouds where every point maps to specific coordinates in the pattern domain $\mathcal{B}$. Then, we align a mesh template M to the resulting point clouds.

In the *Garment Avatar* model branch, we use a sparse set of registered frames as the driving signal to our **Sparse-View Animation** module, described in Section 6, see Figure 9. The animation module consists of our *Pixel Driving* network and a coarse geometry estimation procedure. Specifically, the *Pixel Driving* receives the pixel location signal detected per camera, as defined in the UV space, and outputs the world space 3D coordinates signal in the same UV domain. Crucially, to ensure generalization to newly seen poses at inference time, we define the inputs and outputs to our network relative to a coarse geometry fitted to our sparse-view pixel registrations.

4 PATTERNED REGISTRATION

To enable accurate cloth captures, we manufacture a piece of cloth with a fine-grained color-coded pattern. We follow a similar approach to pattern design as described by Scholz et al. [2005]. We introduce novel methods for robust registration of color-coded patterns that enables dense alignment of cloth.

4.1 Pattern Design

The pattern consist of a colored board where cells takes one of seven colors. Cells are separated by grid lines to improve contrast at edges and corners of the board. We assign colors to cells in such a way that the color configuration on each 3×3 cell-set is unique, including w.r.t. board rotations. We further impose adjacent cells to have different colors to improve color disambiguation.

We print a color board with 300×900 cells on polyester fabric with a resolution of 2.7mm per cell. A t-shirt is manufactured by sewing the cut garment panels extracted from the fabric, see Figure 4. Our manufactured t-shirt model contains 98618 cells.

Our choice of primitive colors and cell configuration balances locality, distinctiveness and uniqueness. A 3×3 cell-set provides us with a good match between locality and uniqueness: for a cell of 2.7mm length, we can uniquely localize a pattern element from any visible 8mm×8mm patch containing it. Since we have multiple (up to nine) candidate patches that allow such a pattern element to be localized, our method could take advantage of this duplicated information. It uses this duplicated information in two ways: 1) to localize border pattern elements visible in the image that do not have a visible 3x3 patch centered around them, thus achieving enhanced coverage of localized regions in each image frame 2) For handling error correction and resolving localization ambiguity in the presence of detection noise, as described in Section 4.4. Getting unique detections for cell-set smaller than 3×3 can be achieved by increasing the number of color primitives, however this reduce the distinctiveness of the color primitives, increasing quantization errors, making the pattern registration less robust.

4.2 Image Detection

We propose an image pattern detector, *PatterNet*, consisting of two separate networks. *SquareLatticeNet* detects the corners and the centers. *ColorBitNet* classifies the pixel color. Both networks have been implemented using a UNet. The location of the square center can reveal valuable information about the topology; however, under a general affine transformation, it leads to a non-unique graph, as shown in Figure 5, due to the lattice symmetry. To restrict ambiguity, we additionally detect the square corners. We empirically observed

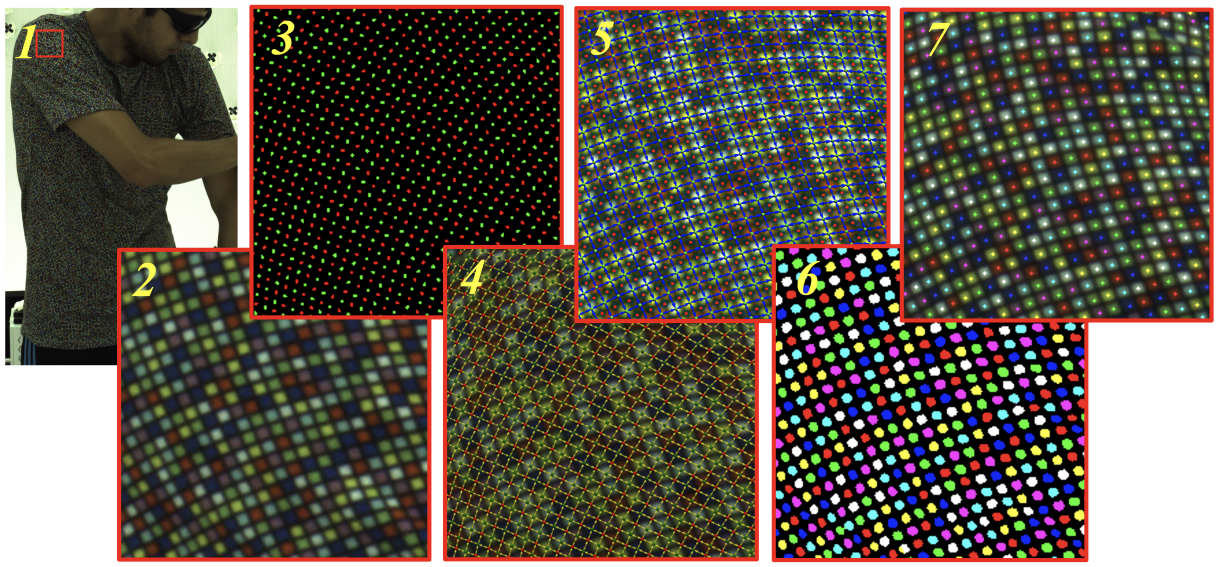


Fig. 3. **Patterned Cloth Registration.** 1) Original frame 2) Zooming in on the pattern 3) keypoint detection by *SquareLatticeNet* - corners (red) and centers (green) 4) heterogeneous graph 5) homogeneous graph 6) color detection by *ColorBitNet* - pixel classification 7) Registered pixels: every pixel detection is assigned with coordinates in the pattern domain $\mathcal{B}$.

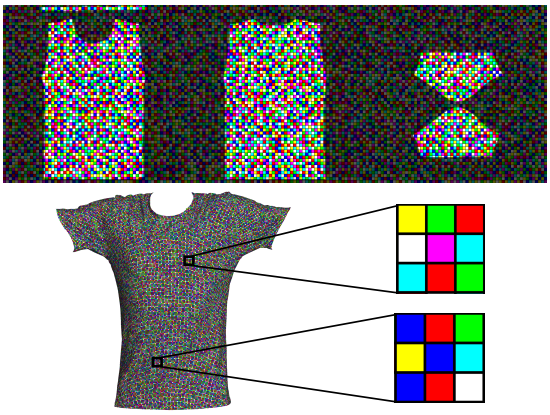


Fig. 4. **Pattern design and fabrication**. The colored board is printed on fabric (top) to manufacture a t-shirt (bottom left). Each 3×3 cell of the pattern (bottom right) has unique color configuration.

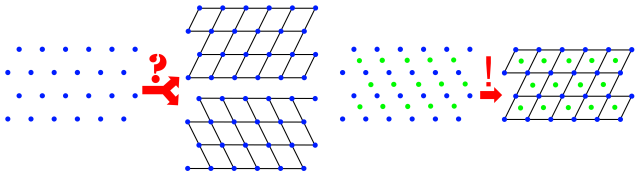


Fig. 5. Lattice neighborhood inference with squares and corners detection. The center (blue) topology is ambiguous without the corners' cue (green). Introducing the corners eliminates the incorrect topology out of the two.

that in our setting, the combination of centers and corners is sufficient for our algorithm to determine the correct local neighborhood of each visible square in the image in most cases. Graph ambiguities are usually caused by potential high-skew transformations that create aliasing, and by adding the corners, we make the aliasing occur less frequently. We refer the reader to the supplementary material for a detailed review of the image pattern detector.

4.3 Graph Processing

This stage reconstructs the graph topology of the square grid from the center and corner detections obtained by the previous step. The detection contains a certain amount of noise, such as outliers, and parts of the graph are not visible in the image due to self-occlusion. Additionally, near the invisible graph parts, it's common to see discontinuities in the periodic structure of the detected lattice points, where nodes disconnected in the graph become proximate in the image. Our approach is to recover the heterogeneous graph defined on the square grid by connecting each center to its surrounding corners and vice versa. Then, we construct the homogeneous grid-graph connecting the centers from the former. Please see the supplementary material for a detailed description of both graph generation algorithms. Our approach has the following advantages:

(1) We have precise geometric requirements between opposite type neighbor, namely corners and centers, derived from the locally affine approximation that we can utilize to filter outlier detection and outlier graph edges.
(2) Neighbors in the heterogeneous graph fall closer in the image than neighbors in the homogeneous graph. As a result, we can apply geometric regularizations between neighbors by neglecting the non-rigid deformation and assuming a locally affine transformation.

