DreamCoser: Controllable Layered 3D Character Generation and Editing

Yi Wang*
Tianjin University
China
stardust66@tju.edu.cn

Jian Ma*
Tianjin University
China
jianma@tju.edu.cn

Zhuo Su
ByteDance China
China
suzhuo13@gmail.com

Guidong Wang
ByteDance China
China
guidong.wang@bytedance.com

Yu-Kun Lai
Cardiff University
United Kingdom
laiy4@cardiff.ac.uk

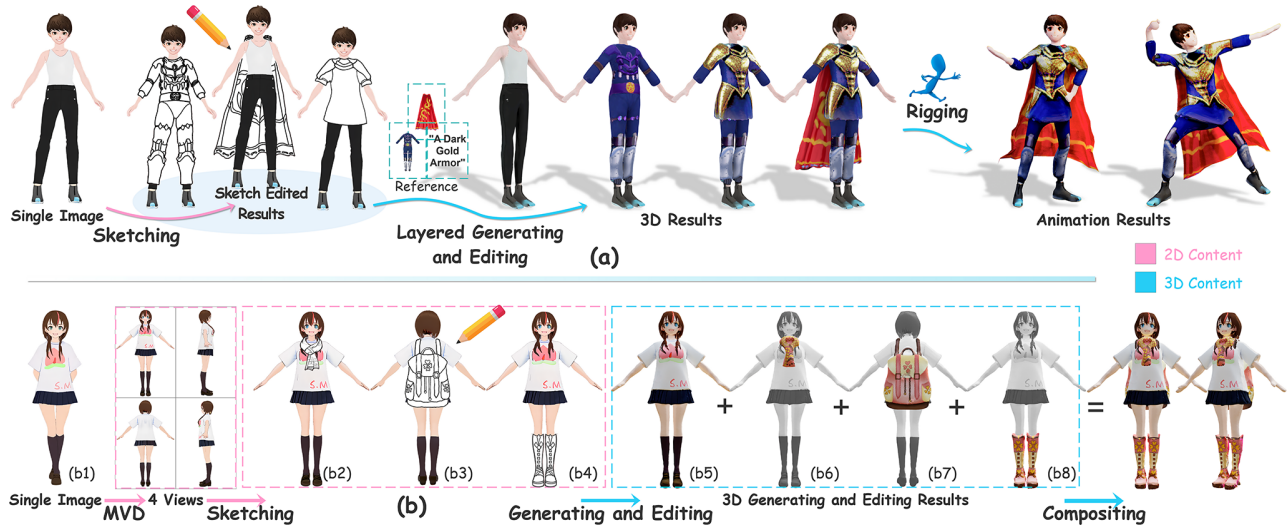Kun Li[†]
Tianjin University
China
lik@tju.edu.cn

Figure 1: Our DreamCoser method supports high-quality layered 3D character generation and editing, based on character images and sketch inputs. As shown in (a) Multi-Layer Editing: Given a single character image in an arbitrary pose, our method can generate a high-quality A-pose 3D character model. Furthermore, users can modify the initial 3D character in a layered manner via direct sketch-based editing. In (b) Part-Level Editing: We demonstrate fine-grained local modifications (e.g., adding scarf, backpack, boots) through sketch edits on multi-view images, where (b6-b8) display the edited results.

Abstract

This paper aims to controllably generate and edit layered 3D characters based on hand-drawing. Existing methods rely on global optimization or entangled representations, limiting fine-grained local editing and clothing replacement. To address this, we propose an innovative sketch-based method for layered 3D character generation and part-level editing. Our approach introduces a sketch-to-3D decoupled generation network for fine-grained layered control and a progressive upsampling module that enhances texture quality and complex geometric structures. Extensive experiments on public datasets and in-the-wild data demonstrate our method effectively generates high-quality layered 3D characters while supporting precise local editing through hand-drawn sketches. The code will be available at http://cic.tju.edu.cn/faculty/likun/projects/DreamCoser.

