Learning Cloth Dynamics: 3D+Texture Garment Reconstruction Benchmark

Meysam Madadi MMADADI@CVC.UAB.ES
Computer Vision Center, Barcelona, Spain

Hugo Bertiche HUGO_BERTICHE@HOTMAIL.COM
Universitat de Barcelona and Computer Vision Center, Spain

Wafa Bouzouita WAFE.BOUZOUITA@GMAIL.COM
Computer Vision Center, Barcelona, Spain

Isabelle Guyon GUYON@CHALEARN.ORG
Université Paris-Saclay and INRIA, France

Sergio Escalera SERGIO@MAIA.UB.ES
Universitat de Barcelona and Computer Vision Center, Spain

Editors: Hugo Jair Escalante and Katja Hofmann

Abstract

Human avatars are important targets in many computer applications. Accurately tracking, capturing, reconstructing and animating the human body, face and garments in 3D are critical for human-computer interaction, gaming, special effects and virtual reality. In the past, this has required extensive manual animation. Regardless of the advances in human body and face reconstruction, still modeling, learning and analyzing human dynamics need further attention. In this paper we plan to push the research in this direction, e.g. understanding human dynamics in 2D and 3D, with special attention to garments. We provide a large-scale dataset (more than 2M frames) of animated garments with variable topology and type, called **CLOTH3D++**. The dataset contains RGBA video sequences paired with its corresponding 3D data. We pay special care to garment dynamics and realistic rendering of RGB data, including lighting, fabric type and texture. With this dataset, we hold a competition at NeurIPS2020. We design three tracks so participants can compete to develop the best method to perform 3D garment reconstruction in a sequence from (1) 3D-to-3D garments, (2) RGB-to-3D garments, and (3) RGB-to-3D garments plus texture. We also provide a baseline method, based on graph convolutional networks, for each track. Baseline results show that there is a lot of room for improvements. However, due to the challenging nature of the problem, no participant could outperform the baselines.

Keywords: 3D garment reconstruction, texture prediction, CLOTH3D++, dynamics, NeurIPS2020

1. Introduction

3D human reconstruction from still images has been widely explored over the last few years. This is due to the wide applicability it has in entertainment and video game industries, and recently, in the VR/AR domains as well. Understanding of 3D scenarios allows for a higher level of human-computer interaction. While body pose and shape regression has undergone a significant progress by the scientific community, new research lines focus on recovering

© 2021 M. Madadi, H. Bertiche, W. Bouzouita, I. Guyon & S. Escalera.

garments along the body. This task comes with additional challenges, such as more complex geometries, different topologies and color variability.

Despite the growth of deep learning algorithms in the domain of RGB-based 3D garment reconstruction, there is not a common benchmark dataset available to compare these algorithms. While capturing RGB data from the real world is easy, obtaining accurate and rich annotations for garments is very challenging. Due to this, available datasets are either of small-scale von Marcard et al. (2018); Wang et al. (2018); Zhu et al. (2020) or poor in terms of cloth dynamics Pumarola et al. (2019). Recently, Bertiche et al. (2019) proposed CLOTH3D dataset, a large scale synthetic dataset with variable pose, garment type and dynamics. However, it is a purely 3D dataset with no RGB data. We aim to complement these data through realistic RGBA renderings of its animated 3D dressed humans. We then extend CLOTH3D dataset into a more complete dataset, presented as CLOTH3D++[1] (see Fig. 1 for some examples), the first large-scale video dataset of animated dressed humans with dense 3D annotations for body and cloth, as well as high resolution textures. Therefore, one can train deep algorithms to reconstruct full garments, i.e. 3D plus texture. CLOTH3D++ is compared with available RGB-based 3D garment datasets in Tab. 1.

Available garment generation techniques are either image-to-image, which are not explicitly aware of garment dynamics and body-garment interactions, or 3D-to-3D without texture. In this paper, we give a special attention to the **image-based 3D garment and texture reconstruction** which is a highly dynamic problem with objects of variable topology and shape. In this regard, we implement baselines based on convolutional graph neural networks for three tasks: 1) garment animation, 2) RGB-based 3D garment reconstruction and 3) texture estimation; and conduct a competition on CLOTH3D++ benchmark. This is the first event taking a deep look at garment dynamics, either in data structures, deep models or evaluation metrics.

Briefly, our contributions are:

- We build a new large scale synthetic dataset, CLOTH3D++, for RGB-based garment and texture reconstruction,
- We organize a competition on CLOTH3D++ dataset in three tracks. We build a platform, implement baselines and necessary code to work with the data, design the rules and evaluation metrics,
- We thoroughly discuss the results.

2. Related Work

RGB-based 3D Garment Datasets. To date, only a few available repositories focus on RGB-based 3D garment datasets. The 3DPW dataset von Marcard et al. (2018), captured from outdoor scenes, contains 18 clothed models that can be shaped and posed as SMPL. This dataset does not provide garment texture and its main focus is on shape and pose. Recently, the authors of Bhatnagar et al. (2019) propose a dataset of garments and body shapes from 3D scans. However, the amount of samples is in the order of a few hundreds and is limited to five clothing styles. More recently, Zhu et al. (2020) proposes a real data of over 2000 clothing models. Similar to 3DPW, garment texture is not provided. The works of Wang et al. (2018); Pumarola et al. (2019) propose synthetic datasets generated through

1. The dataset and starting kit can be downloaded here: http://chalearnlap.cvc.uab.es/dataset/38/description/

Dataset	3DPW	Untitled	3DPeople	Fashion3D	CLOTH3D++
Resolution	2.5cm	1cm	-	$\approx 0.5cm$	1cm
Missing	x	x	x	✓	x
Dynamics	x	x	x	x	✓
Garments	18	3	High	563	12.9K
Texture	x	✓	✓	x	✓
Fabrics	x	x	x	x	✓
Poses	Low	Very low	Low	Low	High
Subjects	18	2K	80	Low	9.7K
Layered	x	✓	-	x	✓
#samples	51k	24K	2.5M	2078	2.2M
Type	Real scan	Synth.	Synth.	Real Multi-view	Synth.
GT error	26mm	None	None	Unknown	None