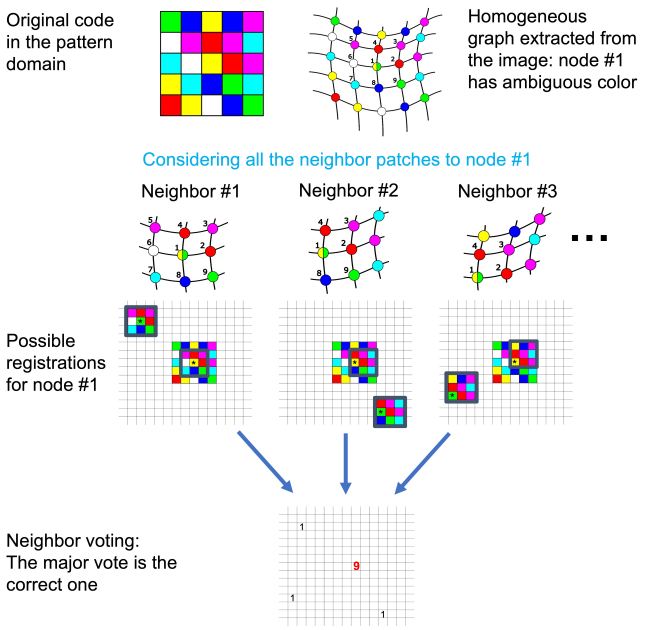


Fig. 6. **Neighbor voting algorithm illustration:** a single color-bit ambiguity is resolved by the majority vote.

(3) Once the centers are registered, the heterogeneous graph enables the registration of the corners and the later triangulation, thus doubling the resolution of the registered cloth surface. We didn't exploit the last advantage since corner registration introduces additional steps into our registration pipeline and template generation, which we defer to future work.

4.4 Hash Code Inference

This stage aims to calculate the hash code for every node in the homogeneous graph and to register the node to a board location using this code. After the previous stage, we have a 3×3 grid-graph attribute around the center of each node in the homogeneous graph. Each node has a color attribute that is either a single color for confident detections or several candidate colors for ambiguous results. We are robust against possible color ambiguities and additionally offer a way to reject codes where the color is not ambiguous but the detection is wrong. We propose the neighbor-voting-based hash extraction algorithm that enables information recovery given incorrect detections. Most code errors produce an empty result in the hash query. However, correct codes always have a unique match on the board, and neighborhood relations are preserved from the homogeneous graph to the patches in the board domain, whereas this is not the case for incorrect codes. With this insight, we suggest the *neighbor voting* algorithm to help recover ambiguous codes and to reject faulty but unambiguous codes. The algorithm is described in the supplementary and visualized in Figure 6.

5 MULTI-VIEW SURFACE ALIGNMENT

To build realistic priors of garment deformation, we capture precise cloth motion in a multi-view studio and align a template mesh representation of the the garment to the pattern registrations.

5.1 Setup

The capture studio consists of 211 calibrated cameras with varying focal lengths that provide distinct levels of detail of our pattern.

For our capture, a subject wearing the patterned cloth is located at the center of stage and asked to perform several types of actions. We capture natural cloth deformation produced by deep breathing, realistic wrinkle formation derived by torso motion, and challenging cloth deformation produced by hand-cloth interactions.

5.2 Pattern Triangulation

The first step in the panoptic registration is triangulating the multi-view pixel registrations from all the camera views to get the registered point cloud. Each 3D point in the resulting point cloud maps to the pattern square grid coordinates. This step functions as an independent filtering stage, filtering any remaining outlier pixel registrations from the pattern registration pipeline. We perform triangulation using the RANSAC algorithm, considering only intersections within 1 millimeter radius of at least 3 different rays, as a valid triangulation. The details of the triangulation algorithm are covered in the supplementary. We illustrate the surface coverage provided by the triangulation of the registered pattern in Figure 7.

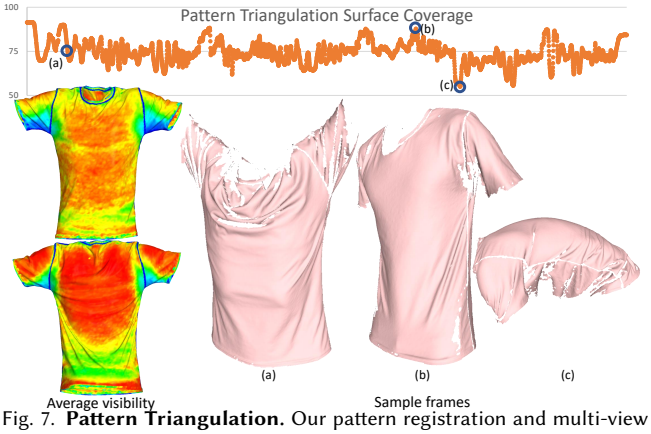


Fig. 7. **Pattern Triangulation.** Our pattern registration and multi-view system, enables a dense registered triangulation. We achieved between 54% and 90% of surface coverage per frame, with average of 75% (see top plot). We show representative frames attaining mean(a),max(b), and min(c) coverage. We also provide a heatmap illustrating average per vertex visibility.

5.3 Surface Template Model

A template mesh $\mathcal{M}$ is constructed to model the state of the full garment surface. Our template mesh enables intuitive mapping between the pattern board, the garment surface, and the UV layout.

The 2D garment pattern on the printed fabric (top row of Figure 4) provides an initial estimate of the pattern cells that are visible on the shirt. However, some of these cells get occluded after sewing,

requiring further refinement based on multiview pattern registration. We construct an initial planar mesh by triangulating these active cells in such a way that cell centers are covered by mesh vertices. This produces a mesh with 5 connected components that have a 1-1 correspondence from vertices to visible pattern cells. To produce a fully connected mesh, we lift our planar components to fit pattern triangulations, and manually extend the mesh tessellation to close seam gaps in regions of good visibility. Our zippered template mesh contains 101286 vertices: 98618 corresponding to visible cell centers and 2668 introduced for seam closure. We define the initial embedding of $\mathcal{M}$ by fitting the tessellation to a T-pose frame and running ARAP[Sorkine and Alexa 2007] refinement to closely preserve isometry to the planar mesh.

We produce a UV parameterization for the zippered model by extending the initial planar mesh to incorporate the triangles introduced on the seam closure. We use the work by Sawhney et al. [2017] to compute a flat extension for the new triangles. The final UV layout is obtained by arranging the charts to compactly fit in a square.

5.4 Surface Alignment

5.4.1 Deformation Parameterization. To align our surface template $\mathcal{M}$ to each frame of the multi-view capture, we use a dense parametric model of surface deformation. The deformation model associates each vertex $v_k \in \mathcal{M}$ with a fixed coordinate frame F_k centered at v_k, and a local rigid transformation ϕ_k. For a given local transformation ϕ_k, we obtain a global transformation $\tilde{\phi}_k := F_k \circ \phi_k \circ F_k^{-1}$, such that $\tilde{\phi}_k(v_k)$ corresponds to the deformed vertex position in world coordinates. Given a vector of local transformations $\boldsymbol{\phi} := \{\phi_k\}$, we denote the fully deformed mesh as $\mathcal{S}(\boldsymbol{\phi})$. In particular, $\mathcal{S}(\boldsymbol{I}) = \mathcal{M}$, where $\boldsymbol{I}$ is a vector of identity transformations.

5.4.2 Incremental Optimization. We run registration optimization in several stages: first, we optimize each frame independently, from the rest state $\mathcal{M}$ to a deformation state $\mathcal{S}$. Then, we run refinement passes to update $\mathcal{S}$ using the previous pass state (i.e., a Jacobi style update). This enables full parallelism on the capture processing. We empirically observed that this incremental approach provides a satisfactory trade-off between performance and fitting quality compared with direct optimization from the rest state.

Initialization. : The first step of the optimization is to optimize $\boldsymbol{\phi}$ such that $\mathcal{S}(\boldsymbol{\phi})$ matches well to vertices with valid triangulated positions. We first construct a mesh $\mathcal{P}$ from triangulated vertices in the cloth pattern. We construct $\mathcal{P}$ to have identical topology as $\mathcal{M}$. The detected vertices in $\mathcal{P}$ are assigned to triangulated positions, while the positions of the remaining non-detected vertices are set via Laplacian filling [Meyer et al. 2003].

For each vertex $v_k \in \mathcal{M}$ we initialize ϕ_k according to the as-rigid-as-possible transformation [Sorkine and Alexa 2007] mapping the neighborhood of v_k in $\mathcal{M}$ to the respective neighborhood in $\mathcal{P}$.

