InterCoser: Interactive 3D Character Creation with Disentangled Fine-Grained Features

Yi Wang[1,2*], Jian Ma[1*], Zhuo Su[3†], Guidong Wang[3], Jingyu Yang[1], Yu-Kun Lai[4], Kun Li[1‡]

[1]Tianjin University,
[2]Changzhou Institute of Technology,
[3]ByteDance China,
[4]Cardiff University
{stardust66, jianma}@tju.edu.cn, suzhuo13@gmail.com, guidong.wang@bytedance.com, yjy@tju.edu.cn, laiy4@cardiff.ac.uk, lik@tju.edu.cn

Abstract

This paper aims to interactively generate and edit disentangled 3D characters based on precise user instructions. Existing methods generate and edit 3D characters via rough and simple editing guidance and entangled representations, making it difficult to achieve precise and comprehensive control over fine-grained local editing and free clothing transfer for characters. To enable accurate and intuitive control over the generation and editing of high-quality 3D characters with freely interchangeable clothing, we propose a novel user-interactive approach for disentangled 3D character creation. Specifically, to achieve precise control over 3D character generation and editing, we introduce two user-friendly interaction approaches: a sketch-based layered character generation/editing method, which supports clothing transfer; and a 3D-proxy-based part-level editing method, enabling fine-grained disentangled editing. To enhance 3D character quality, we propose a 3D Gaussian reconstruction strategy guided by geometric priors, ensuring that 3D characters exhibit detailed local geometry and smooth global surfaces. Extensive experiments on both public datasets and in-the-wild data demonstrate that our approach not only generates high-quality disentangled 3D characters but also supports precise and fine-grained editing through user interaction.

Code —
http://cic.tju.edu.cn/faculty/likun/projects/InterCoser

Introduction

In the fast-paced domains of AIGC, gaming, and AR/VR, creating user-friendly tools for 3D character modeling and editing carries substantial practical significance. A core challenge lies in enabling users to interactively and controllably generate and edit high-quality 3D characters with disentangled fine-grained representations, which is crucial for applications like virtual wardrobe customization and granular character design workflows. In this paper, we propose an innovative approach to the precise and controllable generation and editing of disentangled 3D character representations through direct user interaction, as illustrated in Fig. 1.

Existing 3D character generation and editing methods struggle with three key limitations: inability to precisely control character generation and editing, entanglement of clothing and body representations limiting customization, and generation quality issues due to the lack of optimization constraints. Firstly, although 3D character generation methods (Cao et al. 2024; Kolotouros et al. 2023; Zhang et al. 2024a,c) support coarse pose/shape editing using pose conditions or SMPL priors, they lack fine-grained part-level control. Although text-based character generation methods (Liao et al. 2024; Gong et al. 2024; Xue et al. 2024) enable limited editing of characters via textual prompts, the semantics of the text cannot precisely localize editable regions, making it difficult to convey complex 3D editing instructions. Secondly, current approaches (Chen et al. 2023; Peng et al. 2024) can only produce results with entangled clothing and body. Although the method (He et al. 2025) achieves separated body and clothing through 3D semantic disentanglement, its clothing diversity remains limited by predefined categories in the dataset and fails to edit unrestricted categories. Third, recent methods (Tang et al. 2024; Li et al. 2024; Zou et al. 2024) have improved 3D appearance quality by combining Multi-View Diffusion (MVD) and Triplane representations with Gaussian Splatting (GS) (Kerbl et al. 2023). These approaches, even when using some 3D priors (e.g., SMPL) or surface constraints, still exhibit artifacts due to insufficient geometric regularization.

To address the aforementioned issues, we propose InterCoser, an innovative method designed to achieve disentangled generation and editing of high-quality 3D characters through user interaction. To achieve this, we introduce two user-friendly interaction approaches: 1) a sketch-based (including hand-drawing) layered character interactive generation and editing approach, and 2) a 3D-proxy-based part-level disentangled editing method. First, to enable precise control over character generation and editing via sketch interactions in a layered manner, and to achieve clothing transfer between differently shaped bodies, we propose the Sketch-to-3D Decoupled Matching (SDM) Network. The

*These authors contributed equally.
†Project Lead.
‡Corresponding author

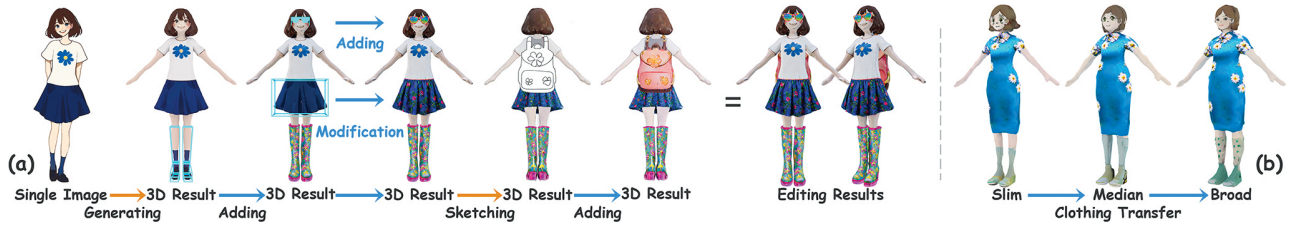

Figure 1: Our InterCoser method enables high-quality disentangled 3D characters generation and editing through user interaction. As shown in (a), our framework first generates an initial 3D character in A-pose from a single image (in arbitrary poses), then enables layered, part-level editing via sketches and 3D proxies. (b) demonstrates that our method supports transferring clothing layers between human bodies of different shapes.

SDM network takes character edits (e.g., clothing drawn by users on character images) in arbitrary poses as input, generates four standard-pose views of the edited content via a decoupled MVD, and then produces initial clothing from these views using a fine-tuned LRM model (Hong et al. 2023). Next, our novel 3D Matching Module enables both layer-wise optimization of initial clothing and clothing transfer between differently shaped bodies through layered geometry and semantic matching optimization. Finally, the layered clothing is further refined via our geometry-prior-guided 3D Gaussian reconstruction network. The SDM network enables simultaneous editing from sketches to 3D content, allowing precise interactive editing of character clothing in a layered manner.

To achieve finer-grained part-level editing of 3D characters using intuitive 3D proxies with disentangled representations, we propose an interactive GS-based editing strategy. The 3D proxy is composed of coarse geometric primitives. Unlike existing Gaussian Splatting (GS) editing methods (Wang et al. 2024b; Chen et al. 2024b), which only supervise the local semantics of edited content, our proposed strategy employs a multi-branch SDS (Score Distillation Sampling) loss combined with a Gaussian Splatting object ID (GS Object ID) mechanism to jointly optimize local edits for global character semantic consistency. Among them, the object ID of each Gaussian point is assigned through volumetric selection by the 3D proxy. Meanwhile, to enhance the quality of GS editing, we employ a progressive GS attribute decay strategy and adaptive interactive semantic loss. The GS attribute decay strategy stabilizes early edits, while allowing flexibility for new points by gradually increasing attribute regularization based on generation order. Experiments show that this interactive editing strategy can generate fine-grained, high-quality 3D disentangled content aligned with character semantics, such as glasses and accessories.