Table 1: CLOTH3D++ vs. publicly available 3D cloth datasets (i.e. 3DPW (von Marcard et al. (2018)), Untitled (Wang et al. (2018)), 3DPeople (Pumarola et al. (2019)) and Fashion3D (Zhu et al. (2020))). CLOTH3D++ shows rich annotations and features, including fabrics and high variability in garment type, dynamics and body pose. It provides a challenging benchmark to study RGB-based 3D garment reconstruction.

physical simulation. The dataset of Wang et al. (2018) presents an automatic garment resizing based on real patterns, only providing static samples on a few different poses for three clothing styles. Closer to our work, 3DPeople dataset Pumarola et al. (2019) provides a large dataset of synthetic 3D humans with clothing. Nevertheless, this dataset differs from ours in many aspects. On one hand, this dataset is given as multi-view images. It includes RGB, depth, normal and scene flow, but not 3D models. On the other hand, the clothing are rigged without proper dynamics. Our CLOTH3D++ dataset aims to address previous datasets limitations. We provide a large-scale dataset (more than 2.4M frames) of animated garments with a huge variability on clothing type, topology, shape, size, tightness and fabric, with realistic cloth dynamics. In Tab. 1, we show detailed comparisons of properties for existing datasets and ours.

RGB to 3D garment reconstruction. Prior work based on parametric 3D body models encoding shape and pose deformations separately, being learnt from thousands of scans of real people Dragomir et al. (2005); Loper et al. (2015). These body models provide a good prior for 3D garment reconstruction. However, these models are trained to just capture the human body. There are attempts to reconstruct clothed body from video inputs Alldieck et al., 2018), RGB-D data Yu et al. (2019) and multi-view images Bhatnagar et al. (2019); Xu et al. (2019). Although, in these approaches, richer inputs clearly provide more information than a single image, the developed pipelines yield additional setup/hardware constraints and extra computation, limiting applicability. Recently, new approaches based on deep learning Varol et al. (2018); Sun et al. (2018); Saito et al. (2019); Ryota et al. (2019); Alldieck et al. (2019a,b); Zheng et al. (2019); Lazova et al. (2019) addressed single-view dressed body estimation. However, for all these methods, heavy manual post-processing is needed to extract the clothing surface from the reconstructed result. Furthermore, the reconstructed garments still lack realism. Closer to our work, there are few approaches

that propose to reconstruct garment as a layer separated from the body. DeepGarment Danerek et al. (2017) proposes to use physics based simulations as supervision for learning a garment shape estimation model. However, it only works for seen garments and does not provide realistic results. Lehnar *et al.* Lahner et al. (2018) design a method to synthesize garment wrinkles onto a coarse garment mesh following a given pose. This method, however, needs a computationally demanding step to register the template cloth to the captured 4D scan. Additionally, the method is limited to a fixed topology and cannot scale well to large deformations. Multi-Garment Net Bhatnagar et al. (2019) learns per-category garment reconstruction from images using 3D scanned data. However, this method typically requires 8 input RGB images and fails to reconstruct complex clothing topology such as skirts and dresses.

RGB to 3D garment and texture reconstruction. Some recent works Lazova et al. (2019); Alldieck et al. (2019b); Lahner et al. (2018) propose to use 2D UV map representation for estimating geometry and color details. Particularly, the Tex2Shape method of Alldieck *et al.* Alldieck et al. (2019b) aims to reconstruct high quality 3D geometry by regressing displacements in an unwrapped UV space. Nevertheless, this type of approach is limited by the topology of the template mesh (need of different mesh topology for skirts and dresses) and the topology of the UV parametrization (e.g. visible seam artifacts around texture seams). Other works Tulsiani et al. (2017); Sun et al. (2018) propose volumetric voxel representations for colored 3D reconstruction. For example, Im2Avatar Sun et al. (2018) performs textured single-image reconstruction, using colored voxels as the output representation. Other approaches Saito et al. (2019); Oechsle et al. (2019); Ryota et al. (2019) use implicit functions representation to recover shape and texture of clothed human bodies. Unlike explicit representations (e.g. meshes, voxels, point clouds) these methods learn functions to parametrize a 3D volume or surface. For instance, PiFu Saito et al. (2019) learns an implicit surface function based on aligned image features. This model generates clothes details but does not predict a realistic texture of the occluded regions of the dressed person (e.g. back of the person). Also, it is less robust to pose variations. Here, we propose a novel architecture that exploits the shape and topology of the human body to explicitly predict analytical 3D surfaces as garments from still images with a differentiable geometry. In addition, we show that it can be directly applied to color prediction as well.

3. Contest overview

In this section, we provide generic information about the competition including tracks, design and metrics.

3.1. Tracks

The competition consisted on three tasks:

- **3D-to-3D garment reconstruction:** participants must train their models on input 3D data including a 3D garment in rest pose, body shape and a sequence of body pose. The goal of this task was to learn garment dynamics to build generative models for 3D reconstruction.
- **Image-to-3D garment:** participants must reconstruct 3D garments from either a single RGB image or a sequence of images. This was a more challenging task since

proposed methodologies must be able to deal with lighting, occlusions, viewpoint, body pose, etc. In this (and next) track, we provided SMPL root joint location, along with RGBA images, available at inference time. The reason was to avoid ambiguities in scale due to subject size or distance to camera.

- **Image-to-3D garment and texture reconstruction:** In this task participants' models must be able to recover the color (without light effect) along with 3D garments. We did not restrict participants in the output format of predicted colors, that is, it could be a per vertex color or texture UV map. This was the most challenging task since the models must be able to deal with variable lighting and self-occlusions to predict a realistic texture along with 3D garments.

The predicted garments had to have a 3D mesh format for evaluation. Although we provided a ground truth grid topology per 3D garment in our dataset, participants had freedom in their model design and they could generate any topology they found more suitable for the data. Participants also could opt to submit their solutions for one or all tasks.

3.2. Design

We run the challenge in two stages: development and final phases.