Then, we refine $\boldsymbol{\phi}$ by jointly minimizing the detection and distortion terms, namely E_{Det} and E_{Dist}. E_{Det} is the data term with both triangulated vertex positions and Laplacian-filled vertices, and is formulated as

$$E_{\text{Det}}(\boldsymbol{\phi}) := \|\mathcal{P} - \mathcal{S}(\boldsymbol{\phi})\|^2. \tag{1}$$

E_{Dist} is the regularization term measuring mesh distortion. Inspired by Sumner et al. [2007], we measure local distortion by comparing the action of transformations defined at adjacent vertices of the mesh. Given edge $e_{kl} \in \mathcal{M}$, we denote by $\{s\}_{kl}$ a set of distinctive points associated to edge e_{kl} (in practice we use the midpoint, and edge corners). We compute the distortion error as,

$$E_{\text{Dist}}(\boldsymbol{\phi}) := \sum_{e_{kl} \in \mathcal{M}} \|\tilde{\phi_k}(\{s\}_{kl}) - \tilde{\phi_l}(\{s\}_{kl})\|^2. \tag{2}$$

This produces an initial deformed mesh $\mathcal{S}_0$ that closely approximates $\mathcal{P}$ and further corrects triangulation outliers due to E_{Dist}. Due to the density of our pattern registration, we noticed that a Laplacian filling was sufficient to provide a reasonable initialization on unobserved regions as illustrated in Figure 8. A bi-Laplacian filling would provide a smoother filling but we would obtain similar mesh deformation once the distortion energy in Equation 2 is active.

After this initialization step, we freeze all the valid detected vertices, and only optimize for the non-detected ones in all the subsequent stages. In Figure 8, the vertices in the red regions are frozen by the optimization and we only optimize for the deformation parameters of the gray regions. This allows to accelerate the remaining optimization steps due to the sparsity of non-detected vertices.

Optimizing Non-Detected Vertices. : As non-detected vertices are likely occluded, we introduce the following prior: non-detected vertices are more likely to belong to occluded-concave sections (e.g., occluded sections in wrinkles or armpits) than to visible-concave ones. We consider this prior in optimization by formulating the *pushing constraint* on the non-detected vertices, as

$$E_{\text{Push}}(\boldsymbol{\phi}) := \|(\mathcal{S}_0 - \epsilon N(\mathcal{S}_0) - \mathcal{S}(\boldsymbol{\phi})\|^2. \tag{3}$$

This pushing constraint regularizes each non-detected vertex to move in the opposite direction of its normal N computed from S_0 with a magnitude ϵ. We also add an additional data term E_{Recon} with 3D scan mesh $\mathcal{R}$ computed by Yu et al.'s method [2021], as some of non-detected vertices could be registered to multi-view stereo. E_{Recon} is formulated as

$$E_{\text{Recon}}(\boldsymbol{\phi}) := \|\text{ICP}(\mathcal{S}; \mathcal{R}) - \mathcal{S}(\boldsymbol{\phi})\|^2, \tag{4}$$

where $\text{ICP}(\mathcal{S}; \mathcal{R})$ is the (reduced) iterative closest point association from $\mathcal{S}$ to $\mathcal{R}$. More precisely, for each vertex in $\mathcal{R}$ we compute the closest in $\mathcal{S}$, and define a partial correspondence map from $\mathcal{S}$ to $\mathcal{R}$ by applying reduction.

Solving Self-Intersections. : The above optimization steps do not guarantee intersection-free meshes. To alleviate mesh collisions, we employ a physics-based correction module that locally corrects intersecting regions by running a cloth simulation [Stuyck 2018] while leaving non-intersecting regions unmodified. We first detect intersecting regions using a surface area heuristic [Baraff et al. 2003], and then run cloth simulation where we apply corrective forces to these regions to resolve self-intersection. We observed that first applying the pushing prior (Equation 3) and the reconstruction term (Equation 4), provided a better initialization for the quasi-static simulation than applying it directly from the Laplacian filling initialization. Figure 8 illustrates the mesh state at different stages of the optimization.

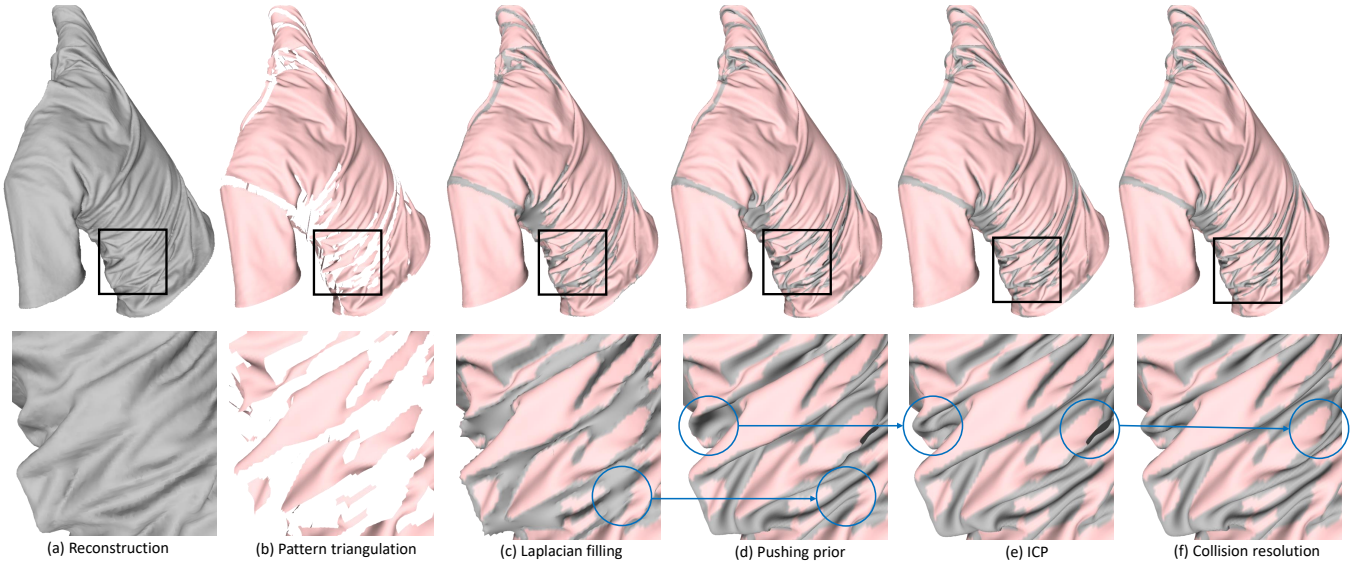


Fig. 8. **Multi-view Registration** We implemented an incremental per-frame registration approach to fit multi-view reconstructions and pattern triangulations. Our deformation state is initialized from the triangulated vertices (detected) and Laplacian filling (non detected). We use a pulling prior to deform non detected regions towards occluded-concave sections, and ICP to cover unregistered reconstruction surface. Finally we run physics based correction module to resolve collisions. We highlight regions of improvement after each step of our incremental method.

Temporal refinement. : The final step of cloth surface alignment is to make meshes temporally smooth, thus optimizes non-detected vertices via E_{Recon}, E_{Dist} and the smoothness constraint E_{Smooth} formulated as

$$E_{\text{Smooth}}(\boldsymbol{\phi}_t) := \|\mathcal{S}(\boldsymbol{\phi}_t) - \frac{1}{4}(\mathcal{S}'_{t-1} + 2\mathcal{S}'_t + \mathcal{S}'_{t+1})\|^2, \tag{5}$$

where $\mathcal{S}'$ is a mesh state after the step of solving self-intersection and t is a frame index. We observed that this smoothness constraint could avoid oscillatory states that we saw with a typical smoothness constraint formulated as $\|\mathcal{S}(\boldsymbol{\phi}_t) - \frac{1}{2}(\mathcal{S}'_{t-1} + \mathcal{S}'_{t+1})\|^2$. Note that this final optimization can run in parallel because E_{Smooth} depends on $\mathcal{S}'$ instead of $\mathcal{S}$.

We update deformation parameters at each optimization stage by computing analytical Jacobians and solving using the Gauss-Newton method. The full optimization, from initialization to temporal refinement, takes 10 minutes per frame on a single CPU core. We process frames concurrently in a multi-CPU cluster. We show the final result of surface alignment on a few frames of the capture in Figures 12 and 13. We refer to the supplementary video for further qualitative evaluation of the multi-view aligned meshes.

6 SPARSE VIEW ANIMATION

This section introduces Garment Avatar, our data-driven model capable of producing highly realistic cloth reconstructions from sparse image views of the patterned garment. We provide the overview of the model in Figure 9.

6.1 Pixel Registration and Driving

Our approach takes as input two roughly 180° opposing camera views capturing the patterned garment. Each driving image is first processed with our pattern registration module to obtain pixel registration in the UV domain.

We then approach reconstruction task as a particular case of inpainting, since fully inpainted pixel coordinates are mutually interchangeable with the reconstructed 3D coordinates in the UV domain via triangulation and projection. Unlike classical inpainting, where all the channels are either available or not, in our setup input signals are available in non-overlapping regions of the UV domain. In practice, the union of signals' support across cameras leaves a significant part of the cloth region uncovered, making it even more challenging. To this end, we stack the pixel coordinates signals of both camera channels into a tensor containing the joint camera observations $P_t \in \mathbb{R}^{U\times V\times 2C}$, where $C = 2$ is the number of cameras. To ensure generalization to unseen poses, pixel coordinates are normalized with respect to our coarse kinematic model, described below. Whenever a pixel coordinate is not available, we assigned zero value to the corresponding UV normalized coordinate. We then pass P_t as an input to the pixel driving network - UNet-based architecture, which has shown its effectiveness for inpainting and image translation problems [Isola et al. 2017; Yan et al. 2018]. The pixel driving network predicts 3D coordinates in UV-space $R_t \in \mathbb{R}^{U\times V\times 3}$. Finally, we apply the network output as displacements to the kinematic model's coarse geometry, after applying augmentation, as described in Section 6.3.