To further refine the quality of 3D characters in a layered manner while enabling topology-free editing capability, we propose a novel geometry-prior-guided 3D Gaussian reconstruction strategy. We place all GS components on the mesh layer as convex combinations of face vertices and supervise GS normals using a fine-tuned model (Bae and Davison 2024). Experiments conducted on public datasets (VRoid 2022) and in-the-wild datasets demonstrate that our method not only generates high-quality, representation-disentangled 3D characters, but also enables convenient interactive and controllable character editing.

In summary, our paper makes the following key contributions:

- We propose a novel interactive 3D character generation and editing framework—InterCoser, which enables precise control of 3D character generation and editing via user instructions, using disentangled character representations. Extensive experiments show our method enables layered, part-level editing while generating high-quality GS-based 3D characters.
- We introduce the Sketch-to-3D Decoupled Matching (SDM) Network, which can interactively generate and edit layered dressed characters based on sketches, and supports the transfer of layered clothing across different body shapes.
- We design a disentangled interactive editing strategy based on Gaussian Splatting with intuitive 3D proxies, enabling fine-grained part-level manipulation of 3D characters.
- We introduce a geometry-prior-guided 3D Gaussian reconstruction strategy that ensures detailed local geometry while maintaining global surface smoothness for layered character models.

Related Work

3D Character Generation and Editing. Diffusion-based text-to-3D generation methods (Ren et al. 2023; Cao et al. 2024; Zhang et al. 2024a; Wang et al. 2024a) leverage score distillation or LLM integration, while skeletal/Gaussian-based approaches (Hu, Hong, and Liu 2024; Pan et al. 2024; Guo et al. 2025; Zheng et al. 2024; Peng et al. 2025) utilize geometric priors. Single-image reconstruction combines implicit diffusion (Kolotouros et al. 2024; Zhang et al. 2024b; Ho et al. 2024) with explicit 3D guidance (Cao et al. 2023; Zhou et al. 2024; Xue et al. 2024), whereas sparse-view transformers (Peng et al. 2024; Chen et al. 2023; Men et al. 2024; Jiang et al. 2023) address stylized domains. However, the above 3D character generation methods do not support disentangled representation generation, whereas our approach enables layered and part-level decoupled generation and editing. Although STDGEN (He et al. 2025) achieves component-wise generation, it loses geometric details with sparse semantic segmentation, and the precomputed textures

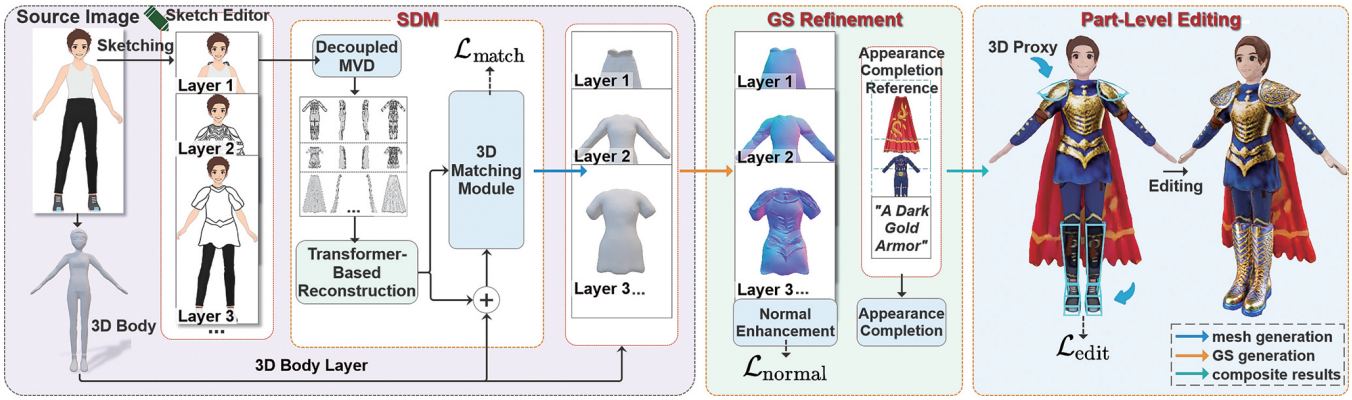

Figure 2: **Overview of our InterCoser.** We propose an interactive framework for 3D character generation and editing with disentangled representation. The framework first generates an initial 3D character model in A-pose from a single input image of a character in an arbitrary pose. Users can then perform layered editing through sketch-based interactions. Next, the system converts the layered mesh into a refined Gaussian Splatting (GS) representation, leveraging geometric priors. Finally, the GS-based layered character can be further edited through fine-grained, part-level interactive manipulation.

in DAGSM (Zhuang et al. 2025) constrain fine-grained local editing, despite its clothing separation.

3D Content Editing. For text-guided 3D editing, CustomNeRF (He et al. 2024) introduces foreground-aware editing via Local-Global Iterative Editing; DreamEditor (Zhuang et al. 2023) pioneers semantic-driven NeRF optimization for geometric edits, and TipEditor (Zhuang et al. 2024) leverages 3D GS for alignment accuracy in bounded edits. Recent advances like methods (Qu et al. 2025; Luo et al. 2025) enable direct 3D manipulation via control points, while MVDrag3D (Chen et al. 2024a) achieves multi-view consistent editing via diffusion priors. For sketch-based editing, methods (Mikaeili et al. 2023; Liu et al. 2024a; Zang et al. 2024) enable 3D creation from sketches, and interactive methods like Progressive3D (Cheng et al. 2023) enable multi-step localized editing via coarse selection, while Interactive3D (Dong et al. 2024) supports drag-and-drop operations. However, existing methods face three key limitations: text-based approaches lack precise region localization, sketch-based systems require either multiple inputs or manual 3D guidance, and interactive tools rely on coarse selection and drag-and-drop operations. Our method overcomes these issues by enabling precise editable region definition via single-sketch input with intuitive 3D proxies, while preserving superior geometric fidelity. We summarize the main differences between our work and related work in Tab. 1.

Method	ML	DR	LE	PE
LGM (Tang et al. 2024)	✗	✗	✗	✗
CharacterGen (Peng et al. 2024)	✗	✗	✗	✗
SKED (Mikaeili et al. 2023)	✗	✗	✗	✓
SketchDream (Liu et al. 2024a)	✗	✗	✗	✓
STDGEN (He et al. 2025)	✗	✓	✗	✓
Ours	✓	✓	✓	✓

Table 1: Comparison with state-of-the-art (SoTA) single-image-based 3D generation and editing methods. Abbreviations: ML (Multi-Layer Generation), DR (Disentangled Results), LE (Layer-wise Editing), PE (Part-level Editing).

Method

As shown in Fig. 2, we propose an interactive 3D character generation framework for creating and editing 3D characters in a disentangled representation manner. To achieve this, we introduce a Sketch-to-3D Decoupled Matching (SDM) network during the sketch-based layered generation and editing stage. This network maps arbitrarily posed character images and their corresponding sketch edits into a four-view A-pose representation, generating an initial layered 3D dressed character and supporting clothing transfer. Next, in the 3D GS Refinement stage, we reconstruct a high-fidelity 3D character with detailed local geometry and smooth surfaces based on the layered 3D mesh, using a geometry-prior-guided 3D GS reconstruction strategy. Finally, to enable finer-grained part-level editing of the character in a disentangled manner, we introduce a GS-based interactive editing strategy for part-level manipulation.