- **Development Phase:** We released labeled training data (with meta-data) and unlabeled validation data at the beginning of this phase. Then, participants could train their models on training samples. Training data was common among all tracks while validation (and test) data was common among track 2 and 3 and different from track 1. During this phase, participants could submit their predictions on the validation data to the provided platform and obtain feedback on the leaderboard. This phase has been designed to allow participants to tune their models to avoid overfitting on the test set. This phase continued for six months.
- **Final Phase:** This phase started immediately after the development phase and continued for two weeks. One week before the end of development phase, due to the big size of data, we released encrypted validation set ground truth and unlabeled final (test) data for all 3 tracks such that participants with poor internet connection could download the data in time. In this phase, participants could finetune their models with validation data. The winners of each track were determined by the leaderboard rank of this phase.

We used the CodaLab platform[2] to run all the tracks with the aid of Google cloud (a quad-core CPU server) as the backend computing. Participants had to submit their predictions to Codalab. Due to huge prediction files ($\tilde{1}$GB) and long evaluation time, we limited participants to 1 daily submission per team and a maximum of 10 submissions during the final phase. Participants were not allowed to use any other data than the provided dataset for the purpose of 3D reconstruction. However, intelligent data augmentation was allowed. Finally, participants had to outperform our baseline score as a minimum requirement to enter the evaluation process. After the evaluation process, top three ranked participants for each track had to send code and fact sheets describing their methods to be eligible for prizes. In overall, 80, 58 and 55 participants registered in tracks 1, 2 and 3, respectively. However,

2. https://competitions.codalab.org/

no team could outperform our baseline scores in test stage. Therefore, the competition had no winner.

3.3. Metrics

In order to rank participants, we used the following metrics based on the task.

- **Track 1 and 2:** We evaluated predicted outfit surface using Surface-to-Surface distance (S2S), an extension of Chamfer distance (CD). It was computed based on the nearest face rather than nearest vertex. We defined S2S distance as:

$$\frac{1}{2N_1}\sum_{p\in S_1}\min_{f\in T_2} dist(p,f) + \frac{1}{2N_2}\sum_{p\in S_2}\min_{f\in T_1} dist(p,f), \tag{1}$$

  where S_1 and T_1 are the set of 3D surface vertices (with N_1 points) and mesh triangles for ground truth outfit, respectively, likewise S_2 and T_2 belong to the predicted outfit, and $dist(p,f)$ is the Euclidean distance between vertex p and the triangle f.
- **Track 3:** Quantitative evaluation for surface and texture did not guarantee that the top-ranked method necessarily predicts the best quality results, specially on texture. Therefore, in this track we measured the quality of the full reconstructed model. We measured this score through qualitative evaluation done by several human judges. Firstly, we ranked participants based on S2S metric and top 10 teams were considered in the qualitative measurements. Then, we reconstructed the full garment model (3D+texture) for a number of samples and asked the judges to answer the following questions by comparing teams in a pairwise manner.
  - Which team performs better in terms of realistic garment dynamics in the sequence?
  - Which team looks more similar to the ground truth in terms of garment type and 3D reconstruction?
  - Which team performs better in terms of realistic texture pattern? A team must be penalized if always generate the same texture pattern or a plain color.
  - Which team looks more similar to the ground truth in terms of texture pattern and color?

  This resulted in a tensor with size (#Participants, #Participants, #Questions, #Judges). The final score for each participant was the average over the last three dimensions. We ranked participants based on this score. Note that this scoring was done just once at the end of the competition and not shown in the leaderboard. However, we showed S2S score on the leaderboard during the development phase. Also, judges did not take into account human skin or hair color for their decisions. We provided a python script to the judges to visualize 3D predicted garments and rank them by the perceived generation quality.

4. CLOTH3D++ Dataset

Among the current available datasets on 3D garments, CLOTH3D Bertiche et al. (2019) has the highest subject and outfit variability, plus different fabrics and rich cloth dynamics. CLOTH3D is a purely 3D dataset of animated dressed humans (SMPL Loper et al. (2015)) obtained through Physically Based Simulation (PBS). For CLOTH3D++, we pick

Figure 1: CLOTH3D++ RGBA samples.

Figure 2: Left) PBR materials used for rendering. From left to right: skin, cotton, silk, denim and leather. Right) We generate uniform skin, hair and eye color variability as an effort towards an unbiased artificial intelligence.

CLOTH3D data and render RGB images with different textures, skin color and lighting. Additionally, we follow the same protocol as CLOTH3D to simulate and render new 3D garments as our test set. We show some examples of CLOTH3D++ in Fig. 1. Next, we explain details of the dataset.

4.1. Rendering

We use an unbiased ray-tracing engine, Cycles, integrated into an open-source 3D creation suite (Blender), to render realistic images. Such render engines obtain images through a PBS of light rays that appear in the 3D scene, achieving a natural light interaction w.r.t. the object materials and camera. We use Physically Based Rendering (PBR) materials for the different fabrics and human skin. Fig. 2 shows an example of each.

We render garments with different PBR materials and random uniform colors or textured patterns. To do so, 3D meshes are unwrapped into UV maps on top of a texture image. If not unwrapped properly, textures will show significant distortions on the renderings. Available tools (e.g. Blender) generate UV maps as non-continuous and non-uniform patches. This yields broken textures on rendering. In CLOTH3D++, we build an automatic tool to create continuous UV maps. We do so by computing seams of minimal path length that connect garment boundaries (see Sec. 4.4 for more details). We gathered over 100 texture images of size 2048×2048. Although, the choice of the garment textures could exhibit some bias in the dataset, we applied some transformations during rendering to minimize the possibility of repeating textures. More precisely, we transformed texture colors by shifting the image hue and randomly scaling saturation and value (HSV) to increase data variability. Additionally, we scaled UV maps randomly to obtain different pattern sizes. In Fig. 3 we illustrate some samples of texture images, the proposed UV maps and their corresponding renderings. Finally, aiming towards ethnically unbiased artificial intelligence, we sample different colors for hair, skin and pupils (See Fig. 2). For this same reason, we only store color labels (or patterns) for garments, not for the skin.