6.2 Coarse Geometry Approximation

We obtain a coarse approximation for the animated cloth by fitting a kinematic model, automatically constructed from template $\mathcal{M}$, to the pixel projection seen in each driving camera. This way, our network maps the offset signal in pixel coordinates to the offset

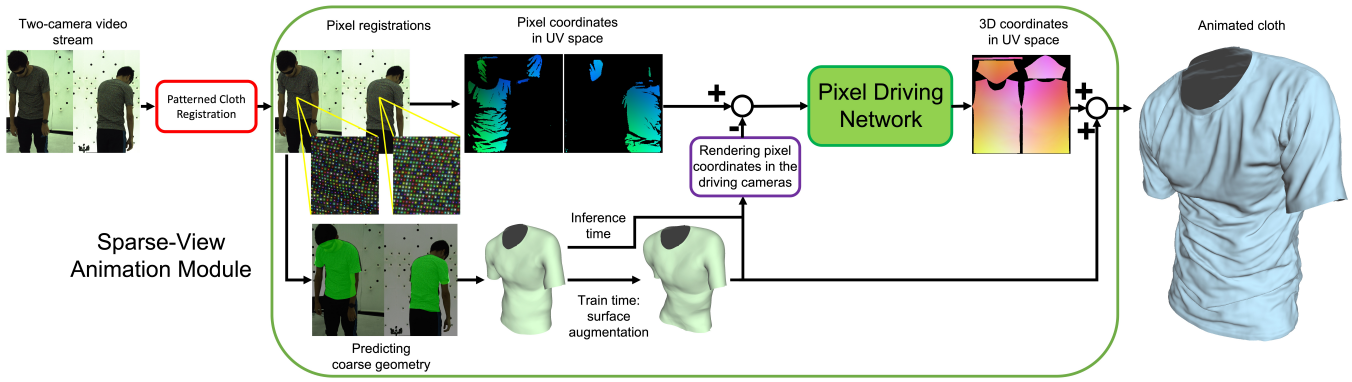


Fig. 9. **Garment Avatar Model**. Our model receives a two-camera video stream as an input and generates the animated cloth. The first stage is pattern registration of the input frames, producing pixel registrations for each camera. Then, we use the pixel registrations as input to our **Sparse-View Animation** module. First, the pixel registrations are mapped from the image space to the UV space, serving as an input branch to our Pixel Driving network. Additionally, we predict the coarse geometry that aligns with the pixel registrations, generating a reference signal of the pixel coordinates in the UV space of the coarse geometry as rendered in each camera. To get the input to the network, we subtract the reference signal from the primary signal. To get the final cloth reconstruction, we add the 3D coordinates predicted by the Pixel Driving network as an offset on top of the coarse geometry's 3D coordinates.

signal in world coordinates. With this normalization, the input and the output signal distribute roughly like Gaussian random variables.

6.2.1 Kinematic Model. A common approach to construct kinematic models for garments is by restricting to certain components of the full body models. For instance, kinematic models for shirts are commonly generated by restricting to torso and upper arms. This provides a good prior for cloth deformation based on body kinematics, but it is only valid for tight cloth deformations. Given the motion complexity we aim to model, and the dense detection provided by our pattern, we created a kinematic model with additional flexibility. Our kinematic model $\mathcal{K}$, shown at top left of Figure 10 is a hierarchical deformation graph (i.e., a skeleton) with 156 nodes. We use a procedural approach for its construction that can be applied to other types of garments. Please refer to the supplemental material for details on its construction.

The kinematic model is animated as in standard skeletons [Loper et al. 2015b]: local rigid transformations $\boldsymbol{\theta} = \{\theta_i\}$ defined at the nodes of the skeleton, are propagated through the kinematic chain, and vertex positions are computed by linear blend skinning. By abuse of notation we denote the skinned mesh by $\mathcal{K}(\boldsymbol{\theta})$.

6.2.2 Generative Coarse Deformation Model. To get realistic cloth deformation, we learn a generative parameter model for $\mathcal{K}$ from the training set. First, we compute deformation parameters $\boldsymbol{\theta}_t$ for each multi-view aligned surface $\mathcal{S}_t$ in the training set by minimizing vertex and distortion error:

$$\boldsymbol{\theta}_t := \operatorname{argmin} ||\mathcal{S}_t - \mathcal{K}(\boldsymbol{\theta})||^2 + \lambda E_{\text{Dist}}(\boldsymbol{\theta}). \tag{6}$$

Here we use the same distortion as in Equation 2, defining adjacency of skeleton nodes by overlapping skin support.

The set of deformation parameters $\Theta = \{\boldsymbol{\theta}_t\}$ characterize the space of realistic cloth deformations given by our training set. We normalize these deformation parameters by removing the root transformation. A variational autoencoder-based generative network is trained on this normalized parameters, similar to [Pavlakos et al. 2019]. Although we supervise the reconstruction error in the deformed mesh space rather than in the parameter space.

We denote the generative parameter model by $\mathcal{G}$. It is designed as an MLP with two hidden layers mapping a 32-dimensional latent code to the 1085 parameter space (rigid transformation of all nodes but the root) of the kinematic model.

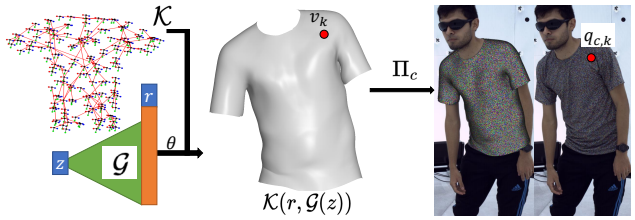


Fig. 10. **Coarse Deformation Fitting**. A generative pose model $\mathcal{G}$ is trained from the multi-view aligned meshes to represent the space of coarse cloth deformations on skeleton $\mathcal{K}$. We use this deformation prior to robustly estimate the coarse state of our cloth from sparse view points.

6.2.3 Coarse Deformation Fitting. Given a vertex $v_k \in \mathcal{M}$ and camera $c \in C$, we denote by $w_{c,k}$ to the binary value specifying the vertex visibility, and $q_{c,k}$ the pixel coordinate of its corresponding detection (if any). We denote Π_c the camera projective operator. Please refer to Figure 10 for further illustration on the notation.

We compute the coarse cloth deformation by optimizing for root transform r^* and latent code z^*, to fit image detections,

$$r^*, z^* = \operatorname{argmin} \sum_{c \in C} \sum_{k \in \mathcal{M}} w_{c,k} ||\Pi_c(\mathcal{K}(r, \mathcal{G}(z))(v_k)) - q_{c,k}||^2. \tag{7}$$

We use $\mathcal{K}(r^*, \mathcal{G}(z^*))$ for signal normalization in the Pixel Driving Network.

6.3 Training and Inference Details

We train our Pixel Driving network with L2 loss with respect to the ground-truth registered meshes. We apply the L2 loss between the 3D vertex positions and include an L2 between the surface normal vectors to promote the preservation of fine geometric detail. Our training data consists of the first 3500 frames of the sequence, while the remaining 500 frames are used as the test set. The train pose distribution is diverse, including torso motion, jogging, and hand-cloth interactions.

We found that in coarse surface data augmentation is critical for good generalization: in practice we randomly deform the coarse geometry using a deformation space spanned by the Laplace-Beltrami eigenfunctions. Please refer to supplementary material for more implementation details and to the discussion section for future approaches to improving the animation quality.

7 RESULTS

In this section, we showcase our results. We start by evaluating our aligned registered meshes, introducing the comparison methods and the metrics. Then, we display our driving results from sparse observations in two different settings. First, the body pose serves as the driving signal. Then, we use our pattern registrations obtained from two opposite-facing cameras as the driving signal. We show qualitative results and report the numerical driving errors in each case.

7.1 Evaluating the 3D Registration Accuracy

7.1.1 Triangulation. Figure 7 evaluates the coverage of our triangulated point clouds and the average visibility of different cloth regions. The precision of the registered point cloud we obtained from RANSAC triangulation is at least 3 inlier intersecting rays within an intersection radius of 1 mm.

