Garment3DGen: 3D Garment Stylization and Texture Generation - Supplementary

Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li,
Jovan Popovic, Rakesh Ranjan

Meta Reality Labs
nsarafianos.github.io/garment3dgen

We refer the interested reader to the supplemental video where we provide a wide variety of results ranging from image/text to 3D textured garments as well as applications of our method in downstream tasks such as physics-based cloth simulation, hand-garment interaction in VR using a headset and sketch to 3D garment reconstruction. Below we provide some additional details regarding the implementation of our key components as well as some additional ablation studies to showcase the impact of our design decisions.

Garment3DGen Details

Insights and Key Contributions: We believe that our approach provides three key insights that will be valuable to the community:

1. Mesh-based deformations provide the right properties to generate (or stylize) new garments that we can utilize for downstream tasks other than rendering.
2. A text-prompt or a single image alone cannot provide enough guidance to generate the desired garment exactly the way a user might want it
3. 3D supervisions, if done right, can provide strong enough supervision signal in order to generate the desired garments with the proper topology and structure.

Our approach builds upon these insights and introduces a novel yet simple solution to generate high-quality, physically plausible garments. As input to the method, we require only a single garment image (or alternatively, a text prompt that can generate this image using a text to image model) and a base garment template mesh. This template mesh is not required to be similar to the image guidance. For example, we demonstrate results where our method can go from a shirt to a puffer jacket, from a tank-top to a dress or even a T-shirt to a fantastical sea armor. Note that the closer the base mesh is to the target geometry, the easier the task is. For example, starting from a dress mesh to go to a shirt is a difficult task while starting from something closer to the target simplifies this problem.

Differences with Past Works: We highlight the unique and novel aspects of our method and its differences with prior works. Our deformation-based formulation is inspired by the Neural Jacobian Fields [1] work and its application to text-based deformations in TextDeformer [2]. TextDeformer however suffers

Fig. 1: Impact of the pre-trained CLIP on garment data: We disable all other supervisions and explore the impact of a pre-trained CLIP model on fashion data versus using the regular model to enforce embedding supervisions. We observe that regular CLIP embeddings result in distorted and unusable geometries regardless of whether the input is a text prompt or an image.

from some key limitations: a) it can only work for general-purpose objects and text-promts (*e.g.* turning a cow to a giraffe) and fails to capture the intricate details of garments, b) it is a severely under-constrained problem since a single text-prompt is used to guide the deformation and as such the authors enforced additional supervisions (*i.e.*, multi-view consistency, Jacobian regularization) that add marginal value to the solution. Instead our proposed formulation utilizes 3D pseudo ground-truth to supervise the deformation process and as such it does not require any additional consistency losses or Jacobian regularizations. Furthermore several past works [2, 5, 6] rely on traditional CLIP embeddings to guide their text-to-mesh tasks which provide limited signal for specific domains or categories (*e.g.* garments or humans). Until we have a 3D foundation model that can capture such intricacies well, we believe that fine-tuned domain-specific models can provide valuable guidance for text/image to 3D tasks and we provide an ablation study to showcase this in Fig. 1. Another line of work approaches this task from an image to 3D reconstruction viewpoint. Methods such as Wonder3D, Zero123++ or ZeroShape generate watertight geometries that lack fine-level details (due to the use of Marching Cubes) that require additional post-processing and manual editing (to open holes in the neck, waist and arm regions) in order to be able to be fitted to human bodies. We believe that our approach strikes the

