Contents

A. Contribution, Novelty, and Limitation

We reiterate our contribution, novelty, and limitations.

A.1. Contribution and Novelty

Our main contribution lies in a new direction: enabling non-professional users to create and edit 3D garment with single-view input. While existing works have made strides in reconstructing clothed humans [1, 3, 6, 7] or garment [4] from a single image, they mainly rely on optimizing predefined garment or human templates. In contrast, we target a more flexible, template-free garment reconstruction framework. Specifically, we propose to progressively synthesize depth-accurate novel view images with enhanced cross-view consistency. Moreover, our method enables single-view 3D editing, including part-based or local surface edits — capabilities that are absent in the aforementioned methods.

A.2. Scope and Limitations

As discussed in Section 5 of the main paper, our method has certain limitations. We mainly focus on garment in a rest pose. As will be shown in Section D.7, our method may struggle to accurately capture the geometry of garments in non-rest poses. With that said, this scope is a deliberate choice, as rest poses provide a consistent and intuitive baseline that aligns well with the needs of garment editing applications.

B. Ethics and Social Impacts

We focus on advancing garment digitization. We do not foresee any ethical concerns or negative societal impacts arising from our work. Our training and evaluation processes do not involve any sensitive data, human identities, or personal information. All experiments and datasets used in this study are compliant with ethical research practices. By advancing template-free garment reconstruction for non-professional users, our method avoids potential biases associated with specific body or garment templates, promoting inclusiveness in digital garment reconstruction.

C. Additional Implementation Details

In this section, we provide additional implementation details of our method omitted in the main text.

C.1. Conditional Image Generation

Our image generation model is finetuned from the Stable Zero-1-to-3 checkpoint[1]. To account for the additional projected image as input, we add 4 additional channels to the input convolution layer of the denosing UNet and initialize the weights to be zeros. The training resolution is 512×512. We train the mode on 4 NVIDIA A6000 GPUs with a total batch size of 256 for 20k iterations for 2 days.

C.2. Conditional Depth Generation

Our conditional image generation model is finetuned from the Sapiens-0.3B depth checkpoint[2]. To add the projected partial depth map as the additional condition, we add 1 extra channels to the input projection layer of the vision transformer backbone and initialize its weights to be zeros. The training resolution is 512×512. We train the model on 4 A6000 GPUs with a total batch size of 24 for 3 days.

C.3. Computational Efficiency

The inference time and memory consumption of our method are approximately 1 minute and 10 GB, respectively, on a single A6000 GPU. These values are comparable to those of most baseline methods, which have inference times ranging from 10 seconds to 1 minute.

C.4. Measures to Reduce Error Accumulation

Since our method synthesizes novel views in sequential steps, it is susceptible to error accumulation. To address

[1] https://huggingface.co/stabilityai/stable-zero123
[2] https://huggingface.co/facebook/sapiens-depth-0.3b

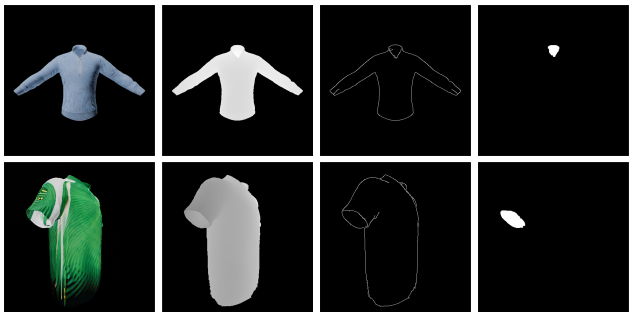

Figure S1. **Open hole detection in garments.** We note that interior regions of open holes in a garment exhibit greater depth values compared to the boundary pixels. Leveraging this observation, we propose a simple yet effective algorithm to detect open holes and exclude these regions during point cloud completion, improving the robustness of the pipeline.

this, we incorporate a series of techniques aimed at mitigating such errors and improving overall robustness.

C.4.1 Point Cloud Outlier Removal.

Depth predictions near the edges of discontinuities (with large jumps in depth values) are occasionally inaccurate, resulting in some floating points in the point cloud. To address this, we apply a classical outlier removal method at each step to eliminate these floating points, ensuring a cleaner and accurate point cloud.

C.4.2 Open Hole Detection.

We observe that depth predictions are less reliable in open-hole regions of a garment surface, such as holes in collars and sleeves. Additionally, the surface orientation derived from the estimated depth map in these areas can be reversed. These errors can propagate and lead to artifacts in subsequent steps. To address this issue, we develop a simple algorithm to detect open holes and exclude these regions during point cloud completion, improving the robustness of the pipeline.

The detection algorithm is based on the observation that the interior regions of open holes typically exhibit greater depth values compared to the boundary pixels. As shown in Figure S1, after synthesizing the completed image and depth maps from a novel viewpoint, we first detect edges in the depth map and identify connected regions enclosed by these edges using classical methods. A connected region R is classified as an open hole if more than a threshold ϵ of its boundary pixels have depth values smaller than the average depth of the region. For all our experiments, we found that ϵ can be robustly set to 0.85.