7.1.2 Comparison to Related Work. Figure 11 showcases our registration quality by projecting our registered surface to the original image as a wireframe and our reconstructed cloth surface on the left. On the right we show comparisons with related work. We compare our registered meshes with our implementation of cloth registration pipeline in ClothCap [Pons-Moll et al. 2017][1]. In this method, a kinematic body model is used to track a single-layer surface describing the clothed subject. Using the T-pose frame, the cloth topology is segmented from the single-layer topology and used to define a cloth template. Then, this cloth template is registered to the clothing region of the single-layer surface using non-rigid Iterative Closest Point (ICP). For better initialization, we used Biharmonic Deformation Fields [Jacobson et al. 2010] to align the boundary between the deformed template and the target mesh such that the interior distortion is minimal as suggested by Xiang et al. [2021]. We also compare our registered meshes to those produced by the inverse rendering optimization stage presented by Xiang et al. [2021], which aimed to refine the non-rigid ICP registration described above. Finally, we compare our reconstructed geometry to the unregistered multi-view stereo reconstruction [Yu et al. 2021].

[1]We use our implementation of ClothCap [Pons-Moll et al. 2017] for all the comparison.

Visual examination of the registration results reveals a few observations. First, the reconstruction quality, especially in the wrinkles, is the best with our registration pipeline, even when compared to the unregistered multi-view stereo reconstruction [Yu et al. 2021]. Secondly, since our method does not rely on a kinematic body model for its operation, it is stable even when the cloth moves tangentially over the body or detaches from the body, as can be seen by examining the sleeve region in Figure 11. On the contrary, ClothCap [Pons-Moll et al. 2017] relies on the kinematic body model and shows artifacts at the sleeve region where the cloth is not perfectly overlapping with the body model. More severe instability due to tangential movement appears in the supplementary video. Similarly, Xiang et al. [2021] also rely on the body kinematic model for initialization, but later it refines the registration using inverse rendering optimization. This stage helps fix the initial artifacts but is vulnerable in the invisible cloth regions, such as the armpits and the areas occluded by the body parts, producing artifacts visible in Figure 11 and 14 and more severe artifacts visible in the supplementary video. Additionally, the photometric optimization produces more blurred geometry since the geometry is modeled by an autoencoder (bottleneck effect). On the other hand, our registration pipeline has smooth results, even in the invisible regions.

Figure 12 showcases a few sampled frames from our registered mesh sequence, and Figure 13 displays the different subjects and zoom-in to compare the rendered registered mesh with the captured images.

7.1.3 Error Metrics. Common error metrics such as pixel reprojection and rendering errors are hard to transfer from one capture system to another. To provide reproducible metrics that are invariant to the specific multi-view camera setting, we propose two novel absolute measurements. The first is the correspondence drift rate in [mm/sec] units. The second is the average geodesic distortion in [mm] over the sequence of frames. Table 1 reports these metrics for our work that future work can reference when accurate ground truth is unavailable.

To measure the correspondence drift rate, we unwrap the image texture to the UV domain and measure the optical flow. We translate the optical flow magnitude to correspondence drift rate in [mm/sec] units, using the square size calibration and the known camera frame rate. We also report the per-frame average geodesic distortion (a.k.a Gromov Hausdorf distortion) w.r.t a reference frame in a T-pose. To this end, we randomly sampled the vertex pairs, measured their pairwise geodesic distance, and compared it to the geodesic distance measured in the reference frame. We used *pygeodesic* implementation of the exact geodesic algorithm for triangular meshes [Mitchell et al. 1987]. We don't report the Chamfer distance to the reconstruction surface extracted from the multi-view stream using the work of Yu et al. [2021] because our reconstruction quality is better than this ground-truth candidate, leaving this measurement too coarse for evaluation. The table clearly shows that our correspondence drift rate is negligible relative to the compared methods. Our geodesic distortion is significantly lower. The former is directly related to the drift a texture applied to the geometry will exhibit. The latter is related to the deviation of the reconstructed geometry

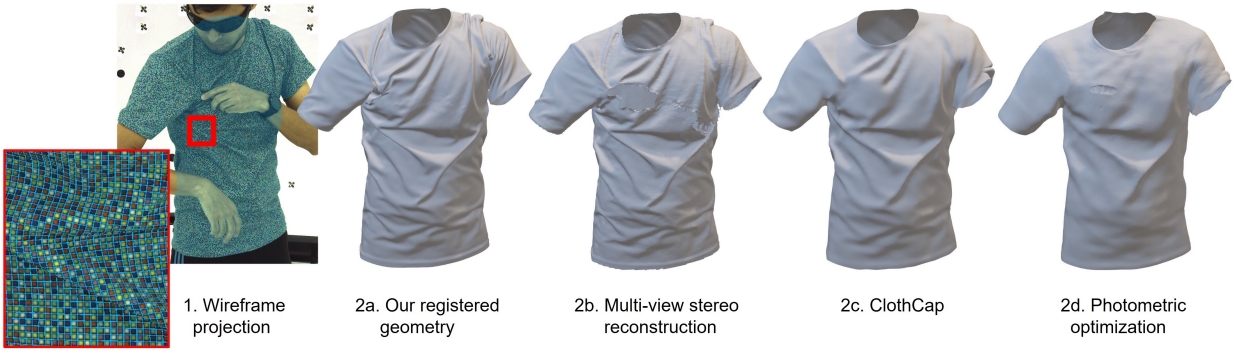


Fig. 11. **Registration Qualitative Evaluation.** From left to right 1) projecting our registered mesh with a wire-frame texture (aligned with the pattern square grid) to the image 2) Registered mesh geometry: a. ours b. multi-view stereo reconstruction [Yu et al. 2021] c. body tracking + Non-rigid ICP using [Pons-Moll et al. 2017] d. body tracking + Non-rigid ICP using + photometric optimization [Xiang et al. 2021].

Fig. 12. Several sample frames from our multi-view aligned mesh sequence. Each row represents a different subject.

from the expected one, assuming the cloth surface deforms roughly isometrically.

We measured the registration metric also on a bigger pattern we used to ablate our registration pipeline. The big pattern square size is 4 mm, compared to the pattern we use of size 2.67 mm. We hoped those metrics would help compare those patterns, but the errors reported in the table for both patterns deviate from others within the precision of the optical flow and geodesic distortion measurement techniques, which are both approximately 1 mm. Hence, we conclude that we should devise a measurement with sub-millimetric precision to analyze the difference between them quantitatively, as the standard measures are too coarse.

Finally, Figure 14 shows our coherent texture mapping, as opposed to other registration methods for which the texture jitters and distort. We include the full videos in the supplementary material.

7.2 Driving from sparse observations

7.2.1 Driving from pose. To demonstrate that our accurate cloth registrations can enhance existing cloth driving methods, we evaluate the driving method proposed by Xiang et al. [2021] once with their registration data, which refines ClothCap registration [Pons-Moll et al. 2017], and once with our cloth registrations. Figure 15 shows the improved wrinkle modeling in the animated geometry and Table 2 reports the corresponding errors in the predicted geometry, showing a clear advantage for the network trained with our accurate registration. Alongside the improvement in the geometry, our registrations also promote a significant improvement in the appearance modeling. It resolves the challenging problem that all the codec-avatar methods face when trying to recover both unknown textures and unknown geometry from a given capture. Without accurately anchored geometry, the appearance, optimized by inverse rendering, achieves poor results for non-trivial textures, see Figure 16.

7.2.2 Driving from Pattern Registrations. Here, we showcase the results of our sparse-view animation module trained with our high-quality registered clothes data. We evaluate the generalization of our driving network for the different deformation sequence that appears in the test set. Figure 17 shows selected sample frames of

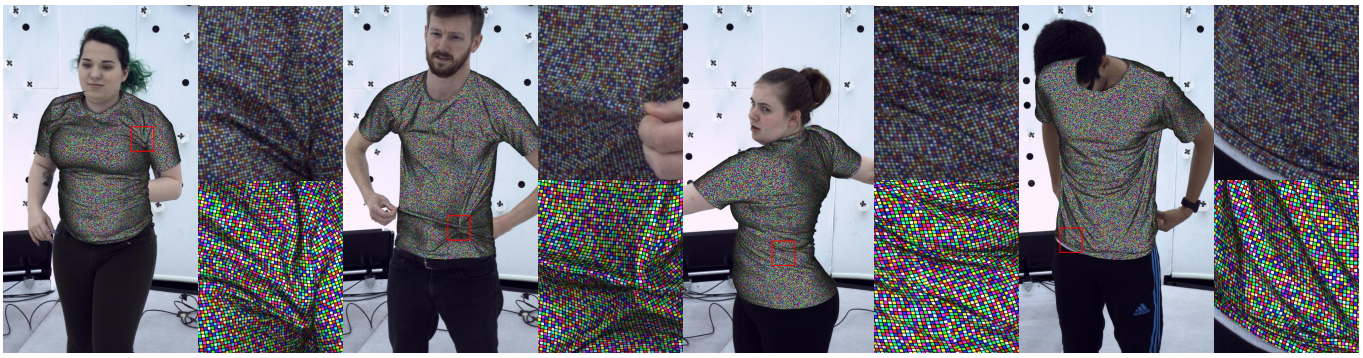


Fig. 13. We captured 4 subjects performing different kind of motions like jogging, hand cloth interactions, and torso rotation. Our method provides precise registration on regions of good visibility and a reasonable approximation on regions with reduced visibility (e.g., armpit and hand contact points). Precise registration on regions with fold overs (right) remains challenging to our current system.