4.2. Setup

To generate garments in a sequence, we ensure the outfit covers both upper and lower body. An outfit is one layer of jumpsuit or dress, or a combination of two sets $top, t-shirt$ and $skirt, trousers$. CLOTH3D consists of dressed humans moving in a 3D space. In CLOTH3D++, to keep as much of the subject in the frame as possible, we center its trajec-

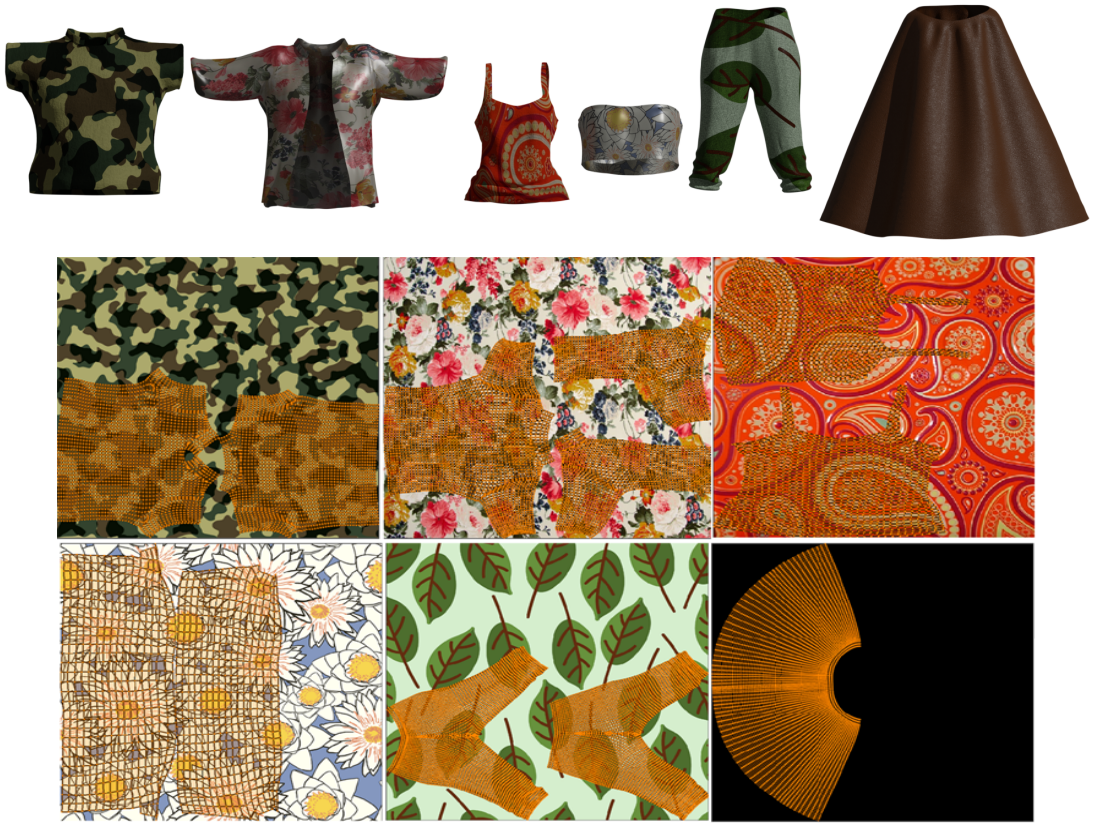

Figure 3: Top: 3D garments with PBR materials and color patterns. Bottom: associated UV maps on top of their corresponding color pattern. Note that even without a color pattern (skirt, bottom-right), it is still necessary an UV unwrapping for the PBR materials.

Train								
Garment type		Top	T-shirt	Trousers	Jumpsuit	Skirt	Dress	All
Gender (%)	Female	46.4	48.5	32.6	48.0	100	100	64.2
	Male	53.6	51.5	67.4	52.0	0	0	35.8
Texture (%)	Pattern	51.2	50.3	37.4	37.3	37.6	38.3	40.9
	Color	48.8	49.7	62.6	62.7	62.4	61.7	59.1
Fabric (%)	Silk	50.1	50.3	22.6	24.6	27.9	24.7	30.8
	Leather	0	0	26.2	25.5	25.3	25.9	19.3
	Cotton	49.9	49.7	24.5	25.0	24.4	24.9	31.1
	Denim	0	0	26.8	24.9	22.4	24.5	18.8
# vertices		1866±501	5311±992	7039±1955	10702±2577	4651±1215	8017±2160	7244±3397
# frames		320K	328K	505K	683K	143K	613K	1.95M
Test								
Garment type		Top	T-shirt	Trousers	Jumpsuit	Skirt	Dress	All
Gender (%)	Female	37.9	35.5	19.8	38.8	100	100	54.5
	Male	62.1	64.5	80.2	61.2	0	0	45.5
Texture (%)	Pattern	71.9	77.3	58.3	56.0	46.3	56.3	61.0
	Color	28.1	22.7	41.7	44.0	53.7	43.7	39.0
Fabric (%)	Silk	46.7	53.1	26.3	22.6	19.5	24.7	30.9
	Leather	0	0	26.0	25.6	30.2	26.8	19.4
	Cotton	53.3	46.9	24.4	26.5	26.8	22.9	31.5
	Denim	0	0	23.3	25.3	23.5	25.6	18.2
# vertices		1862±485	5325±976	6890±1989	10936±2539	4599±1187	8121±2198	7315±3451
# frames		56K	76K	105K	135K	27K	102K	246K

Table 2: CLOTH3D++ dataset statistics.

tory by subtracting the mean horizontal value (XY plane). Camera is always aligned with the X-axis at a uniformly sampled distance within range $[4, 6]$ meters. Unbiased viewpoint variability is obtained by randomly rotating the whole scene around Z-axis (vertical axis). Regarding lights, indoor scenes are emulated by randomly sampling one or few point lights (light bulbs) placed at a constant height (ceiling) but random position in the XY plane. Outdoor scene lightning is represented by a sun light (parallel rays from infinity) with random intensity and direction to simulate different weather conditions. Videos are rendered with a resolution of 640×480 at 30 fps. We opted for RGBA renderings (no background) to allow researchers full flexibility and put the focus on the 3D garment reconstruction and color/pattern retrieval problem.