C.4.3 Clipping Distant Depth Values.

Our observations indicate that synthesized images and depth maps are more robust in regions closer to the camera compared to those farther away. At steps 3 and 4 (corresponding to azimuth angles of 120° and -120°), the entire back side of the garment is synthesized from a side view. For these steps, we only use pixels with smaller depth values for point cloud completion, disregarding pixels with larger depth values.

C.5. Scope of Single-View 3D Editing

As introduced in Section 3.3 of the main paper, GarmentCrafter enables single-view editing through a simple workflow: identify the edited 3D region, remove the original mesh in the identified area, and reconstruct the edited components. We support two types of editing operations, differentiated by their assumptions about the edited regions.

The first category is local surface editing. Given a camera viewpoint and a mask, this approach assumes that only the visible surface intersected by the camera rays corresponding to the masked pixels will be edited. Occluded surfaces are ignored, even if their mesh vertices project within the mask. To facilitate reconstruction, we remove the mesh vertices of the selected surface. Additionally, internal vertices near the external surface are also removed to account for surface thickness.

The second category, part-based editing, involves modifying a 3D garment part, including not only the "front" surface but also the "back" and "internal" surfaces within a masked region. For ease of implementation, we always use the frontal view as the editing perspective and remove all mesh vertices whose 2D projections fall within the mask.

Our editing pipeline is designed under the assumption that both the geometry and the texture will be edited. Therefore, it is not optimized for cases where (1) surface texture is modified while preserving the geometry, or (2) the geometry or pose is altered while preserving the texture.

C.6. 2D Editing Assumptions

In theory, our method is agnostic to the tools used for 2D editing. The edits can be created using deep learning-based image editing models or traditional tools like Photoshop. However, our approach requires the edits to be confined to regions specified by masks in the 2D input. Therefore, global edits such as style transfer that alters the entire image, are not recommended.

D. Additional Results and Analyses

D.1. Intermediate Results of Progressive NVS

In Figure 2 of the main paper, we showed results at one specific camera rotation step during the progressive novel

Figure S2. **Intermediate results of progressive novel view synthesis along a full camera trajectory.** From an input RGB image (top-left), GarmentCrafter progressively synthesize novel view RGB and depth maps following a zigzag camera trajectory.

view synthesis. Here, we illustrate the whole process and show the intermediate results in Figure S2.

D.2. P-NVS for Cross-view Consistency

We demonstrate the effectiveness of P-NVS on improving cross-view consistency in Figure S3. Using our method, the synthesized novel view image aligns more closely with the ground-truth projected image, indicating less inconsistency. This collaborates with our quantitative results reported in Table 2 of the main paper.

D.3. Trimming Operation in Mesh Reconstruction

We apply trimming operation after Screened Poisson surface reconstruction to preserve the correct garment topology. See Figure S4 for an example.

D.4. Comparison with SoTA NVS methods

We present additional quantitative comparisons for novel view synthesis against state-of-the-art methods (Zero-1-to-3++ [5] & MVD-Fusion [2], fine-tuned with same data). For each object in the held-out test set of 150 garment assets, we sample six camera viewpoints with an elevation of 20 degrees and evenly spaced azimuth angles covering 360 degrees. Each method takes a frontal image as input and generates six corresponding novel views, which we evaluate against ground truth images using image similarity metrics (LPIPS, PSNR, and SSIM). We also report our proposed CVCS score. Table S1 shows that our method achieves superior performance across all metrics.

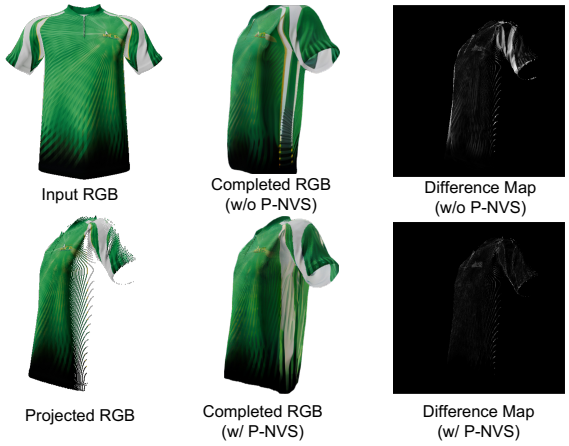

Figure S3. **Analysis of projected image conditioning.** Left: we show original input and projected RGB images. Middle: completed RGB images with and without Progressive Novel View Synthesis (P-NVS). Right: difference between completed and projected images, showing our novel view aligns more closely with the ground-truth projected RGB. Zoom-in for details.

Figure S4. Trimming operation preserves the garment topology during mesh reconstruction.