Sketch-to-3D Decoupled Matching (SDM) Network

The SDM network aims to interactively generate layered 3D characters with editable clothing based on sketch-based input. Existing text/image-guided 3D character generation lacks precise control over generation and editing due to inadequate pose/structure capture in text inputs, missing image-based editing cues, and entangled representations blocking clothing transfer. To address these challenges, we design the SDM network.

Interactive Sketch-Based Editing. The SDM network supports the generation of 3D characters M_{char} from character images or hand drawings I_{char}, by fine-tuning the LRM model (Hong et al. 2023) in the dataset (VRoid 2022). Furthermore, to enable interactive editing of M_{char} via

sketch-based edits I^{edit}_{char} applied to character image I_{char} in arbitrary poses, we design a decoupled MVD that processes I^{edit}_{char} to generate four standard-pose views V^{edit}_{char} of the edited content. 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 F_{ij} the correspondence matrix between sketch edit I^{edit}_{char} and views V_{char}, and by M_{edit} the four-view noise prediction mask. The process is defined as follows:

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

$$M_{edit} = \mathcal{C}_{region} \left(I^{edit}_{char}, F_{ij}, V_{char} \right), \quad \text{(1b)}$$

$$F_{ij} = \mathcal{M}_{sparse} \left(I^{edit}_{char}, V_{char} \right), \quad \begin{cases} i \in I^{edit}_{char} \\ j \in V_{char} \end{cases} \quad \text{(1c)}$$

where the decoupled MVD is trained on pairs of single view character images in arbitrary poses and their corresponding canonical four views using the dataset (VRoid 2022). First, correspondence matrix F_{ij} (Eq. 1c) is obtained using a sparse feature matching method (Lindenberger, Sarlin, and Pollefeys 2023). Then, noise prediction mask M_{edit} (Eq. 1b) is computed through a confidence network $\mathcal{C}_{region}(.)$.

Next, through diffusion denoising (Eq. 1a), identity-consistent edited four views V^{edit}_{char} matching V_{char} are obtained, where t is a time step and ϵ is the scheduled noise at t. Unlike other MVD methods (Hu et al. 2024; Liu et al. 2024b; Melas-Kyriazi et al. 2024) that apply a uniform noise injection rate, we use M_{edit} to control noise ϵ, ensuring that noise is only injected into editable regions. Finally, decoupled four views V^{edit}_{decoup} are obtained from V^{edit}_{char} via F_{ij}, and initial clothing M_{cloth} is generated from V^{edit}_{decoup} using the fine-tuned LRM.

3D Matching Module. Then, to optimize initial clothing or accessories M_{cloth} in a layer-wise manner and match it with the initial character model M_{char}, we design a novel 3D Matching Module with dual optimization stages for geometric and semantic matching.

First, to make the n-th layer mesh M^n_{cloth} geometrically match the combined mesh M_{cp} formed by the previous n layers (clothing + body), we use a normal offset prediction network to optimize the vertices vs_{cloth} of M^n_{cloth} to prevent vs_{cloth} from penetrating the combined mesh M_{cp}. Specifically, the position of the vertices vs_{cloth} is optimized along their normal direction n_{cloth}. For each vertex $v_{cloth} \in vs_{cloth}$, its nearest neighboring vertices are searched on the mesh M_{cp}, and the view-visible vertices $v^{nn}_{cloth} \in vs'_{cp} \subseteq vs_{cp}$ are selected. Meanwhile, a penalty term is introduced when the normalized direction $\vec{d}_{cloth}$ from v_{cloth} to its corresponding v^{nn}_{cloth} opposes the direction of n_{cloth}. Similarly, for each vertex $v_{cp} \in vs'_{cp}$, its corresponding nearest neighbor $v^{mm}_{cp} \in vs_{cloth}$ is located on M^n_{cloth}, and a penalty is applied if the normalized direction $\vec{d}_{cp}$ from v_{cp} to v^{mm}_{cp} aligns with the normal direction n_{cp} of v_{cp}. Finally, the matching loss function is defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{match}} = & \left(\lambda_{cp} \cdot \vec{d}_{cp} \cdot n_{cp} - \lambda_{cloth} \cdot \vec{d}_{cloth} \cdot n_{cloth} \right) \\ & + \lambda_{reg} \left\| \Delta v_{cloth} + \Delta v_{cp} \right\|_2^2, \end{aligned} \quad (2)$$

where $\left\| \Delta v_{cloth} + \Delta v_{cp} \right\|_2^2$ is the displacement regularization term for v_{cloth} and v_{cp}. $\lambda_{cp}, \lambda_{cloth}$, and λ_{reg} are the weight attributes of each loss.

Furthermore, to enable fine-grained semantic alignment between M^n_{cloth} and M_{cp}, we introduce a semantic matching loss into the 3D Matching Module. The optimization objective is as follows:

$$\nabla_\theta \mathcal{L}^{cloth}_{\text{SDS}}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}^{cl}_t; y^{cl}, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \quad (3)$$

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

where $\mathcal{L}^{cloth}_{\text{SDS}}$ and $\mathcal{L}^{cp}_{\text{SDS}}$ are used to optimize the n-th clothing layer for semantic alignment with the previous n layers. $\mathbf{x}^{cl}_t$, y^{cl} and $\mathbf{x}^{cp}_t$, y^{cp} denote noisy samples of normal maps and text prompts corresponding to the 3D content of the n-th clothing layer and the previous n layers, respectively. See preliminary of Suppl. for other SDS definitions. In addition, the 3D Matching Module supports transferring clothing layers across characters with different body shapes, as illustrated in Fig. 1(b).

The overall objective of sketch-based layered generation and editing is as follows:

$$\mathcal{L}_{\text{layer}} = \lambda_{match} \mathcal{L}_{\text{match}} + \lambda_{cloth} \mathcal{L}^{cloth}_{\text{SDS}} + \lambda_{cp} \mathcal{L}^{cp}_{\text{SDS}}, \quad (5)$$

where $\lambda_{match}, \lambda_{cloth}$, and λ_{cp} are the weights of loss terms.

3D Gaussian Refinement

To address the lack of surface constraints during the optimization of GS-based generation methods, which often leads to poor surface reconstruction quality, we design a geometry-prior-guided 3D Gaussian reconstruction strategy. Moreover, the strategy ensures that our method achieves high-fidelity appearance and topology-free editing capabilities based on Gaussian Splatting (GS).

First, to ensure proper initialization of the Gaussian distributions, we place the Gaussian components of each reconstruction layer on the surface of the corresponding initial mesh layer. For each triangle face V on the mesh, we represent the Gaussian mean as a convex combination of the vertices of V.