Table 1. Quantitative evaluation of the registration method

Registration errors		
Method	Correspondence drift rate [mm/sec]	Geodesic distortion [mm]
Our pattern registration	1.50	9.27
Our pattern registration (big pattern)	0.77	9.98
Body tracking + non-rigid ICP [Pons-Moll et al. 2017]	72.3	25.94
Body tracking + non-rigid ICP + IR optimization [Xiang et al. 2021]	64.6	35.49

Table 2. Driving metric evaluation

Driving errors w.r.t. ground truth registered surfaces		
Method	Euclidean distance [mm]	Chamfer distance [mm]
Our pixel registration driving (surface kinematic model)	2.08	1.59
Our pixel registration driving (LBS model)	7.22	4.05
Pose driving [Xiang et al. 2021] trained with our registrations	15.89	5.05
Pose driving [Xiang et al. 2021] trained with non-rigid ICP registration	22.64	5.42

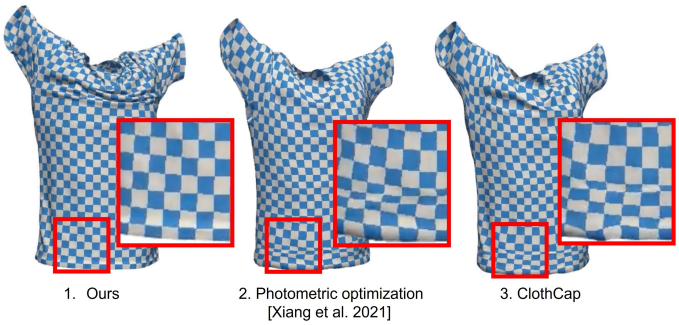


Fig. 14. From left to right: along the deformation sequence, custom texture is overlayed coherently with our registration (1), while with (2) [Xiang et al. 2021] and (3) [Pons-Moll et al. 2017] the texture jitters and distort

the resulting animated cloth sequence. Since our method processes a dense driving signal, while pose-driven neural models rely on the sparse skeletal joint parameters, we cannot directly compare them. Whether to apply pose driving or drive from pixel registration depends on the specific application. Figure 18 shows the level of detail and realism that the pixel registration driving adds to the final driving results. We show the pose-driven network [Xiang et al. 2021] and our pixel registration-driven model, both trained with our high-quality registration data, differing in the driving mechanism. Table 2 reports the driving errors for each driving method. Finally, to empirically justify our choice of the surface kinematic model, we ablate the kinematic model used to estimate the coarse geometry in our method. We report the driving errors when LBS is used as the kinematic model and when we use our surface kinematic model.

8 DISCUSSION, LIMITATIONS AND FUTURE WORK

We developed a pattern registration and surface alignment pipeline producing a dynamic sequence of tracked meshes representing the cloth deformation in a fine-grained resolution, low localization error, and correspondence accuracy significantly better than any prior art. We design our pattern detector to be robust to illumination due to the pattern primitives (processed in grayscale), the significant contrast between adjacent cells, and the extensive image augmentation we applied in training.

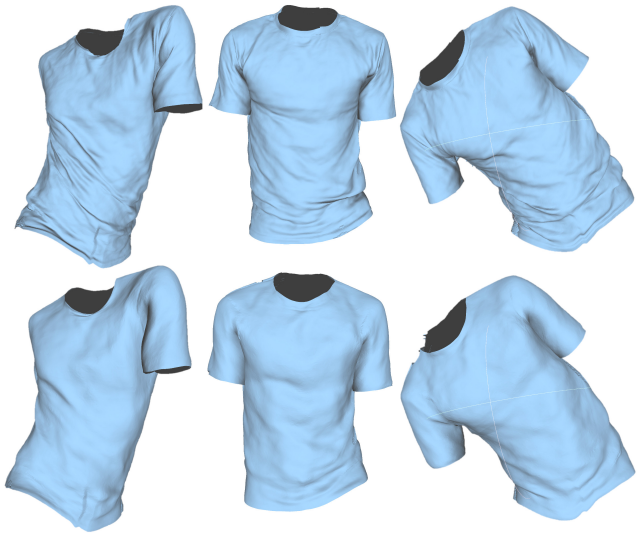


Fig. 15. Qualitative evaluation of the animated geometry training for pose driving with our registrations (top) and with ClothCap [Pons-Moll et al. 2017] registrations (bottom)

Fig. 16. Our high-quality cloth registrations improve the performance of the appearance model in the pose driving network. Training for pose driving with our registrations (left), and with the registration pipeline in [Pons-Moll et al. 2017] (right)

Fig. 17. A selected set of frames of our cloth sequence animated from two cameras pixel registrations: frontal view (top), and rear view (bottom).

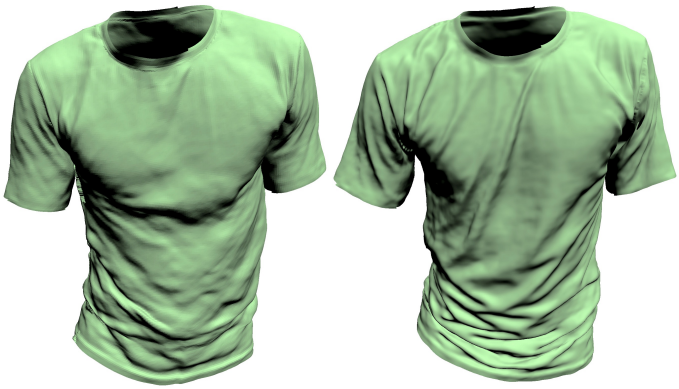


Fig. 18. Pixel registration driving (right) enhances the driving results' realistic geometry modeling compared to pose driving (left).

We design our system to be garment-agnostic by avoiding body priors, instead modeling the garment as a general non-rigid surface. We demonstrated robustness of our system in challenging occlusion and deformation configurations produced by hand-cloth interactions. Based on Figure 7, we expect our system to provide good registration of dynamic garments under moderate surface visibility (>60%). Please refer to the supplementary material and video for registration results obtained with a patterned skirt. We could model garments with changing topology as long as a template continuously deforms to the different configurations (for a jacket, we would use an unzipped mesh as template). We could extend our deformation model to include local scaling parameters to support highly stretchy materials, like spandex.

As an application, we demonstrated that our cloth registration technique could improve existing pose-driving methods by training them with our accurate registrations without changing the inference protocol. Finally, we proposed a drivable Garment Avatar model generating realistic cloth animation facilitated by a dense tracking signal obtained from two roughly 180° opposing cameras capturing our patterned garment. We focused on this setup because it is more accessible than 3+ views and maximizes the optical coverage under the two-cameras constraint. Our experiments show that the entire surface is reconstructable even when triangulation is impossible, given only a single observation per surface point. This work aims to provide a proof of concept for driving from a dense tracking signal and doesn't intend to handle the most generic case, which we deferred to future work. We discuss these limitations below.

8.1 Limitations

Our registration coverage is currently limited to portions of the cloth with surrounding 3×3 visible cells. Regions with lower visibility (2×2 or 1×1 cells) could be localized using other existing priors. Regions where the pattern is not visible (see fold over on Figure 13 right) would require further detection and modeling effort. To guarantee high surface visibility in the data registration stage, we used a large number of cameras. In the supplementary material, we analyze the surface coverage and the registration accuracy versus the number of cameras to estimate how our method performs in a smaller setup. Guaranteeing fully collision-free results (while satisfying observation fitting and temporal smoothness) remains challenging. Using finer temporal and spatial resolutions or doing multiple sequential pass registrations are directions that could alleviate collision but were not explored due to computational cost (we prioritized scalability). Template construction requires manual intervention to close seams. Further supervision of the cloth fabrication process and exploring relevant works for automatization [Grim et al. 2016; Halimi and Kimmel 2018; Halimi et al. 2019b] are left for future work.

Similar to existing cloth driving models, our Garment Avatar model does not handle **driving of an unseen garment**. Capturing a larger corpus of garments and exploring architectures supporting shape and material variations would be an exciting research direction. Our setup assumes two cameras with **known calibration**, similarly to many pose-driven methods [Bagautdinov et al. 2021; Xiang et al. 2021] that use two cameras (or more) to triangulate the user's 3D pose. Auto-calibration could rely on the co-detection of human body landmarks in different views. The current form assumes the **same view configuration at train and driving time**. We could train the model with random camera combinations to allow new camera positioning at inference. To handle the arbitrariness in the camera provided in each channel, we could add the ray direction 3D field as a context to the pixel coordinates signal. One can expect a decrease in **optical coverage** under a disadvantageous positioning of the cameras. Hence, shape completion in regions uncovered in different camera placements is a relevant direction to pursue, incorporating non-visual priors. Furthermore, our animations exhibit oscillations in large undetectable areas of the cloth, for example, when the hand occludes the shirt (better to observe in the supplementary video). In this case, our pattern-based driving also produces sub-optimal results, while using non-visual priors in conjunction with the pattern could improve our approach. Our current effort targets high quality over **runtime performance**; see the supplementary for the per-module computation-time. Future work would seek to optimize our driving model to enable real-time performance.