4.3. Data statistics

We use CLOTH3D dataset as the ground truth for our training set (1.95M frames). Additionally, we simulate and render 2K sequences with a maximum length of 300 frames as our test set. We pick SMPL pose parameters of the test sequences from AMASS dataset Mahmood et al. (2019). Around 25% of the test poses are not seen in the training. Fig. 4 shows training and test pose distribution. CLOTH3D++ contains a rich set of annotations: 1) Body SMPL parameters, 2) Garment Type, fabric, topology, per-vertex location, UV map and RGB texture pattern, and 3) Scene camera matrix, light type and configuration. Dataset statistics are shown in Tab. 2.

4.4. UV unwrapping

CLOTH3D Bertiche et al. (2019) dataset contains thousands of sequences with different garments. Each of these garments is represented as a mesh. In order to provide of color patterns to these garments, a prior UV unwrapping step is necessary. Since each garment mesh structure is different, each must be unwrapped independently. Typical automatic unwrapping methods are not sufficient. Fig. 5 illustrates typical automatic unwrapping techniques against our proposed approach. The first approach flattens the mesh without splitting into submeshes. As it can be seen, this generates significant distortions. Flattening

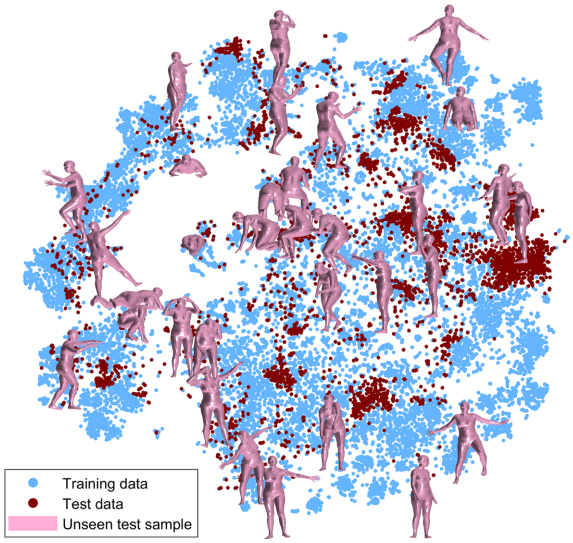

Figure 4: CLOTH3D++ SMPL pose distribution in training and test. t-SNE with perplexity of 50 is used. We show some examples of the test that are unseen during the training. Body meshes are computed with all-zero shape parameters.

a curved 3D surface into a 2D plane is always subject to this effect (similar to world maps). On the second example we see the result obtained through an automatic UV unwrapping algorithm built into Blender (open-source 3D creation suite). It splits the mesh by cutting it along seams. These seams are defined taking into account the angle differences among faces. This methodology avoids any distortion effect, but it is likely that it will produce discontinuities in the color patterns on rendering. In the last example, we obtain a more accurate result by using the minimal seams to avoid distortion. Usually, such unwrappings are performed by hand. Nonetheless, due to the size of CLOTH3D, it becomes intractable. We propose an approach to deal with this in a fully-automatic way for generic unstructured garment meshes.

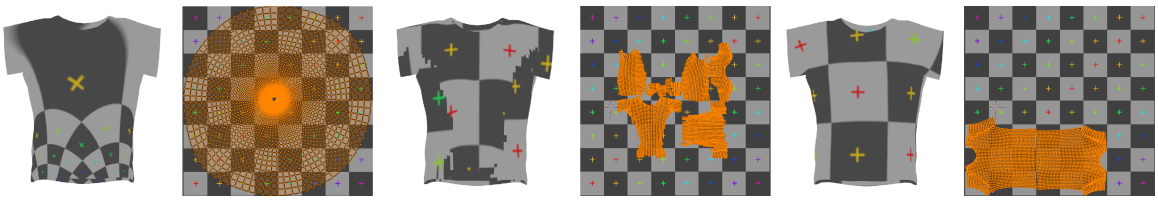

Figure 5: Different UV unwrapping approaches applied to garments. Left: without cutting the mesh, as it can be seen, it generates high distortions. Middle: automatic UV unwrapping (Blender), cuts the meshes according to angles between faces. It generates too many cuts and results in fragmented textures on rendering. Right: our proposed approach, minimizes distortions and cuts.

The idea behind it is to find the optimal seams that connect all garment boundaries with a minimal length. To do so, we first assume that all edges of a given garment have the same length. This permits finding minimal paths along a graph with BFS (Breadth First Search). For each boundary, we want to have the minimal path to each of the other

boundaries. We begin the exploration from an imaginary vertex that it is connected to all the vertices that make the starting boundary. This will implicitly compute the minimal path from all the vertices of the given boundary at the same time. Then, given boundaries $\boldsymbol{B} = \{B_0, B_1, ..., B_N\}$, we compute paths $\boldsymbol{P} = \{(B_i, B_j) \quad \forall (B_i, B_j) \in \boldsymbol{B} \times \boldsymbol{B} \mid i \neq j\}$. Finally, we compute the possible combinations of paths in P that fulfill that each boundary B_i is connected to exactly 2 different boundaries. From all of these combinations, we select the one with minimal path length. Note that for garments with only two boundaries (skirts and tube dresses), the problem is simplified, as it is only necessary to compute the minimal path between B_0 and B_1.

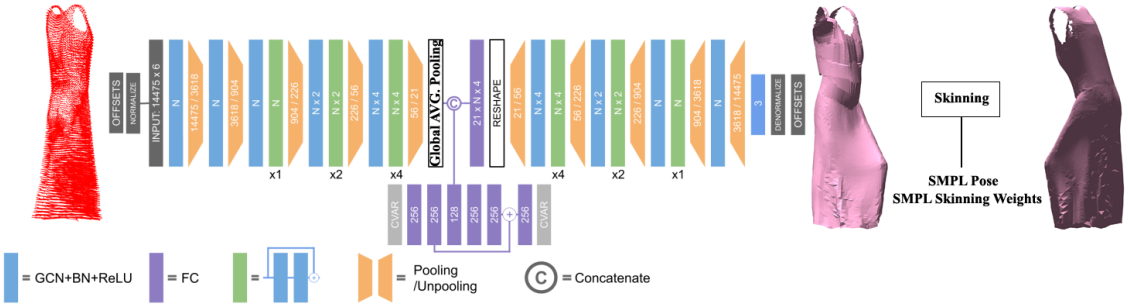

Figure 6: Track 1 architecture receiving template garment offsets and generating updated offsets. At the end garment is animated through skinning.