Second, to make the GS distributions better conform to the geometry, we optimize the GS distributions to become flattened, using the smallest scaling axis as an approximate normal vector. Specifically, we compute the rotation matrix R and the scale vector $\boldsymbol{s}(s_1, s_2, s_3)$ from the GS quaternion, and define the geometric normal of the Gaussian as:

$$n_i = R \cdot \text{OneHot}(\arg\min(s_1, s_2, s_3)), \text{OneHot}(.) \in \mathbb{R}^3. \quad (6)$$

The optimization objective to minimize the smallest scaling axis of $\boldsymbol{s}$ is:

$$\mathcal{L}_{\text{scale}} = \sum_i \left\| \arg\min \left(s_{i,1}, s_{i,2}, s_{i,3} \right) \right\|_1 \quad (7)$$

Meanwhile, the normals are composited by alpha, and the estimated normal at each rasterized pixel is computed as:

$$\hat{\mathbf{N}} = \sum_i n_i \alpha_i T_i, \quad (8)$$

where α_i and T_i denote opacity value and cumulative transmittance, respectively. To obtain smoother normal estimations, we supervise the predicted normals $\hat{\mathbf{N}}$ using the normal map $\mathbf{N}$ predicted by model (Bae and Davison 2024), with the following optimization objective:

$$\mathcal{L}_{\hat{\mathbf{N}}} = \frac{1}{|\hat{\mathbf{N}}|} \sum \|\hat{\mathbf{N}} - \mathbf{N}\|_1. \tag{9}$$

Additionally, we apply a regularization loss on the gradient of the predicted normals to enforce smoothness across neighboring pixels:

$$\mathcal{L}^{reg}_{\text{normal}} = \sum_{m,n} \left(\|\nabla_m \hat{\mathbf{N}}_{m,n}\|_1 + \|\nabla_n \hat{\mathbf{N}}_{m,n}\|_1 \right), \tag{10}$$

where $\hat{\mathbf{N}}_{mn}$ denotes the estimated normal in pixel (m, n) and ∇ represents the finite difference operator computed by convolving with $[-1, 1]$ along the m axis and its transpose $[-1, 1]^\top$ along the n axis. The overall objective of 3D Gaussian refinement is as follows:

$$\mathcal{L}_{\text{normal}} = \lambda_{scale}\mathcal{L}_{\text{scale}} + \lambda_{\mathbf{N}}\mathcal{L}_{\hat{\mathbf{N}}} + \lambda^{reg}_n \mathcal{L}^{reg}_{\text{normal}}, \tag{11}$$

where λ_{scale}, $\lambda_{\mathbf{N}}$, and λ^{reg}_n are the weight attributes of each loss. Finally, the appearance of the GS reconstruction is refined and completed using a pre-trained appearance completion module (refer to Supplementary Material).

Part-Level Interactive Editing

Finally, to enable finer-grained part-level editing of the character, we introduce a GS-based disentangled interactive editing strategy.

Local-to-Global Semantic Consistency. To ensure that local editing regions remain semantically consistent with the overall character, the proposed strategy employs a local-to-global semantic optimization combined with the GS Object ID to optimize the local editing region, unlike other methods that constrain only local semantics. The optimization objective is as follows:

$$\nabla_\theta \mathcal{L}^{local}_{\text{SDS}}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(M_l \otimes \mathbf{x}_t; y^l, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \tag{12}$$

$$\nabla_\theta \mathcal{L}^{global}_{\text{SDS}}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon} \left[\omega(t) \left(\hat{\epsilon}_\phi \left(\mathbf{x}_t; y^g, t \right) - \epsilon \right) \frac{\partial \mathbf{x}}{\partial \theta} \right], \tag{13}$$

where $\mathcal{L}^{local}_{\text{SDS}}$ and $\mathcal{L}^{global}_{\text{SDS}}$ are used to jointly optimize local character editing while maintaining semantic consistency with the overall character. M_l denotes the GS mask for the locally edited region, obtained by using the Object ID. The Object ID of each Gaussian point is determined by the volume selection of the initial 3D proxy. $\mathbf{x}_t$ denotes noisy samples of the character image. y^l and y^g refer to the text prompts for the local and global regions of the character, respectively. See preliminary of Suppl. for other SDS definitions.

Enhancing GS Editing Quality. Meanwhile, to improve the quality of generated edits, we employ a progressive GS attribute decay strategy and an adaptive interactive semantic loss. The stochastic and discrete GS optimization, without hierarchical networks' memory capacity, often causes unstable training convergence. We introduce a GS attribute decay strategy to consolidate early GS editing results, while allowing new GS points a degree of flexibility. The loss of regularization on the GS attributes is defined as follows:

$$\mathcal{L}^{reg}_{\text{p}} = \sum_{i=0}^{k} \lambda_i \|p_i - \hat{p}_i\|_2^2, \tag{14}$$

where k denotes the total number of Gaussians and p_i denotes a certain property of Gaussian points. $\hat{p}_i$ refers to the historical value of p_i from the previous optimization iteration. λ_i denotes the regularization strength applied to each Gaussian property. The overall objective of part-level interactive editing is as follows:

$$\mathcal{L}_{\text{edit}} = \lambda_{loc}\mathcal{L}^{local}_{\text{SDS}} + \lambda_{glob}\mathcal{L}^{global}_{\text{SDS}} + \lambda^{reg}_p \mathcal{L}^{reg}_{\text{p}}, \tag{15}$$

where λ_{loc}, λ_{glob}, and λ^{reg}_p are the weight attributes of each loss.

Experiments

Implementation Details

Training Details. We utilize the VRoid dataset (VRoid 2022), applying a stringent filtering process to exclude non-human characters and low-quality data. This results in a final selection of 15,000 high-quality character samples, split into training and test sets at a 50:1 ratio. For our SDM network, we adopt the Stable Diffusion 2.1 model as the base architecture. A normal prediction model is trained based on the model (Bae and Davison 2024). The normal prediction model takes an RGB image of the character as input and generates the corresponding character normal map as output. The stages of initial 3D character generation, single-layered editing, and part-level Gaussian Splatting editing take approximately 1/2/2 min (speed-priority) and 1/5/5 min (quality-priority), respectively, on a single desktop GPU with 24GB memory.

Hyperparameters. Initial model generation: 3k-step DMTet optimization; Layered editing: 2.5k-step editing optimization, with $1:5$ alternate training (n-th layer vs. first n layers); Gaussian refinement: 10k-step geometry optimization and part-level editing: 2.5k-step editing optimization. All four stages use the AdamW optimizer, with learning rates of 0.001, 0.001, 0.001 and 0.01, respectively.

Results

Notably, our method can not only generate high-quality layered 3D dressed characters in any pose, but also allows for fine-grained interactive editing of the layered characters using a convenient sketch and 3D proxy. As shown in Fig. 3, our method enables intuitive editing of the initial 3D character through simple doodles or hand-drawn sketches in a layered and input-consistent manner. Moreover, by combining our proposed joint local-global semantic optimization with

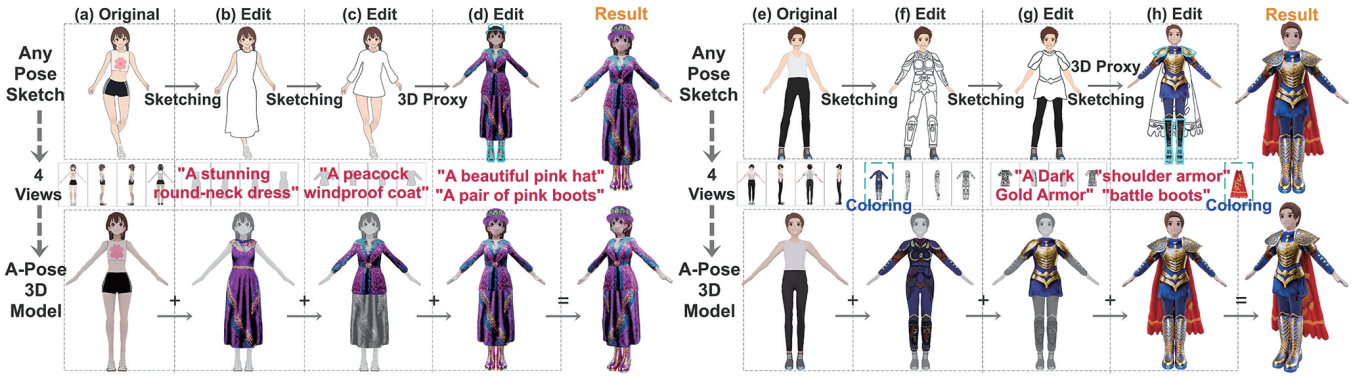

Figure 3: **Interactive 3D Character Generation and Editing Results.** Our method supports both layered editing of 3D characters through direct clothing sketching on the initial character image, such as (b), (c), (f) and (g), and fine-grained part-level editing via simple 3D proxy specification, such as (d) and (h).