*Equal contribution.
[†]Corresponding author.

CCS Concepts

• **Computing methodologies → Artificial intelligence**; **Shape modeling**; **Image manipulation**.

1 Introduction

Generation of high-quality 3D characters with customizable and interchangeable clothing is highly demanded for applications in movie production, game development, and AR/VR. Specifically, techniques to achieve disentangled representation and high-quality controllable generation of 3D characters are essential to enable seamless clothing change.

However, 3D character generation methods [Peng et al. 2024] typically generate an entangled geometry globally consisting of the body and clothing from text or a single image. Therefore, they lack the ability to generate and edit 3D characters in a layered and part-level manner. Text-based methods [Liao et al. 2024] further refine Score Distillation Sampling (SDS) [Poole et al. 2023] supervision to produce high-quality 3D characters, but text inputs fail to precisely describe the local shapes, textures, and poses of the characters. Meanwhile, some methods [Long et al. 2024] combine multi-view diffusion (MVD) models with 3D perception attention modules to improve the consistency and accuracy of 3D generated results, but these methods are limited by the low-resolution multi-view images generated through MVD models, making it challenging to obtain detailed geometry and fine textures. Notably, the above-mentioned methods do not directly support layered generations of 3D content due to their entangled representations.

Our goal is to achieve controllable generation and editing of high-quality, layered 3D characters with customizable clothing, guided by anime-style images and sketches. We propose an innovative Sketch-to-3D Decoupled Generation (SDG) network. Specifically, the SDG network can directly add or modify the 3D content of the character by sketching on the initial 2D character image (including hand-drawing), and the modifications will be synchronously applied to the 3D character (Fig. 1). Furthermore, to ensure high quality of the generated 3D character, we propose a sketch-aware progressive upsampling module (PUP) for refinement of texture and geometry.

2 Method

As shown in Fig. 2, we propose a sketch-to-3D decoupled generation (SDG) network (Sec. 2.2) for sketch-based 3D character generation and editing. First, for 3D character generation, the initial character image can be used to generate a high-quality initial character model via our progressive upsampling module, PUP (Sec. 2.1). Then, for 3D character editing, sketch-based edits on the initial character image can be mapped into decoupled four-view images via a decoupled MVD (Sec. 2.2.1) module of SDG. The decoupled four views of the edited content are then input into the PUP module to generate the 3D edited content, which is combined with the initial 3D character for layered refinement via a 3D layered module (Sec. 2.2.2). Finally, the texture and geometry of the 3D layered editing results are further refined through the texture completion (Sec. 2.3.1) and the geometry-aware anti-penetration module (Sec. 2.3.2).

2.1 Progressive Upsampling (PUP) Module

2.1.1 Progressive Upsampling. To enhance the resolution of color and normal maps for MVD's four-view images while preserving multi-view consistency, we employ a sketch-aware progressive super-resolution method. First, we train a normal diffusion model using paired RGB and normal images from the dataset [VRoid 2022]. The four views V from Sec. 2.2 are upsampled to 512×512 (V') via a multi-view-aware ControlNet, then refined to 2048×2048 using [Wang et al. 2021]. Finally, the trained diffusion model predicts normal maps from V', which are then upsampled to 2048×2048 resolution using the super-resolution model [Wang et al. 2021].

2.1.2 Coarse-to-Fine Generation Strategy. Given the four-view images of the layered character, we obtain coarse 3D content from an LRM model [Hong et al. 2023] fine-tuned on [VRoid 2022]. The PUP module then refines texture and geometry using super-resolution normal maps via a masked normal loss:

$$\mathcal{L}_{\text{normal}} = \sum_k Mask_k^{sup} \otimes \left\| N_k - N_k^{sup} \right\|_2^2, \tag{1}$$

where $Mask_k^{sup}$ is the mask corresponding to a view of the final super-resolution four-view images. N_k and N_k^{sup} represent the normals of the coarse model and the predicted normals.