5. Baselines

We use Bertiche et al. (2019) architecture as the backbone of our baselines which is a graph convolutional network (GCN). Garments are first registered on top of body surface, represented by SMPL model Loper et al. (2015), in order to obtain a uniform topology among all garments. Then, body vertices that are not paired with garment vertices are masked out. In this paper, default SMPL body topology and vertices are updated to represent a new template for skirt-like garments which do not follow body topology. This architecture allows us to train a single model on the whole dataset with variable garment topology and type. Next, we explain specific modifications for each track.

5.1. 3D-to-3D garment reconstruction

This baseline is an encoder-decoder network conditioned on SMPL body shape and a sequence of pose (see Fig. 6 for the architecture pipeline). Conditioning variables (CVAR) are processed in an autoencoder network to obtain balanced conditioning features. Conditioning network is trained independent to the main encoder-decoder branch (by L1 loss) and its weights are freezed. The encoder receives garments offsets from SMPL surface in rest pose obtained by registration. The features in the last encoder layer are fused by a global average pooling and is concatenated with the middle layer of conditioning network. These features are processed through a fully connected layer and reshaped to form the decoder input tensor. The decoder output is added to SMPL body template in rest pose.

The template is selected based on garment type. If the garment is a skirt or a dress, the modified template is applied. Finally, the garment is posed through skinning with SMPL blend weights.

We apply two loss functions to train the network: L1 loss on the reconstructed 3D garment vertices and L1 loss on the surface normals computed on mesh faces. We train the network for 50 epochs with batch size of 24 including both skirt and non-skirt garments and Adam optimizer with a learning rate of $1e-4$.

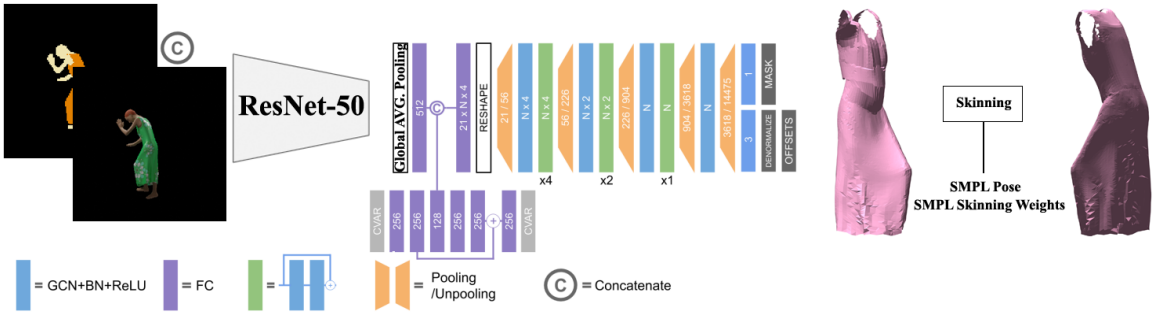

Figure 7: Track 2 architecture. Features are extracted from RGB image concatenated with estimated garment semantic labels. Similar to track 1 architecture, the network is conditioned on previously predicted body pose and shape. At the end, garment offsets and the mask are predicted.

5.2. RGB-to-3D garment reconstruction

For this task, we apply a similar baseline to Sec. 5.1 with two differences (see Fig. 7): 1) the encoder is ResNet50 architecture, and 2) the decoder outputs a garment-vs-body binary mask along with garment offsets. In this track, there is no data and meta-data (except SMPL root joint location) available at inference and all necessary information for garment reconstruction must be learned from RGB images. Necessary data to run this baseline are 1) SMPL pose and shape parameters, and 2) garment topology (skirt vs. non-skirt). For this reason we train two preprocessing networks: 1) SMPLR (Madadi et al. (2018)) to obtain SMPL pose and shape parameters as conditioning variables, and 2) PSPNet (Zhao et al. (2017)) to estimate per pixel garment semantic segments. Semantic labels are estimated for two reasons: 1) as input to the network along with RGB image helping to estimate a more accurate garment mask, and 2) to properly select the topology during inference by a hand-crafted top-down strategy. Note that RGB images are first preprocessed through cropping and color normalization (division by 255). Since SMPL root joint location is available, we can crop the images homogeneously using a fixed-size box centered on root joint and projected to camera plane. Then, the cropped image is resized to 256×256 pixels. We train this baseline with the same loss and hyperparameters as track 1, as well as L1 loss on the garment mask.

Now we explain our top-down strategy to classify garment type. We train PSPNet based on 8 semantic labels: 6 garment types, body and background. Let N_M be the number of pixels in the resized image belonging to the body and garment, $P \in \mathcal{R}^6$ be the PSPNet

average probability per garment class, $N \in \mathcal{R}^6$ be the number of estimated garment labels normalized by N_M and $S = P \odot N$ be a weighted probability vector per garment class where $\odot$ is Hadamard product. Then, if the maximum score index belongs to "dress" or "jumpsuit", we have a one layer outfit with its associated topology. Otherwise, we have a two-layer outfit (e.g. "top" plus "skirt" or "Tshirt" plus "trousers") and we check the second maximum score and compare it with the first one. For instance, if the first score is a "skirt", then the second score cannot be a "trousers". In this case, we take the third maximum score index.