the GS Object ID, the fine-grained, high-quality editing results can be achieved, such as localized modifications to the boots and armor in Figs. 3(d),(h) and the glasses and skirt in Fig. 1(a). Importantly, these edits are fully decoupled. Specifically, our SDM network supports the free transfer of layered-generated clothing between human bodies of different shapes, which greatly facilitates virtual outfit changes, as demonstrated in Fig. 1(b). Additional generation results can be found in the supplementary materials and demo videos.

Comparison

We compare our generation method with three state-of-the-art (SoTA) single-image-based 3D generation methods: (1) LGM (Tang et al. 2024), which uses a large multi-view Gaussian model to generate 3D models from text prompts or single-view images; (2) CharacterGen (Peng et al. 2024), which uses an image-conditioned diffusion model to generate 3D content from a single character sketch; and (3) STDGEN (He et al. 2025), which employs a semantic-aware large reconstruction model to generate semantically decomposed 3D characters from single images.

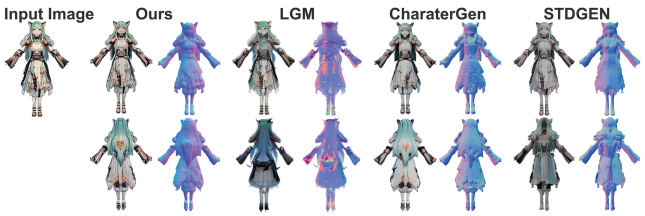

Figure 4: Qualitative comparison of single-image-based generation methods (For a detailed view, please zoom in.)

Qualitative Results. To compare under a unified posture, we use a single character image in A-pose as input for a qualitative comparison between our generation method and SoTA methods (Tang et al. 2024; Peng et al. 2024; He et al. 2025). Fig. 4 shows that our generation results outperform SoTA results (see Supp. for more comparisons). LGM lacks fine textures and smooth geometric structures due to its lightweight asymmetric U-Net architecture, which sacrifices some texture details. CharacterGen loses local geometry such as hair or clothing, despite its introduction of multi-view pose normalization to improve handling of complex poses. While STDGEN uses multi-view normal maps for geometry, its 3D segmentation from sparse 2D representations yields geometric artifacts. In contrast, our method generates 3D results with fine geometric details and smooth surfaces through our geometry-prior-based reconstruction strategy.

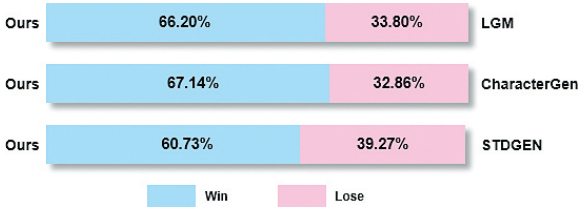

Figure 5: User preference comparison to single-image-based generation methods.

Method	SSIM ↑	LIPIPS ↓	FID ↓
LGM (Tang et al. 2024)	0.5632	0.3176	231.56
CharacterGen (Peng et al. 2024)	0.5525	0.4058	205.68
STDGEN (He et al. 2025)	0.5586	0.4111	227.66
InterCoser (Ours)	**0.5643**	**0.3106**	**184.24**

Table 2: Quantitative fidelity comparison to single-image-based generation methods.

Quantitative Results. We quantitatively compare the proposed method with three SoTA methods (Tang et al. 2024; Peng et al. 2024; He et al. 2025). Inspired by (Kirstain et al. 2023), we use user preference metrics to compare the generation quality to the SoTA methods. Fig. 5 shows the superior performance of our method compared to the SoTA methods in generation quality. Additionally, we calculate the FID score (Heusel et al. 2017) between the rendered views of the

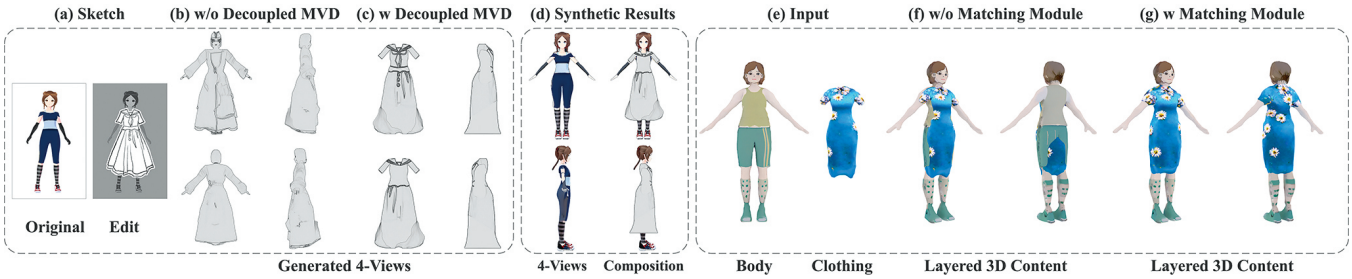

Figure 6: Ablation study of the Sketch-to-3D Decoupled Matching (SDM) Network.

Method	GSDM ↓	Sobel ↓	B-IoU: ↑
LGM (Tang et al. 2024)	0.2540	0.0417	0.0092
CharacterGen (Peng et al. 2024)	0.1976	0.0462	0.0153
STDGEN (He et al. 2025)	0.1953	0.0502	0.0136
InterCoser (Ours)	**0.1917**	**0.0397**	**0.0180**

Table 3: Normal quality comparison to single-image-based generation methods. GSDM, Sobel, and B-IoU denote the gradient magnitude similarity deviation of normal maps, normal map edge smoothness, and the contour IoU between normal maps and input images.

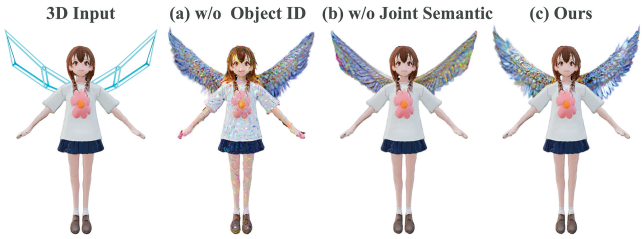

Figure 7: Ablation study of the Part-Level Interactive Editing.

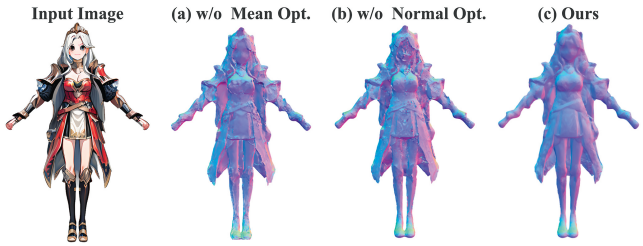

Figure 8: Ablation study of the Geometry-Prior-Guided 3D GS Reconstruction.