Finally, the overall objective of the 3D generation is as follows:

$$\mathcal{L}_{\text{gen}} = \lambda_1 \mathcal{L}_{\text{mse}} + \lambda_2 \mathcal{L}_{\text{mask}} + \lambda_3 \mathcal{L}_{\text{smooth}} + \lambda_4 \mathcal{L}_{\text{normal}}, \tag{2}$$

where $\mathcal{L}_{\text{mse}}$, $\mathcal{L}_{\text{mask}}$, and $\mathcal{L}_{\text{smooth}}$ correspond to MSE loss, mask loss, and geometric smoothness loss, respectively.

2.2 Sketch-to-3D Decoupled Generation (SDG) Network

Our SDG network aims to enable interactive layered editing of 3D characters based on sketch input. To predict the distribution of 2D sketch editing in 3D space, disentangle it, and generate it in a 3D layered manner, we introduce the SDG network, which consists of a decoupled MVD and a 3D layered module.

2.2.1 Decoupled MVD. We design a decoupled MVD that can inject noise into editable regions according to a ratio, preserve non-edited regions, and generate the four views of disentangled edited content, which differs from other MVD methods that use a uniform noise ratio and an entangled representation. First, a 3D character M_{char} is initially generated from character images or hand drawings I_{char} using our PUP module. Denote by V_{char} the four canonical views generated from input I_{char}, by V'_{char} latent representation of V_{char} in the diffusion model, by V_{char}^{edit} the four views for editing guidance, predicted by sketch editing I_{char}^{edit} applied on I_{char}, by F_{ij} the correspondence matrix between sketch edit I_{char}^{edit} and views V_{char}^{edit}, and by M_{edit} the four-view noise prediction mask. The sketch editing process employs decoupled MVD, defined as follows:
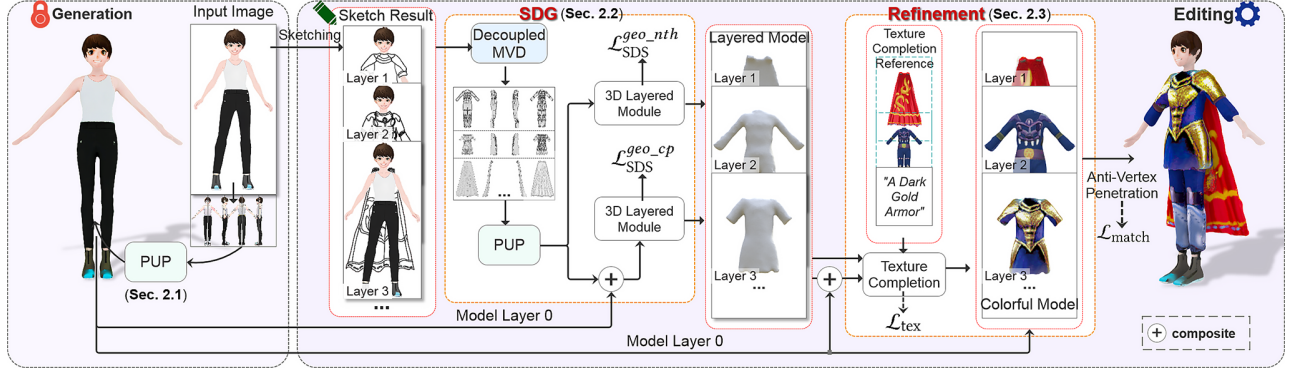
Figure 2: The overview of our method for sketch-based generation and editing. Given a character image in arbitrary pose, our method generates an A-pose 3D character model that can be modified through direct sketch-based editing on the input image in a layered manner.