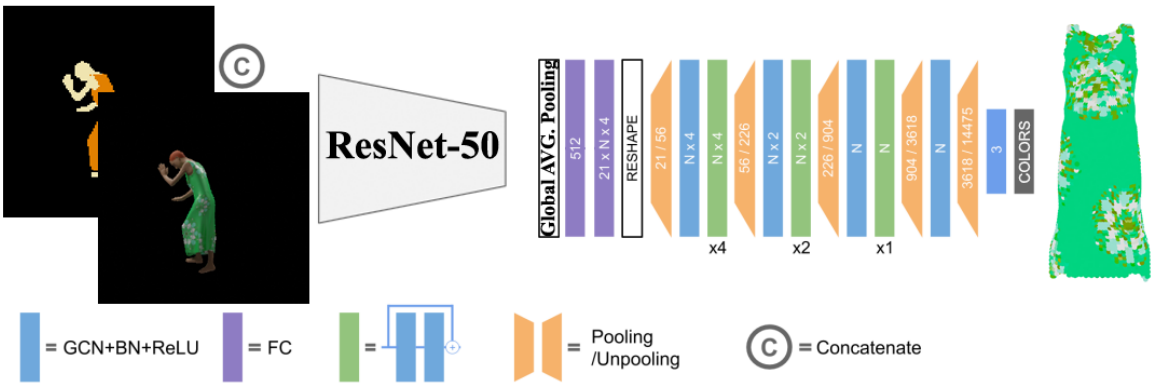

Figure 8: Track 3 architecture. Per vertex color is predicted in a similar fashion to the track 2 while no conditioning variable is fed.

5.3. RGB-to-texture reconstruction

We update track 2 baseline to estimate texture as per vertex color. To do so we remove conditioning branch and garment skinning, and instead of garment offsets we predict vertex RGB color. We also remove binary mask output, because we assume it is estimated in track 2 and can be reused in this baseline as well. We show the architecture in Fig. 8. We train the network with L2 loss. Hyperparameters are set as the other tracks.

6. Results

Since the competition has no winning solution, we only show the baseline results in this section both quantitatively (i.e. S2S error on track 1 and 2) and qualitatively. We show the results in different subsets of the test set in Tab. 3 and 4. Categorization in Tab. 3 is based on texture type (pattern or plain color), lighting conditions (indoor or outdoor), pose distribution (seen or unseen pose during training) and fabrics (leather, denim, silk and cotton). Note that regarding pose distribution, we do not have any repeating pose sequence in the whole dataset. To define unseen poses, we compute Euclidean distance between test poses and the nearest pose in training set. Test samples with a distance above a threshold (> 1.5) are considered as unseen.

In Tab. 3, without any surprise, RGB-to-3D baseline works more than 2.5 times worse than 3D-to-3D baseline. The reason is the challenges in RGB images: lighting, viewpoint

	Pattern texture	Plain color	Indoor	Outdoor	Seen pose	Unseen pose	Leather	Denim	Silk	Cotton	All
Track 1	-	-	-	-	10.6	13.7	12.1	11.4	10.1	10.6	11.3
Track 2	32.5	29.4	29.0	29.3	27.9	33.6	27.0	30.0	33.4	30.1	29.2

Table 3: S2S error (in mm) on different subsets of data.

	Top	T-shirt	Trousers	Jumpsuit	Skirt	Dress	All
Track 1	8.5	9.8	7.1	7.0	18.2	19.5	10.8
Track 2	23.9	43.3	40.6	22.1	32.8	29.3	31.3

Table 4: S2S error (in mm) per garment.

and, more importantly, loss of third dimension in projection from 3D to image plane. As expected, unseen poses have around 25% worse error than seen poses in both tracks. Interestingly, silk category has the lowest error among all fabrics in track 1, while in track 2 it has the highest error. This behavior can be explained by silk light reflecting and diffusion (as compared in Fig. 2). This is in reverse for leathers which have the lowest dynamical behavior among others. Regarding the texture type in track 2, garments with plain colors have slightly lower error than pattern textures. This is while different lighting conditions (indoor vs. outdoor) do not show any meaningful difference in error. Track 2 baseline is conditioned on body pose and shape, per pixel semantic labels and classified garment types. Our trained SMPLR on CLOTH3D++ has 68.4 mm average per joint error on test set. This is while our proposed garment classification based on PSPNet semantic labels performs very good with 0.90 F1 score on test set. Therefore, body pose has more impact on the garment reconstruction than other conditioning variables.

In Tab. 4, we analyse per garment error. While a similar pattern can be observed between both tracks, Tshirt and trousers are exceptions with surprisingly high error in track 2. One reason is a wrong predicted garment mask for these garments. In track 1, the model performs worse on Tshirt than trousers, perhaps because of a more challenging dynamic on Tshirt, specially open-front shirt. According to the data statistics in Tab. 2, jumpsuit has the largest number of frames forcing the network biasing towards it yielding the lowest error in both tracks. Although dress has the second largest number of samples, it shows a high error due to its complex dynamics (along with skirt). Interestingly, although top has a reasonable amount of samples with simple dynamic, it does not show the best performance in both tracks.

Although our baselines in track 1 and 2 show average errors as small as 11.3mm and 29.2mm, respectively, there is a huge space for improvements. This can be seen qualitatively in Fig. 9 and 10. We discuss limitations of our baselines w.r.t. these figures: 1) in track 1, registration is required as preprocessing at inference for each sequence, 2) predictions show a large amount of garment-body penetration without having a loss penalizing them during training, 3) network converges to average smooth garments and not learning low frequency details, 4) although the network is conditioned on temporal poses, the decoder can not implicitly handle dynamics, e.g. on skirts, 5) skinning is performed based on SMPL blend weights and garment-specific blend weights are not modeled which causes broken surfaces,

Figure 9: Track 1 qualitative results. Pink: ground truth, green: prediction.

Figure 10: Track 2 qualitative results. Pink: ground truth, green: prediction.

and 6) track 2 baseline can not handle multi-layer outfits at once and each garment is processed separately.

Finally, we show some qualitative samples of track 3 in Fig. 11. As one can see, our baseline produces smooth homogeneous per-vertex colors visually close to ground truth. However, it fails to predict pattern textures and over-smooths high frequency details.