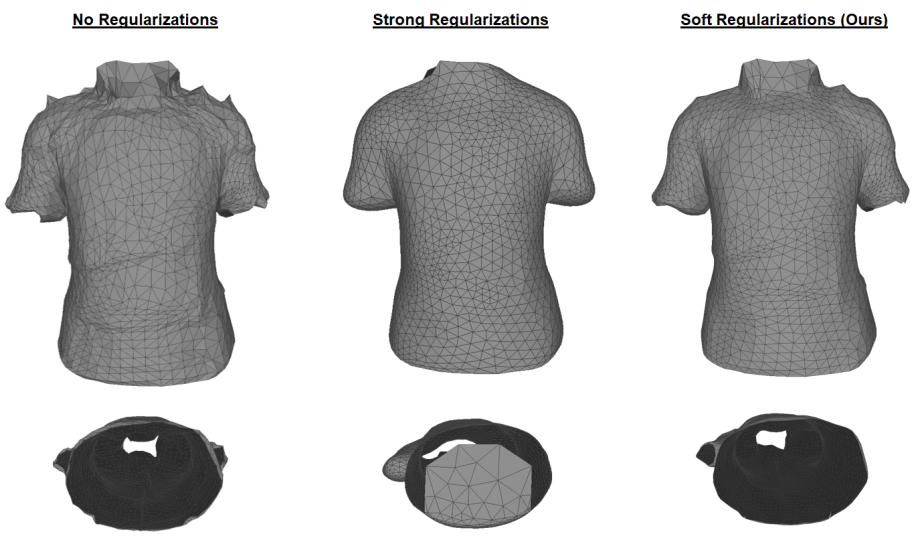

Fig. 2: Impact of regularizations on the final armor geometry: Enforcing no regularizations (Laplacian smoothing, penalization of small triangles etc.) on the output mesh results in a crisp output armor mesh with arm/body holes but its quality is not at the level required to perform physics-based simulation. On the other hand, enforcing strong regularizations results in overly smoothed meshes with closed holes. Our output strikes a good balance between capturing those fine-level details that make an armor geometry look like one yet making it suitable for downstream tasks.

right balance between reconstruction-based and deformation-based approaches. It benefits from the ability of reconstruction-based approaches to generate 3D pseudo ground-truth from a single image input that can act as a stronger supervision signal and at the same time remain in the mesh deformation space which better ensures topology preservation and output meshes that can fairly easily be fit to parametric human bodies and simulated.

Automatic View Selection: The goal of this algorithm is to automatically select the least-painted view and paint it. In this way, we can solve the 3D texture generation problem in a coarse-to-fine manner, and ensure the overall consistency. Alg. 1 provides a detailed description of the automatic view selection algorithm: given the input UV texture T with painted front and back views, there could be N candidate views. We maintain a binary mask T^B that marks the painted pixel as 1, and unpainted pixel as 0. We can select the view with the most unfilled pixels as the next view to generate the appearance, and update the binary mask T^B. This process is repeated iteratively until most of the pixels are painted, or reaching a certain iteration number.

This approach prioritized filling in the large areas first before moving on to smaller and more occluded regions.

Cloth Simulation and Material Parameter Selection: Our method produces simulation-ready meshes that can be simulated using any cloth simulator. In our examples, we use a GPU implementation of the XPBD [4] algorithm to ob-

Fig. 3: Impact of Texture Module: given the left image as a condition, the texture enhancement module enriches the details and enhances the overall image quality by effectively utilizing the powerful 2D priors.

Algorithm 1: Automatic View Selection

Input: an input mesh M_{def} with UV texture T with front and back views painted, a binary mask T^B marking the painted pixels of T, and N uniformly distributed candidate views $\{C_i\}_{i=1}^{N}$;
for *number of iterations* **do**
  Calculate the binary mask T_i^B for each view i from T^B: $\{T_i^B\}_{i=1}^{N}$;
  Select the least painted view C_j: $j \leftarrow \arg\min_{i=1}^{N} \sum T_i^B$;
  Generate the appearance image I_i and update T^B;
end

tain real-time results. Since different garments are made out of different fabrics, we manually pick material parameters to model the difference, e.g. the armor will be modeled using a higher bending and stretching stiffness compared to the other garments. Producing material parameters in conjunction with simulation-ready meshes is out of scope for this work but parameters could be automatically recovered using recent advances in differentiable simulation such as DiffAvatar [3].