$$V_{edit} = \mathcal{D}_\theta \left(V'_{char} \odot (1 - M_{edit}), \epsilon \odot M_{edit}, t \right), \quad (3a)$$

$$M_{edit} = C_{region} \left(I^{edit}_{char}, F_{ij}, V^{edit}_{char} \right), \quad (3b)$$

$$F_{ij} = \mathcal{M}_{sparse} \left(I^{edit}_{char}, V^{edit}_{char} \right), \quad \begin{cases} i \in I^{edit}_{char} \\ j \in V^{edit}_{char} \end{cases} \quad (3c)$$

where the decoupled MVD is trained on pairs of character images in arbitrary poses and their corresponding canonical four views using the dataset [VRoid 2022]. First, the correspondence matrix F_{ij} (Eq. 3c) is obtained using a sparse feature matching method [Lindenberger et al. 2023]. Then, the confidence network $C_{region}(.)$ evaluates the per-pixel feature matching scores between I^{edit}_{char} and V^{edit}_{char} under cross-view consistency constraints, and generates the noise-prediction mask M_{edit} (Eq. 3b) using adaptive normalization. Next, through diffusion denoising (Eq. 3a), identity-consistent edited four views V_{edit} matching V_{char} are obtained, where t is a time step and ϵ is the scheduled noise at t. Finally, decoupled four views V^{decoup}_{edit} are obtained from V_{edit} via $C_{region}(.)$, and initial 3D clothing M_{cloth} is generated from V^{decoup}_{edit} using the PUP module.

2.2.2 3D Layered Module. To perform geometric layering and refinement on the initial models M_{char} and M_{cloth}, we design a 3D layered module based on joint SDS loss [Poole et al. 2023] optimization. The 3D layered module effectively ensures semantic matching between the n-th clothing layer $layer_n$ and the combined results of the previous n layers $layer_{cp}$ (clothing + body), while alleviating mutual geometric interpenetration. The optimization objective is as follows:

$$\nabla_\theta \mathcal{L}^{geo\_nth}_{SDS}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}^{nth}_t; y^{nth}, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \quad (4)$$

$$\nabla_\theta \mathcal{L}^{geo\_cp}_{SDS}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}^{cp}_t; y^{cp}, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \quad (5)$$

$$\mathcal{L}_{layer} = \alpha_1 \mathcal{L}^{geo\_nth}_{SDS} + \alpha_2 \mathcal{L}^{geo\_cp}_{SDS} + \alpha_3 \mathcal{L}_{reg}, \quad (6)$$

where $\mathcal{L}^{geo\_nth}_{SDS}$ and $\mathcal{L}^{geo\_cp}_{SDS}$ are used to optimize the 3D content of $layer_n$ and $layer_{cp}$, respectively, inside out. $\mathbf{x}^{nth}$, y^{nth} and $\mathbf{x}^{cp}_t$, y^{cp} are noisy samples of the normal maps and text prompts for the 3D content of $layer_n$ and $layer_{cp}$. $\mathcal{L}_{layer}$ is the overall objective of

the layer optimization and $\mathcal{L}_{reg}$ is the loss of surface regularization. Other definitions of SDS loss are provided in [Poole et al. 2023].

2.3 Refinement of Layered Editing

To facilitate user-friendly texture editing of layered 3D content, we propose a dual-modal texture completion module that supports both single-view hand-drawing and text input. Furthermore, to prevent vertex penetration between clothing and the character body, we introduce a geometry-aware anti-penetration module.