3D characters and the input reference images. Tab. 2 shows that our method achieves the lowest FID score, indicating the best generation quality. Furthermore, as shown in Tab. 2, our method achieves the highest SSIM score and the lowest LPIPS score, further validating its ability to produce detailed and accurate character appearances. Notably, Tab. 3 demonstrates our method achieves the lowest GSDM (Chen, Yang, and Xie 2006) and Sobel scores, along with the highest B-IoU, indicating our method's superior normal map smoothness and optimal alignment with input image contours.

Ablation Study

Effectiveness of the Sketch-to-3D Decoupled Matching (SDM) Network. Our decoupled MVD (Figs. 6 (b-d)) maintains semantic consistency with inputs while ensuring precise multi-view alignment. The 3D Matching Module (Fig. 6 (f-g)) resolves sparse-view semantic mismatches and enables clothing adaptation to varying body shapes. Additionally, the 3D Matching Module confines inter-layer penetration percentage to within 1.5%.

Effectiveness of the Part-Level Interactive Editing. Fig. 7(a) shows that without the Object ID mechanism, GS points move randomly during training, preventing precise edit confinement. Fig. 7(b) reveals that without joint local-global optimization, edited objects (like wings) lack global semantic consistency and may hinder convergence. Fig. 7(c) demonstrates our complete model achieves optimal semantic consistency, fine details, and perfect preservation of unedited regions.

Effectiveness of the Geometry-Prior-Guided 3D GS Reconstruction. Fig. 8(a) demonstrates that unconstrained Gaussian means cause excessive positional freedom, creating artifacts. Fig. 8(b) shows that without normal constraints on GS, the results lack smooth surfaces and fine details. Fig. 8(c) shows that our complete model achieves optimal quality with both smooth surfaces and high-fidelity details.

Conclusions

This paper introduces InterCoser, an innovative framework for disentangled interactive generation and editing of 3D characters. Our contribution enables precisely controlled generation and fine-grained editing of clothed characters while maintaining high quality. Specifically, we propose a novel SDM network that supports layered generation and clothing transfer across different body shapes via sketch interaction. We also introduce a part-level editing method preserving semantic consistency through simple 3D proxy manipulations, and a geometry-prior guided 3D Gaussian reconstruction that delivers both high-fidelity details and smooth surfaces. Experiments validate InterCoser's superior performance in the generation and editing of 3D characters.

Acknowledgments

This work was supported in part by National Key R&D Program of China (2023YFC3082100), National Natural Science Foundation of China (62501416), Science Fund for Distinguished Young Scholars of Tianjin (No. 22JCJQJC00040), and Natural Science Foundation of Tianjin (24JCYBJC01300).

References

Bae, G.; and Davison, A. J. 2024. Rethinking inductive biases for surface normal estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9535–9545.

Cao, Y.; Cao, Y.-P.; Han, K.; Shan, Y.; and Wong, K.-Y. K. 2023. Guide3D: Create 3D avatars from text and image guidance. *arXiv preprint arXiv:2308.09705*.

Cao, Y.; Cao, Y.-P.; Han, K.; Shan, Y.; and Wong, K.-Y. K. 2024. DreamAvatar: Text-and-shape guided 3D human avatar generation via diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 958–968.

Chen, G.-H.; Yang, C.-L.; and Xie, S.-L. 2006. Gradient-based structural similarity for image quality assessment. In *2006 international conference on image processing*, 2929–2932. IEEE.

Chen, H.; Lan, Y.; Chen, Y.; Zhou, Y.; and Pan, X. 2024a. MVDrag3D: Drag-based creative 3D editing via multi-view generation-reconstruction priors. *arXiv preprint arXiv:2410.16272*.

Chen, S.; Zhang, K.; Shi, Y.; Wang, H.; Zhu, Y.; Song, G.; An, S.; Kristjansson, J.; Yang, X.; and Zwicker, M. 2023. PAniC-3D: Stylized single-view 3D reconstruction from portraits of anime characters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21068–21077.

Chen, Y.; Chen, Z.; Zhang, C.; Wang, F.; Yang, X.; Wang, Y.; Cai, Z.; Yang, L.; Liu, H.; and Lin, G. 2024b. GaussianEditor: Swift and controllable 3D editing with Gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 21476–21485.

Cheng, X.; Yang, T.; Wang, J.; Li, Y.; Zhang, L.; Zhang, J.; and Yuan, L. 2023. Progressive3D: Progressively local editing for text-to-3D content creation with complex semantic prompts. *arXiv preprint arXiv:2310.11784*.

Dong, S.; Ding, L.; Huang, Z.; Wang, Z.; Xue, T.; and Xu, D. 2024. Interactive3D: Create what you want by interactive 3D generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4999–5008.

Gong, C.; Dai, Y.; Li, R.; Bao, A.; Li, J.; Yang, J.; Zhang, Y.; and Li, X. 2024. Text2Avatar: Text to 3D human avatar generation with codebook-driven body controllable attribute. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 16–20. IEEE.

Guo, C.; Su, Z.; Wang, J.; Li, S.; Chang, X.; Li, Z.; Zhao, Y.; Wang, G.; and Huang, R. 2025. SEGA: Drivable 3D Gaussian Head Avatar from a Single Image. arXiv:2504.14373.

He, R.; Huang, S.; Nie, X.; Hui, T.; Liu, L.; Dai, J.; Han, J.; Li, G.; and Liu, S. 2024. Customize your NeRF: Adaptive source driven 3D scene editing via local-global iterative training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6966–6975.

He, Y.; Zhou, Y.; Zhao, W.; Wu, Z.; Xiao, K.; Yang, W.; Liu, Y.-J.; and Han, X. 2025. StdGEN: Semantic-decomposed 3D character generation from single images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 26345–26355.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, I.; Song, J.; Hilliges, O.; et al. 2024. SiTH: Single-view textured human reconstruction with image-conditioned diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 538–549.

Hong, Y.; Zhang, K.; Gu, J.; Bi, S.; Zhou, Y.; Liu, D.; Liu, F.; Sunkavalli, K.; Bui, T.; and Tan, H. 2023. LRM: Large reconstruction model for single image to 3D. *arXiv preprint arXiv:2311.04400*.

Hu, H.; Zhou, Z.; Jampani, V.; and Tulsiani, S. 2024. MVD-Fusion: Single-view 3D via depth-consistent multi-view generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9698–9707.

Hu, T.; Hong, F.; and Liu, Z. 2024. StructLDM: Structured latent diffusion for 3D human generation. In *European Conference on Computer Vision*, 363–381. Springer.

Jiang, R.; Wang, C.; Zhang, J.; Chai, M.; He, M.; Chen, D.; and Liao, J. 2023. AvatarCraft: Transforming text into neural human avatars with parameterized shape and pose control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14371–14382.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4): 139–1.

Kirstain, Y.; Polyak, A.; Singer, U.; Matiana, S.; Penna, J.; and Levy, O. 2023. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36: 36652–36663.

Kolotouros, N.; Alldieck, T.; Corona, E.; Bazavan, E. G.; and Sminchisescu, C. 2024. Instant 3D human avatar generation using image diffusion models. In *European Conference on Computer Vision*, 177–195. Springer.