8.2 Future directions

We aim to explore end-to-end models that jointly learn cloth deformation priors while fitting visual observations to improve registration quality. To this end, we plan to take advantage of differentiable rendering frameworks to increase registration coverage. At the same time, the current pipeline could produce doubled resolution by registering the corners with the centers.

To enhance the driven reconstruction along the seams, we plan to implement our pixel-driving network in the mesh domain, utilizing mesh convolutions and mesh pooling and un-pooling layers [Hanocka et al. 2019; Zhou et al. 2020] which will replace the current layers in the UV domain. Our driving model operates independently per frame, exhibiting slight temporal jitter. We plan to overcome this issue by constructing a recurrent architecture for cloth animation [Santesteban et al. 2019, 2021].

We consider that our current dataset could benefit existing and novel research on building realistic models of cloth deformation. For instance, the dataset could help fine-tune simulation methods, improve cloth deformation inference from body pose and shape, and facilitate novel hand-cloth interaction models. Recently, a self-supervised model for learning cloth dynamics was introduced in [Santesteban et al. 2022]. Their model overcomes the requirement to generate simulation training data, supervising the model using a physic-based loss. An interesting analysis would be inferring the material parameters from our mesh sequences, training their self-supervised model with those parameters, and comparing the produced sequences to our pose-driving results. Finally, we left the appearance modeling completely unexplored. However, we demonstrated a highly coherent texture signal when projecting the camera images to our accurate geometry, see Section 7.1.3. The latter implies that learning the function that maps geometry to shading from pure observations should be feasible.

ACKNOWLEDGMENTS

We thank Printful Inc. and Luz Diaz-Arbelaez for assistance on cloth fabrication. Alec Hnat, Autumn Trimble, Bo Peng, Dani Belko, Gadsden Merrill, Hannah Kirschner, Julia Buffalini, Kevyn McPhail, Mallorie Mize, Philippe de Bree, and Rohan Bali for assistance on data capture. Aaqib Habib and Lucas Evans for supervising annotations. Kaiwen Guo and Yuan Dong for contributions to data pre-processing. Oshri Halimi is funded by the Israel Ministry of Science and Technology grant number 3-14719.

REFERENCES

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

Souhaib Attaiki, Gautam Pai, and Maks Ovsjanikov. 2021. DPFM: Deep Partial Functional Maps. *arXiv preprint arXiv:2110.09994* (2021).

Timur Bagautdinov, Chenglei Wu, Tomas Simon, Fabian Prada, Takaaki Shiratori, Shih-En Wei, Weipeng Xu, Yaser Sheikh, and Jason Saragih. 2021. Driving-signal aware full-body avatars. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–17.

Seungbae Bang, Maria Korosteleva, and Sung-Hee Lee. 2021. Estimating Garment Patterns from Static Scan Data. In *Computer Graphics Forum*. Wiley Online Library.

David Baraff and Andrew Witkin. 1998. Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. 43–54.

David Baraff, Andrew Witkin, and Michael Kass. 2003. Untangling cloth. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 862–870.

Jan Bednarik, Shaifali Parashar, Erhan Gundogdu, Mathieu Salzmann, and Pascal Fua. 2020. Shape reconstruction by learning differentiable surface representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4716–4725.

Christian Bersch, Benjamin Pitzer, and Sören Kammel. 2011. Bimanual robotic cloth manipulation for laundry folding. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1413–1419.

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. 2020. CLOTH3D: clothed 3d humans. In *European Conference on Computer Vision*. Springer, 344–359.

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. 2021. PBNS: Physically Based Neural Simulation for Unsupervised Garment Pose Space Deformation. *ACM Trans.*

Graph. 40, 6, Article 198 (dec 2021), 14 pages. https://doi.org/10.1145/3478513.3480479

Kiran S Bhat, Christopher D Twigg, Jessica K Hodgins, Pradeep Khosla, Zoran Popovic, and Steven M Seitz. 2003. Estimating cloth simulation parameters from video. (2003).

Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. 2020a. Combining implicit function learning and parametric models for 3d human reconstruction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 311–329.

Bharat Lal Bhatnagar, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. 2020b. Loopreg: Self-supervised learning of implicit surface correspondences, pose and shape for 3d human mesh registration. *arXiv preprint arXiv:2010.12447* (2020).

Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. 2019. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5420–5430.

Amit Bracha, Oshri Halim, and Ron Kimmel. 2020. Shape Correspondence by Aligning Scale-invariant LBO Eigenfunctions. (2020).

Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. 2008. Markerless garment capture. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 1–9.

D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. 2012. A naturalistic open source movie for optical flow evaluation. In *European Conf. on Computer Vision (ECCV) (Part IV, LNCS 7577)*, A. Fitzgibbon et al. (Eds.) (Ed.). Springer-Verlag, 611–625.

He Chen, Hyojoon Park, Kutay Macit, and Ladislav Kavan. 2021. Capturing Detailed Deformations of Moving Human Bodies. *arXiv preprint arXiv:2102.07343* (2021).

Cheng Chi and Shuran Song. 2021. GarmentNets: Category-Level Pose Estimation for Garments via Canonical Space Shape Completion. *arXiv preprint arXiv:2104.05177* (2021).

Toby Chong, I-Chao Shen, Nobuyuki Umetani, and Takeo Igarashi. 2021. Per Garment Capture and Synthesis for Real-time Virtual Try-on. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 457–469.

David Clyde, Joseph Teran, and Rasmus Tamstorf. 2017. Modeling and data-driven parameter estimation for woven fabrics. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 1–11.

Luca Cosmo, Antonio Norelli, Oshri Halimi, Ron Kimmel, and Emanuele Rodolà. 2020. Limp: Learning latent shape representations with metric preservation priors. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 19–35.

Vinh Ninh Dao and Masanori Sugimoto. 2010. A robust recognition technique for dense checkerboard patterns. In *2010 20th International Conference on Pattern Recognition*. IEEE, 3081–3084.

A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. 2015. FlowNet: Learning Optical Flow with Convolutional Networks. In *IEEE International Conference on Computer Vision (ICCV)*. http://lmb.informatik.uni-freiburg.de/Publications/2015/DFIB15

Marvin Eisenberger, Zorah Lähner, and Daniel Cremers. 2019. Divergence-Free Shape Correspondence by Deformation. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 1–12.

Marvin Eisenberger, Zorah Lahner, and Daniel Cremers. 2020a. Smooth shells: Multi-scale shape registration with functional maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12265–12274.

Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. 2020b. Deep Shells: Unsupervised Shape Correspondence with Optimal Transport. *arXiv preprint arXiv:2010.15261* (2020).

Mark Fiala. 2005. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2. IEEE, 590–596.

Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47, 6 (2014), 2280–2292.

Anna Grim, Timothy O'Connor, Peter J Olver, Chehrzad Shakiban, Ryan Slechta, and Robert Thompson. 2016. Automatic reassembly of three-dimensional jigsaw puzzles. *International Journal of Image and Graphics* 16, 02 (2016), 1650009.

Marc Habermann, Lingjie Liu, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. 2021a. Real-time Deep Dynamic Characters. *arXiv preprint arXiv:2105.01794* (2021).

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

Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. 2021b. A Deeper Look into DeepCap. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

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

Oshri Halimi, Ido Imanuel, Or Litany, Giovanni Trappolini, Emanuele Rodolà, Leonidas Guibas, and Ron Kimmel. 2020. Towards Precise Completion of Deformable Shapes. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*. Springer, 359–377.

Oshri Halimi and Ron Kimmel. 2018. Self functional maps. In *2018 International Conference on 3D Vision (3DV)*. IEEE, 710–718.

Oshri Halimi, Or Litany, Emanuele Rodola, Alex M Bronstein, and Ron Kimmel. 2019a. Unsupervised learning of dense shape correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4370–4379.

Oshri Halimi, Dan Raviv, Yonathan Aflalo, and Ron Kimmel. 2019b. Computable invariants for curves and surfaces. In *Handbook of Numerical Analysis*. Vol. 20. Elsevier, 273–314.

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

Daniel Holden, Bang Chi Duong, Sayantan Datta, and Derek Nowrouzezahrai. 2019. Subspace neural physics: Fast data-driven interactive simulation. In *Proceedings of the 18th annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 1–12.

Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Linjie Luo, Chongyang Ma, and Hao Li. 2018. Deep volumetric video from very sparse multi-view performance capture. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 336–354.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1125–1134.