2.3.1 Dual-Mode Texture Completion Module. We first apply a joint SDS loss to optimize the texture of layered 3D content from local to global semantics, with the following objective:

$$\nabla_\theta \mathcal{L}^{tex\_n}_{SDS}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}^n_t; y^n, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \quad (7)$$

$$\nabla_\theta \mathcal{L}^{tex\_cp}_{SDS}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}^{cp}_t; y^{cp}, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \quad (8)$$

where $\mathbf{x}_t \left(I^{nth} \right) \rightarrow \mathbf{x}^n_t$, $\mathbf{x}_t \left(I^{cp} \right) \rightarrow \mathbf{x}^{cp}_t$ represents the noise sample of the rendered image I^{nth} of the n-th clothing layer $layer_n$ and the rendered image I^{cp} of the combined content of the previous n layers $layer_{cp}$ (body + clothing), with time step t noise. y^n and y^{cp} denote the text prompts for $layer_n$ and $layer_{cp}$, which are achieved by the CLIP interrogator based on the texture reference I_{ref}. Other definitions of SDS loss are provided in [Poole et al. 2023]. The overall objective of texture completion is defined as:

$$\mathcal{L}_{tex} = \beta_1 \mathcal{L}^{tex\_n}_{SDS} + \beta_2 \mathcal{L}^{tex\_cp}_{SDS} + \beta_3 \mathcal{L}_{color} + \beta_4 \mathcal{L}_{vgg}, \quad (9)$$

where $\mathcal{L}_{color}$ is used to compute the color loss between the rendered image of the 3D content and the reference I_{ref} in the specified view $view_e$. $\mathcal{L}_{vgg}$ denotes the VGG loss used to compensate for appearance in views other than view $view_e$ with $\mathcal{L}^{tex\_n}_{SDS}$ and $\mathcal{L}^{tex\_cp}_{SDS}$.

2.3.2 Geometry-Aware Anti-Penetration Module. To match the n-th layer mesh M^n_{cloth} with the combined mesh M_{cp} (previous layers), we optimize the vertices vs_{cloth} of M^n_{cloth} using a normal offset network. Each $v_{cloth} \in vs_{cloth}$ is optimized along its normal n_{cloth}. First, v_{cloth} search for its nearest neighbors on vertices vs_{cp} of M_{cp}, with visible vertices $v^{nn}_{cloth} \in vs'_{cp} \subseteq vs_{cp}$ selected. A penalty
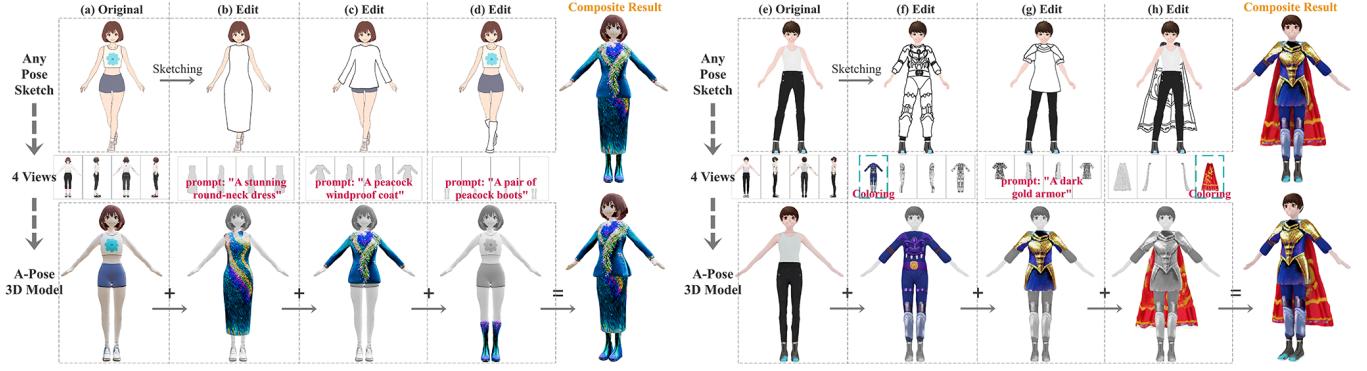
Figure 3: Multi-layer clothing editing. Our method allows users directly sketch clothing on the initial image to generate layered 3D garments for the character. Our method can take a single hand-drawn input, such as (f) and (h) or textual input, such as (b), (c), (d) and (g) for texture completion of the clothing.