Table S1. **Quantitative comparison for novel view synthesis.** Our method outperforms all state-of-the-art novel view synthesis methods cross both image similarity and consistency metrics.

	LPIPS ↓	PSNR ↑	SSIM ↑	CVCS↑
Zero123++	0.1611	18.023	0.7979	0.8957
MVD-Fusion	0.1528	18.529	0.8026	0.9090
Ours	**0.1052**	**22.776**	**0.8557**	**0.9512**

D.5. Comparison with Garment3DGen

We provide qualitative comparison with Garment3DGen [4] on the reconstructed mesh geometry in Figure S5. Our method reconstructs 3D garments with much richer geometric details and much less inference time (1 min vs. 3 hours).

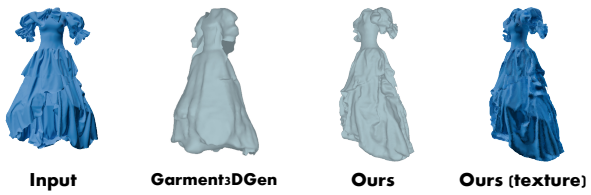

Figure S5. **Qualitative Comparison with Garment3DGen.** GarmentCrafter reconstructs garment meshes with richer details with much lower computational costs.

Table S2. **Analysis of the degree of zigzag camera trajectory.** In our experiments, we use a 60° trajectory as it provides a good balance between view coverage and efficiency. While the choice of degree slightly affects the ability to synthesize side-view garments (i.e., 90°), our analysis indicates that the overall performance is not highly sensitive to this parameter.

Degree	Appearance			Geometry
	SSIM↑	LPIPS↓	PSNR ↑	Chamfer↓
30°	0.8044	0.1675	**20.62**	0.0051
60°	**0.8066**	**0.1638**	20.62	**0.0050**
90°	0.8003	0.1709	20.19	0.0070
120°	0.8053	0.1654	20.51	0.0050

D.6. Additional Analyses and Applications

D.6.1 Degree of Zigzag Camera Trajectory

We have studied all major design choices in our pipeline in the main paper, including the effect of progressive novel view synthesis and camera trajectory. Here, we analyze the impact of the degree of Zigzag Camera Trajectory and show the results in Table S2. In our experiments, we use a 60° trajectory as it provides a good balance between view coverage and efficiency. While the choice of degree slightly affects the ability to synthesize side-view garments (i.e., 90°), our analysis indicates that the overall performance is not highly sensitive to this parameter. We do not notice any other significant hyperparameters in our framework.

D.6.2 Digitizing AI-generated Apparel

We explore the potential of combing GarmentCrafter with AI-generated garment image and show examples in Figure S9. Using a text-to-image generative model, we produce synthetic garment images and apply GarmentCrafter to digitize them. The results demonstrate the broad applicability of our method in handling diverse inputs, including AI-generated designs.

Figure S6. **Failure case.** GarmentCrafter may fail to reconstruct the garment with arbitrary poses.

D.7. Failure Cases

The focus of our work is on reconstructing and editing garments in their rest pose. Consequently, our method struggles with input images in arbitrary poses as such instances lie outside of the training data distribution. As illustrated in Figure S6, an input garment image in a non-resting pose results in the failure of our model to synthesize coherent novel view images, leading to nonsensical reconstructions.

D.8. More Qualitative Results

Reconstruction. Please see more results in Figure S7.

Editing. We provide more qualitative results in Figure S8.

References

[1] Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. Photorealistic monocular 3d reconstruction of humans wearing clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1506–1515, 2022. 1

[2] Hanzhe Hu, Zhizhuo Zhou, Varun Jampani, and Shubham Tulsiani. Mvd-fusion: Single-view 3d via depth-consistent multi-view generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9698–9707, 2024. 3

[3] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2304–2314, 2019. 1

[4] Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li, Jovan Popovic, and Rakesh Ranjan. Garment3dgen: 3d garment stylization and texture generation. In *3DV*, 2025. 1, 4

[5] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023. 3

[6] Yuliang Xiu, Jinlong Yang, Xu Cao, Dimitrios Tzionas, and Michael J Black. Econ: Explicit clothed humans optimized via normal integration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 512–523, 2023. 1

[7] Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):3170–3184, 2021. 1

Figure S7. More qualitative result on single-view 3D garment reconstruction.

Figure S8. **More results on single-view 3D garment editing.** The top row illustrates how GarmentCrafter effectively handles surface edits, even for regions with complex textures. The middle row demonstrates the capability of GarmentCrafter to support full garment changes and swaps, showcasing the potential in virtual try-on scenarios. The bottom row presents an example of removing an entire garment part.

Figure S9. **Compatibility with generative apparel.** By reconstructing both geometry and texture from synthetic garment images, GarmentCrafter demonstrates its adaptability to AI-generated designs. The results showcase the ability of GarmentCrafter to handle diverse and complex inputs, expanding its potential applications to generative fashion and virtual apparel workflows.