Alec Jacobson, Elif Tosun, Olga Sorkine, and Denis Zorin. 2010. Mixed finite elements for variational surface modeling. In *Computer graphics forum*, Vol. 29. Wiley Online Library, 1565–1574.

Ning Jin, Yilin Zhu, Zhenglin Geng, and Ronald Fedkiw. 2020. A Pixel-Based Framework for Data-Driven Clothing. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 135–144.

Zorah Lahner, Daniel Cremers, and Tony Tung. 2018. Deepwrinkles: Accurate and realistic clothing modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 667–684.

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

Yue Li, Marc Habermann, Bernhard Thomaszewski, Stelian Coros, Thabo Beeler, and Christian Theobalt. 2021. Deep physics-aware inference of cloth deformation for monocular human performance capture. In *2021 International Conference on 3D Vision (3DV)*. IEEE, 373–384.

Junbang Liang, Ming Lin, and Vladlen Koltun. 2019. Differentiable cloth simulation for inverse problems. (2019).

Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. 2017. Deep functional maps: Structured prediction for dense shape correspondence. In *Proceedings of the IEEE international conference on computer vision*. 5659–5667.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015a. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34, 6 (Oct. 2015), 248:1–248:16.

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

Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J Black. 2021a. SCALE: Modeling clothed humans with a surface codec of articulated local elements. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16082–16093.

Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. 2021b. SCALE: Modeling Clothed Humans with a Surface Codec of Articulated Local Elements. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J Black. 2020a. Learning to dress 3d people in generative clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6469–6478.

Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. 2020b. Learning to Dress 3D People in Generative Clothing. In *Computer Vision and Pattern Recognition (CVPR)*.

Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J. Black. 2021c. The Power of Points for Modeling Humans in Clothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. 2003. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. In *Visualization and Mathematics III*, Hans-Christian Hege and Konrad Polthier (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 35–57.

Eder Miguel, Derek Bradley, Bernhard Thomaszewski, Bernd Bickel, Wojciech Matusik, Miguel A Otaduy, and Steve Marschner. 2012. Data-driven estimation of cloth

simulation models. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 519–528.

Stephen Miller, Jur Van Den Berg, Mario Fritz, Trevor Darrell, Ken Goldberg, and Pieter Abbeel. 2012. A geometric approach to robotic laundry folding. *The International Journal of Robotics Research* 31, 2 (2012), 249–267.

Joseph SB Mitchell, David M Mount, and Christos H Papadimitriou. 1987. The discrete geodesic problem. *SIAM J. Comput.* 16, 4 (1987), 647–668.

Rahul Narain, Armin Samii, and James F O'brien. 2012. Adaptive anisotropic remeshing for cloth simulation. *ACM transactions on graphics (TOG)* 31, 6 (2012), 1–10.

Gaku Narita, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2016. Dynamic projection mapping onto deforming non-rigid surface using deformable dot cluster marker. *IEEE transactions on visualization and computer graphics* 23, 3 (2016), 1235–1248.

Ryota Natsume, Shunsuke Saito, Zeng Huang, Weikai Chen, Chongyang Ma, Hao Li, and Shigeo Morishima. 2019. Siclope: Silhouette-based clothed people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4480–4490.

Edwin Olson. 2011. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*. IEEE, 3400–3407.

Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. 2020. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7365–7375.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. 2019. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 10975–10985.

Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael Black. 2017. ClothCap: Seamless 4D Clothing Capture and Retargeting. *ACM Transactions on Graphics, (Proc. SIGGRAPH)* 36, 4 (2017). http://dx.doi.org/10.1145/3072959.3073711

David Pritchard and Wolfgang Heidrich. 2003. Cloth motion capture. In *Computer Graphics Forum*, Vol. 22. Wiley Online Library, 263–271.

Abdullah Haroon Rasheed, Victor Romero, Florence Bertails-Descoubes, Stefanie Wuhrer, Jean-Sébastien Franco, and Arnaud Lazarus. 2020. Learning to Measure the Static Friction Coefficient in Cloth Contact. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9912–9921.

Tom FH Runia, Kirill Gavrilyuk, Cees GM Snoek, and Arnold WM Smeulders. 2020. Cloth in the wind: A case study of physical measurement through simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10498–10507.

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

Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. 2020. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 84–93.

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

Igor Santesteban, Miguel A Otaduy, and Dan Casas. 2019. Learning-based animation of clothing for virtual try-on. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 355–366.

Igor Santesteban, Miguel A Otaduy, and Dan Casas. 2022. SNUG: Self-Supervised Neural Dynamic Garments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8140–8150.

Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. 2021. Self-Supervised Collision Handling via Generative 3D Garment Models for Virtual Try-On. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11763–11773.

Rohan Sawhney and Keenan Crane. 2017. Boundary First Flattening. 37, 1, Article 5 (dec 2017), 14 pages. https://doi.org/10.1145/3132705

Volker Scholz, Timo Stich, Marcus Magnor, Michael Keckeisen, and Markus Wacker. 2005. Garment motion capture using color-coded patterns. In *ACM SIGGRAPH 2005 Sketches*. 38–es.

Olga Sorkine and Marc Alexa. 2007. As-Rigid-As-Possible Surface Modeling. In *Proceedings of EUROGRAPHICS/ACM SIGGRAPH Symposium on Geometry Processing*. 109–116.

Fabio Strazzeri and Carme Torras. 2021. Topological representation of cloth state for robot manipulation. *Autonomous Robots* 45, 5 (2021), 737–754.

Tuur Stuyck. 2018. Cloth Simulation for Computer Graphics. *Synthesis Lectures on Visual Computing: Computer Graphics, Animation, Computational Photography, and Imaging* 10, 3 (2018), 1–121.

Zhuo Su, Lan Xu, Zerong Zheng, Tao Yu, Yebin Liu, and Lu Fang. 2020. Robustfusion: Human volumetric capture with data-driven visual cues using a rgbd camera. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*. Springer, 246–264.

Robert W. Sumner, Johannes Schmid, and Mark Pauly. 2007. Embedded Deformation for Shape Manipulation. *ACM Trans. Graph.* 26, 3 (jul 2007), 80–es. https://doi.org/10.1145/1276377.1276478

Garvita Tiwari, Bharat Lal Bhatnagar, Tony Tung, and Gerard Pons-Moll. 2020. Sizer: A dataset and model for parsing 3d clothing and learning size sensitive 3d clothing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 1–18.

Giovanni Trappolini, Luca Cosmo, Luca Moschella, Riccardo Marin, Simone Melzi, and Emanuele Rodolà. 2021. Shape registration in the time of transformers. *Advances in Neural Information Processing Systems* 34 (2021).

Huamin Wang, James F O'Brien, and Ravi Ramamoorthi. 2011. Data-driven elastic models for cloth: modeling and measurement. *ACM transactions on graphics (TOG)* 30, 4 (2011), 1–12.

John Wang and Edwin Olson. 2016. AprilTag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4193–4198.

Ryan White, Keenan Crane, and David A Forsyth. 2007. Capturing and animating occluded cloth. *ACM Transactions on Graphics (TOG)* 26, 3 (2007), 34–es.

Donglai Xiang, Fabian Prada, Timur Bagautdinov, Weipeng Xu, Yuan Dong, He Wen, Jessica Hodgins, and Chenglei Wu. 2021. Modeling Clothing as a Separate Layer for an Animatable Human Avatar. *arXiv preprint arXiv:2106.14879* (2021).

Donglai Xiang, Fabian Prada, Chenglei Wu, and Jessica Hodgins. 2020. MonoClothCap: Towards temporally coherent clothing capture from monocular RGB video. In *2020 International Conference on 3D Vision (3DV)*. IEEE, 322–332.

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

Mustafa B. Yaldiz, Andreas Meuleman, Hyeonjoong Jang, Hyunho Ha, and Min H. Kim. 2021. DeepFormableTag: End-to-end Generation and Recognition of Deformable Fiducial Markers. *ACM Transactions on Graphics (Proc. SIGGRAPH 2021)* 40, 4 (2021).

Zhaoyi Yan, Xiaoming Li, Mu Li, Wangmeng Zuo, and Shiguang Shan. 2018. Shift-net: Image inpainting via deep feature rearrangement. In *Proceedings of the European conference on computer vision (ECCV)*. 1–17.

Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. 2021. Function4D: Real-time Human Volumetric Capture from Very Sparse Consumer RGBD Sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5746–5756.

Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. 2018. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7287–7296.

Chao Zhang, Sergi Pujades, Michael J Black, and Gerard Pons-Moll. 2017. Detailed, accurate, human shape estimation from clothed 3D scan sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4191–4200.

Yi Zhou, Chenglei Wu, Zimo Li, Chen Cao, Yuting Ye, Jason Saragih, Hao Li, and Yaser Sheikh. 2020. Fully convolutional mesh autoencoder using efficient spatially varying kernels. *arXiv preprint arXiv:2006.04325* (2020).