is applied when the direction $\vec{d}_{cloth}$ (from v_{cloth} to v_{cloth}^{nn}) opposes n_{cloth}. Likewise, for each $v_{cp} \in vs'_{cp}$ on M_{cp}, we search for its nearest neighbor $v_{cp}^{mm} \in vs_{cloth}$ and penalize cases where the direction $\vec{d}_{cp}$ from v_{cp} to v_{cp}^{mm} aligns with the normal direction n_{cp} of vs_{cp}. The matching loss is defined as:

$$\mathcal{L}_{match} = \left(\mu_1 \cdot \vec{d}_{cp} \cdot n_{cp} - \mu_2 \cdot \vec{d}_{cloth} \cdot n_{cloth} \right) + \mu_3 \left\| \Delta v_{cloth} + \Delta v_{cp} \right\|_2^2, \tag{10}$$

where $\left\| \Delta v_{cloth} + \Delta v_{cp} \right\|_2^2$ is the displacement regularization term.

Table 1: Quantitative fidelity comparison.

Method	SSIM ↑	LIPIPS ↓	FID ↓
CRM [Wang et al. 2025]	0.64	0.39	305.46
CharacterGen [Peng et al. 2024]	0.65	0.46	300.27
Wonder3D [Long et al. 2024]	0.66	0.43	291.51
DreamCoser (Ours)	**0.67**	**0.38**	**280.06**

3 EXPERIMENTS

Results. Fig. 3 shows that our method can add diverse 3D clothing layers to 3D characters by directly drawing garments on the initial images, while also supporting texture completion guided by either single images or text. More results are shown in the supplementary material (Suppl).

Comparison. We compare our method with three state-of-the-art single-image 3D generation methods: (1) CRM [Wang et al. 2025]; (2) CharacterGen [Peng et al. 2024]; (3) Wonder3D [Long et al. 2024]. Quantitative comparisons in Tab. 1 show that our method achieves the lowest FID, indicating the best generation quality. Furthermore, Tab. 1 shows that our method achieves the highest SSIM score and the lowest LPIPS score, further validating its ability to produce detailed and accurate appearances. Additional qualitative comparisons and experimental results are provided in the Suppl.

4 Conclusion

This paper introduces DreamCoser, an innovative framework for layered generation and editing of 3D characters from sketches. Our main contribution is to address controllable layered generation and fine-grained editing of 3D characters using the SDG network. This network maps the sketch editing of the character to multi-view space, decouples them, and performs semantic and geometric layering on the generated 3D character. Additionally, we design a progressive upsampling module (PUP) to refine geometry and textures, and a dual-modal texture completion module that enhances textures of generated content using either a image or text. Experiments demonstrate that DreamCoser exhibits superior performance in high-quality generation, layered editing, and part-level control.

Acknowledgments

References

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. 2023. LRM: Large reconstruction model for single image to 3D. *arXiv preprint arXiv:2311.04400* (2023).

Tingting Liao, Hongwei Yi, Yuliang Xiu, Jiaxiang Tang, Yangyi Huang, Justus Thies, and Michael J Black. 2024. Tada! text to animatable digital avatars. In *2024 International Conference on 3D Vision (3DV)*. IEEE, 1508–1519.

Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. 2023. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 17627–17638.

Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. 2024. Wonder3D: Single image to 3D using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9970–9980.

Hao-Yang Peng, Jia-Peng Zhang, Meng-Hao Guo, Yan-Pei Cao, and Shi-Min Hu. 2024. Charactergen: Efficient 3D character generation from single images with multi-view pose canonicalization. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–13.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. 2023. DreamFusion: Text-to-3D using 2D Diffusion.

VRoid. 2022. VRoid Hub. https://vroid.com/

Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. 2021. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF international conference on computer vision*. 1905–1914.

Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. 2025. Crm: Single image to 3D textured mesh with convolutional reconstruction model. In *European Conference on Computer Vision*. Springer, 57–74.