Kolotouros, N.; Alldieck, T.; Zanfir, A.; Bazavan, E.; Fieraru, M.; and Sminchisescu, C. 2023. DreamHuman: Animatable 3D avatars from text. *Advances in neural information processing systems*, 36: 10516–10529.

Li, Z.; Chen, Y.; Zhao, L.; and Liu, P. 2024. Controllable text-to-3D generation via surface-aligned Gaussian splatting. *arXiv preprint arXiv:2403.09981*.

Liao, T.; Yi, H.; Xiu, Y.; Tang, J.; Huang, Y.; Thies, J.; and Black, M. J. 2024. TADA! text to animatable digital avatars. In *2024 International Conference on 3D Vision (3DV)*, 1508–1519. IEEE.

Lindenberger, P.; Sarlin, P.-E.; and Pollefeys, M. 2023. LightGlue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF international conference on computer vision*, 17627–17638.

Liu, F.-L.; Fu, H.; Lai, Y.-K.; and Gao, L. 2024a. SketchDream: Sketch-based text-to-3D generation and editing. *ACM Transactions on Graphics (TOG)*, 43(4): 1–13.

Liu, M.; Shi, R.; Chen, L.; Zhang, Z.; Xu, C.; Wei, X.; Chen, H.; Zeng, C.; Gu, J.; and Su, H. 2024b. One-2-3-45++: Fast single image to 3D objects with consistent multi-view generation and 3D diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10072–10083.

Luo, Z.; Cui, Z.; Luo, S.; Chu, M.; and Li, M. 2025. VRDoh: Hands-on 3D Modeling in Virtual Reality. *ACM Transactions on Graphics (TOG)*, 44(4): 1–12.

Melas-Kyriazi, L.; Laina, I.; Rupprecht, C.; Neverova, N.; Vedaldi, A.; Gafni, O.; and Kokkinos, F. 2024. IM-3D: Iterative multiview diffusion and reconstruction for high-quality 3D generation. *arXiv preprint arXiv:2402.08682*.

Men, Y.; Lei, B.; Yao, Y.; Cui, M.; Lian, Z.; and Xie, X. 2024. En3D: An enhanced generative model for sculpting 3D humans from 2D synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9981–9991.

Mikaeili, A.; Perel, O.; Safaee, M.; Cohen-Or, D.; and Mahdavi-Amiri, A. 2023. SKED: Sketch-guided text-based 3D editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14607–14619.

Pan, P.; Su, Z.; Lin, C.; Fan, Z.; Zhang, Y.; Li, Z.; Shen, T.; Mu, Y.; and Liu, Y. 2024. HumanSplat: Generalizable Single-Image Human Gaussian Splatting with Structure Priors. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Peng, C.; Sun, J.; Chen, Y.; Su, Z.; Su, Z.; and Liu, Y. 2025. Parametric Gaussian Human Model: Generalizable Prior for Efficient and Realistic Human Avatar Modeling. arXiv:2506.06645.

Peng, H.-Y.; Zhang, J.-P.; Guo, M.-H.; Cao, Y.-P.; and Hu, S.-M. 2024. CharacterGen: Efficient 3D character generation from single images with multi-view pose canonicalization. *ACM Transactions on Graphics (TOG)*, 43(4): 1–13.

Qu, Y.; Chen, D.; Li, X.; Li, X.; Zhang, S.; Cao, L.; and Ji, R. 2025. Drag your Gaussian: Effective drag-based editing with score distillation for 3D Gaussian splatting. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, 1–12.

Ren, J.; He, C.; Liu, L.; Chen, J.; Wang, Y.; Song, Y.; Li, J.; Xue, T.; Hu, S.; Chen, T.; et al. 2023. Make-a-character: High quality text-to-3D character generation within minutes. *arXiv preprint arXiv:2312.15430*.

Tang, J.; Chen, Z.; Chen, X.; Wang, T.; Zeng, G.; and Liu, Z. 2024. LGM: Large multi-view Gaussian model for high-resolution 3D content creation. In *European Conference on Computer Vision*, 1–18. Springer.

VRoid. 2022. VRoid Hub.

Wang, Y.; Ma, J.; Shao, R.; Feng, Q.; Lai, Y.-K.; and Li, K. 2024a. Humancoser: Layered 3d human generation via semantic-aware diffusion model. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 436–445. IEEE.

Wang, Y.; Yi, X.; Wu, Z.; Zhao, N.; Chen, L.; and Zhang, H. 2024b. View-consistent 3D editing with Gaussian splatting. In *European conference on computer vision*, 404–420. Springer.

Xue, Y.; Xie, X.; Marin, R.; and Pons-Moll, G. 2024. Human-3Diffusion: Realistic Avatar Creation via Explicit 3D Consistent Diffusion Models. *Advances in Neural Information Processing Systems*, 37: 99601–99645.

Zang, Y.; Han, Y.; Ding, C.; Zhang, J.; and Chen, T. 2024. Magic3DSketch: Create colorful 3D models from sketch-based 3D modeling guided by text and language-image pre-training. *arXiv preprint arXiv:2407.19225*.

Zhang, H.; Chen, B.; Yang, H.; Qu, L.; Wang, X.; Chen, L.; Long, C.; Zhu, F.; Du, D.; and Zheng, M. 2024a. AvatarVerse: High-quality & stable 3D avatar creation from text and pose. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7124–7132.

Zhang, J.; Li, X.; Zhang, Q.; Cao, Y.; Shan, Y.; and Liao, J. 2024b. HumanRef: Single image to 3D human generation via reference-guided diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1844–1854.

Zhang, M.; Feng, Q.; Su, Z.; Wen, C.; Xue, Z.; and Li, K. 2024c. Joint2Human: High-quality 3D Human Generation via Compact Spherical Embedding of 3D Joints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zheng, X.; Wen, C.; Zhuo, S.; Xu, Z.; Li, Z.; Zhao, Y.; and Xue, Z. 2024. OHTA: One-shot Hand Avatar via Data-driven Implicit Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zhou, M.; Hyder, R.; Xuan, Z.; and Qi, G. 2024. UltrAvatar: A realistic animatable 3D avatar diffusion model with authenticity guided textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1238–1248.

Zhuang, J.; Kang, D.; Bao, L.; Lin, L.; and Li, G. 2025. GAGSM: Disentangled avatar generation with GS-enhanced mesh. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 292–303.

Zhuang, J.; Kang, D.; Cao, Y.-P.; Li, G.; Lin, L.; and Shan, Y. 2024. TIP-Editor: An accurate 3D editor following both text-prompts and image-prompts. *ACM Transactions on Graphics (TOG)*, 43(4): 1–12.

Zhuang, J.; Wang, C.; Lin, L.; Liu, L.; and Li, G. 2023. DreamEditor: Text-driven 3D scene editing with neural fields. In *SIGGRAPH Asia 2023 Conference Papers*, 1–10.

Zou, Z.-X.; Yu, Z.; Guo, Y.-C.; Li, Y.; Liang, D.; Cao, Y.-P.; and Zhang, S.-H. 2024. Triplane meets Gaussian splatting: Fast and generalizable single-view 3D reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10324–10335.

InterCoser: Interactive 3D Character Creation with Disentangled Fine-Grained Features Supplementary Material

Anonymous submission

In this document, we provide the following supplementary contents:

- Preliminary.
- Dual-Mode Appearance Completion Module.
- Implementation Details.
- User Study.
- Application.
- More Qualitative Comparison Results.
- Interactive Operation Demonstration.