Additional Ablation Studies

Supervisions: When it comes to supervisions we observed that: a) utilizing regular CLIP embeddings provides minimal supervision guidance when it comes to garments and results in poorly deformed meshes which is why we opted for a garment fine-tuned model as shown in Fig. 1, b) explicitly enforcing multi-view consistency losses is not necessary as 3D supervisions can provide better guidance, and iii) there is a trade-off between allowing for heavy garment styl-

Fig. 4: Fantastical Garments generated from text prompts or image inputs: Given a variety of text (or image) inputs along with a base mesh (tank-top, t-shirt and poncho meshes) we deform the base geometry to match the target, generate the corresponding high-fidelity texture and put everything together to render the final results from three different views.

izations/deformations and maintaining a good mesh quality that can be used later on as shown in Fig. 2. Thus we propose to use a combination of 3D supervisions to guide the deformation process to obtain an accurate 3D shape along with 2D and embedding supervisions to obtain the fine-level details of the garment that the 3D pseudo ground-truth might fail to capture. We train for ~1000 iterations with the weights of each loss described in Eq. (6) as follows: $w_{CD} = 20, w_{Lap} = 1, w_{triag} = 1, w_{2D} = 2, w_E = 4$ with the weight of W_{CD} gradually decreasing after the first 500 iterations once we have obtained a fairly accurate pose and shape of the garment to allow for the remaining of the supervisions to distill the fine-level garment details. Note that if we were to enforce strong 3D supervisions we would end up with deformed garments that would have no holes for the body, arms and head.

Texture Module: The impact of the texture enhancement module is shown in Fig. 3. The textures directly synthesized by 3D generation models tend to be low-resolution, smooth and over-simplified, which is due to the scarcity of high quality 3D training data. Thus, the texture enhancement module aims to effectively utilize the 2D priors learned from the large high-quality image dataset. After our image-conditioned image enhancement, we bring back vivid details to the texture, improving the perceptual quality.

Additional Results

In Fig. 4 we provide multi-view renders of our 3D textured garments that we generate from text prompts (first two lines) and an image guidance. From these results we gain the following insights: a) Garment3DGen works just as well with fantastical garments (armors or dresses) that are outside the regular garment distribution, b) our texture estimation module results into high-quality textures that closely match the input text prompt and c) our output geometry does not have to be similar to the input base mesh.

References

1. Aigerman, N., Gupta, K., Kim, V.G., Chaudhuri, S., Saito, J., Groueix, T.: Neural jacobian fields: Learning intrinsic mappings of arbitrary meshes. ACM Trans. Graph. **41**(4) (jul 2022) 1
2. Gao, W., Aigerman, N., Groueix, T., Kim, V., Hanocka, R.: Textdeformer: Geometry manipulation using text guidance. In: ACM SIGGRAPH 2023 Conference Proceedings. pp. 1–11 (2023) 1, 2
3. Li, Y., yu Chen, H., Larionov, E., Sarafianos, N., Matusik, W., Stuyck, T.: Diffavatar: Simulation-ready garment optimization with differentiable simulation. In: CVPR (2024) 4
4. Macklin, M., Müller, M., Chentanez, N.: Xpbd: position-based simulation of compliant constrained dynamics. In: Proceedings of the 9th International Conference on Motion in Games. pp. 49–54 (2016) 3
5. Michel, O., Bar-On, R., Liu, R., Benaim, S., Hanocka, R.: Text2mesh: Text-driven neural stylization for meshes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13492–13502 (2022) 2
6. Mohammad Khalid, N., Xie, T., Belilovsky, E., Popa, T.: Clip-mesh: Generating textured meshes from text using pretrained image-text models. In: SIGGRAPH Asia 2022 conference papers. pp. 1–8 (2022) 2