7. Conclusions

In this paper, we introduced CLOTH3D++ dataset, a new large scale benchmark for 3D garment and texture reconstruction from rendered RGB images of simulated clothes. We designed the rendering conditions to have realistic images and provided a rich set of metadata for the dataset. We also provided three baseline methods for 3D garment or texture reconstruction based on graph convolutional networks applied on registered garments on top

Figure 11: Track 3 qualitative results. Left: prediction, right: ground truth.

of human body. We showed in the results of 3D garment reconstruction that although our baselines performed with low surface-to-surface error, there is a large space for qualitative (and quantitative) improvements specially w.r.t. the garment dynamics. Also, our baseline for texture reconstruction showed poor performance to generate realistic textures for patterns. This requires a further attention as well as designing metrics to evaluate realism on generated textures.

Acknowledgments

This work has been partially supported by the Spanish project PID2019-105093GB-I00 (MINECO/FEDER, UE) and CERCA Programme/Generalitat de Catalunya and partially supported by ICREA under the ICREA Academia programme, and by Amazon Research Awards ARA. We also acknowledge the support of our sponsors, NVIDIA, Facebook Reality Labs, Baidu, and ChaLearn.

References

Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *IEEE Conference on Computer Vision and Pattern Recognition*. CVPR Spotlight Paper.

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

Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2019a.

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

Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Cloth3d: Clothed 3d humans. *arXiv preprint arXiv:1912.02792*, 2019.

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

R. Danerek, E. Dibra, A. C. Öztireli, R. Ziegler, and M. Gross. Deepgarment : 3d garment shape estimation from a single image. *Computer Graphics Forum*, 36, 2017.

Anguelov Dragomir, Srinivasan Praveen, Koller Daphne, Thrun Sebastian, Rodgers Jim, and Davis James. Scape: shape completion and animation of people. *ACM Trans. Graphics*, 24, july 2005.

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

Verica Lazova, Eldar Insafutdinov, and Gerard Pons-Moll. 360-degree textures of people in clothing from a single image. *2019 International Conference on 3D Vision (3DV)*, pages 643–653, 2019.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248:1–248:16, oct 2015.

Meysam Madadi, Hugo Bertiche, and Sergio Escalera. Smplr: Deep smpl reverse for 3d human pose and shape recovery. *arXiv preprint arXiv:1812.10766*, 2018.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, October 2019.

Michael Oechsle, Lars M. Mescheder, M. Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4530–4539, 2019.

Albert Pumarola, Jordi Sanchez-Riera, Gary Choi, Alberto Sanfeliu, and Francesc Moreno-Noguer. 3dpeople: Modeling the geometry of dressed humans. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2242–2251, 2019.

Natsume Ryota, Sait Shunsuke, Huang Zeng, Chen Weikai, Ma Chongyang, Li Hao, and Morishima Shigeo. Siclope: Silhouette-Based Clothed People. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

Yongbin Sun, Ziwei Liu, Yue Wang, and Sanjay E Sarma. Im2avatar: Colorful 3d reconstruction from a single image. *arXiv preprint arXiv:1804.06375*, 2018.

S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 209–217, 2017. doi: 10.1109/CVPR.2017.30.

Gül Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. BodyNet: Volumetric inference of 3D human body shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018.

Tuanfeng Y Wang, Duygu Ceylan, Jovan Popovic, and Niloy J Mitra. Learning a shared shape space for multimodal garment design. *arXiv preprint arXiv:1806.11335*, 2018.

Y. Xu, S. Yang, W. Sun, L. Tan, K. Li, and H. Zhou. 3d virtual garment modeling from rgb images. In *2019 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 37–45, 2019. doi: 10.1109/ISMAR.2019.00-28.

Tao Yu, Zerong Zheng, Yuan Zhong, Jianhui Zhao, Qionghai Dai, Gerard Pons-Moll, and Yebin Liu. Simulcap: Single-view human performance capture with cloth simulation. *arXiv preprint arXiv:1903.06323*, 2019.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

Heming Zhu, Y. Cao, H. Jin, Weikai Chen, Donglei Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. *ArXiv*, abs/2003.12753, 2020.

Appendix A. CLOTH3D++ data format

On one hand, we have 3D data as it is in CLOTH3D. These data contain animated humans, as SMPL Loper et al. (2015) parameter sequences, and garment animation data, as *Point Cache 16*, proposed by the authors as a 16-bit float alternative for PC2 format. We also find additional metadata on each sample, namely: garment labels (t-shirt, trousers, etc.), tightness and fabric. Since these data are not part of our contribution, we would like to refer the reader to its original paper for further detail Bertiche et al. (2019). On the other hand, we have RGBA related data. Video sequences of each sample and metadata about

the color and the *world*, in a render context. More formally:

3D Data.

- Human: Subjects are based on SMPL. This metadata has been used for the generation of the 3D human models.
  - Pose sequence: SMPL pose parameters ($\mathbb{R}^{72\times\#frames}$)
  - Shape: SMPL shape parameters ($\mathbb{R}^{10}$)
  - Gender: female, male
  - Translation sequence: SMPL root joint location ($\mathbb{R}^{3\times\#frames}$). Root joint is first aligned at $(0, 0, 0)$ and later moved to its corresponding location. NOTE: SMPL does not align root joint at origin by default.
- Cloth:
  - Animation data (PC16)
- Metadata:
  - Outfit: type of garments. A subset of $\{Top, Tshirt, Trousers, Jumpsuit, Skirt, Dress\}$ with a maximum of 2 items.
  - Tightness: two-dimensional array that describes garment tightness. For details, we refer to Bertiche et al. (2019).
  - Fabric: type of fabric used for the given garment (Cotton, Silk, Denim and Leather). This has an impact in cloth simulation and rendering.

RGBA Data.

- Video
  - RGB
  - Alpha (losslessly compressed)
- Garment color
  - Type: plain color or pattern.
  - HEX Color (only for plain color)
  - Pattern as PNG (only for pattern)
- World
  - Z-rotation: both human models and garments are rotated a random angle (in radians) around Z-axis. randomly sampled Z-rotation guarantees viewpoint balance.
  - Camera location: a single 3D point, since it is static through sequences.
  - Lights
    - Type: sun or point. There can be more than one point light.
    - Power: intensity of the light (different scales for sun and point)
    - Rotation: 3D orientation in the space (only for sun)
    - Location (Only for point)