We also provide a **demo video** along with this document.

Preliminary

Multi-view diffusion (MVD) The goal of MVD is simultaneously generating multiple images $\{I_0, I_1 \dots I_n\}$ given a text or an image, which have uniformly distributed view angles, with the same elevation. In 3D generation tasks, to control view angles, the absolute camera extrinsic matrix is encoded and added to the time-step embedding in a UNet. To ensure cross-view consistency, the 3D generation work based on MVD (Liu et al. 2023; Lin et al. 2023) utilizes a 3D attention module that shares the queries Q, keys K, and values V in all views.

Score Distillation Sampling (SDS) SDS is a loss function used to optimize generative models and is widely applied in text-guided 3D generation tasks. SDS minimizes the difference in conditional distributions between generated data and target data, gradually guiding the model to converge to the target distribution. Given a noise level t and the added noise ϵ, the goal of SDS is to adjust the model parameters θ by comparing the predicted noise $\hat{\epsilon}_\phi\left(\mathbf{x}_t; y, t\right)$ with the actual added noise ϵ. The specific loss function used in this paper is as follows:

$$\nabla_\theta \mathcal{L}_{\mathrm{SDS}}(\phi, \mathbf{x}=g(\theta, c)) \triangleq \mathbb{E}_{t,\epsilon}[\omega(t)(\hat{\epsilon}_\phi(\mathbf{x}_t; y, t)-\epsilon)\frac{\partial \mathbf{x}}{\partial \theta}], \tag{1}$$

where $g(\theta, c)$ is the rendered image of the 3D representation θ based on the camera condition c. $\omega(t)$ is a time-dependent weight function used to balance the influence of different noise levels on the optimization process. More details can be found in (Shi et al. 2023).

3D Gaussian Splatting (3DGS) 3D Gaussian Splatting (3DGS) (Kerbl et al. 2023) represents scenes using differentiable 3D Gaussian primitives characterized by their mean position $\boldsymbol{\mu} \in \mathbb{R}^3$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{3\times3}$, where the covariance is decomposed into a scaling vector $\boldsymbol{s} \in \mathbb{R}^3$ and rotation quaternion $\boldsymbol{q} \in \mathbb{R}^4$. Each Gaussian additionally possesses opacity $o \in \mathbb{R}$ and view-dependent color $\boldsymbol{c} \in \mathbb{R}^3$ represented through spherical harmonics.

The rendering process projects these 3D Gaussians into camera space as 2D Gaussians, which are then depth-sorted and composited to produce pixel colors. The differentiable splatting rendering process is:

$$C = \sum_{i\in\mathcal{N}} c_i \sigma_i \prod_{i-1}^{j=1}(1-\sigma_j), \quad \sigma_i = \alpha_i e^{-\frac{1}{2}(x)^T \Sigma^{-1}(x)} \tag{2}$$

where j indexes the Gaussians in front of g_i according to their distances to the camera center (i.e. depth), $\mathcal{N}$ is the number of Gaussians contributed to the ray, and c_i, α_i, and x_i represent the color, density, and distance of the sampling point to the center point of the i-th Gaussian.

Dual-Mode Appearance Completion Module

In order to generate fine-grained appearance, we design a dual-mode appearance completion module that supports generating and completing appearance for 3D models based on single-view hand-drawing or text prompts. First, we employ a local-to-global joint appearance loss to complete the appearance of the 3D content with reference I_{ref}. The optimization objective for the appearance completion module is as follows:

$$\nabla_\theta \mathcal{L}_{\mathrm{SDS}}^{tex\_n}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon}\left[\omega(t)\left(\hat{\epsilon}_\phi\left(\mathbf{x}_t^n; y^n, t\right)-\epsilon\right)\frac{\partial \mathbf{x}}{\partial \theta}\right], \tag{3}$$

$$\nabla_\theta \mathcal{L}_{\mathrm{SDS}}^{tex\_cp}(\phi, \mathbf{x}) \triangleq \mathbb{E}_{t,\epsilon}\left[\omega(t)\left(\hat{\epsilon}_\phi\left(\mathbf{x}_t^{cp}; y^{cp}, t\right)-\epsilon\right)\frac{\partial \mathbf{x}}{\partial \theta}\right], \tag{4}$$

where $\mathbf{x}_t\left(I^{nth}\right) \rightarrow \mathbf{x}_t^n, \mathbf{x}_t\left(I^{cp}\right) \rightarrow \mathbf{x}_t^{cp}$ respectively represents the noise sample of the rendered image I^{nth} of the n-th layer and the rendered image I^{cp} of the combined content (body + clothing) of the previous n layers, with time step t noise. y^n and y^{cp} represent the text prompts for the clothing

of the n-th layer and the combined 3D content of the previous n layers, which are achieved by the CLIP interrogator based on the appearance reference I_{ref}. $\mathcal{L}_{\text{SDS}}^{tex\_n}$ is used to independently supervise the generation of appearance for the clothing of the n-th layer, and $\mathcal{L}_{\text{SDS}}^{tex\_cp}$ is used to supervise the semantic adaptation of the appearance for the combined 3D content (body + clothing). For the definitions of other variables, please refer to Eq. 1. The optimization objectives for appearance generation and completion are as follows:

$$\begin{aligned}\mathcal{L}_{\text{tex}} = \lambda_{SDS}^{tex\_n}\mathcal{L}_{\text{SDS}}^{tex\_n} + \lambda_{SDS}^{tex\_cp}\mathcal{L}_{\text{SDS}}^{tex\_cp} \\ + \lambda_{color}\mathcal{L}_{\text{color}} + \lambda_{vgg}L_{\text{vgg}},\end{aligned} \tag{5}$$

where $\mathcal{L}_{\text{SDS}}^{\text{tex\_n}}$ and $\mathcal{L}_{\text{SDS}}^{\text{tex\_cp}}$ are the SDS (Score Distillation Sampling) losses for supervising the appearance generation of the n-th layer and the combination of the previous n layers of 3D content, respectively, as in Eq. 3,4. $\mathcal{L}_{\text{color}}$ is the Huber loss (Carver, O'TOOLE, and RAIFORD 1930) used to compute the color loss between the rendered image I_{bare} of the 3D model M_{bare} and the hand-drawn reference I_{ref} in the specified orthogonal view $view_e$. $\mathcal{L}_{\text{vgg}}$ uses the $block5\_conv3$ layer of VGG (Wu et al. 2015) to compute the feature loss, which compensates for the appearance features generated by the appearance module in views other than view $view_e$, by calculating the feature loss between the rendered image I_{bare} and the appearance reference I_{ref}. The above appearance generation optimization process allows the features of the specified appearance reference I_{ref} to be fully propagated across all views of the model, forming a unified appearance with consistent style and semantics.

Method	GQ ↑	AQ ↑	CR ↑	Overall ↑
LGM	3.17	3.24	3.36	3.25
CharacterGen	3.68	3.71	3.61	3.67
STDGEN	3.94	3.75	3.86	3.85
DreamCoser (Ours)	**4.13**	**3.84**	**3.94**	**3.97**

Table 1: User study for generation quality. GQ, AQ and CR mean geometric quality, appearance quality and consistency with the reference image.

Implementation Details

Character Dataset. Although CharacterGen (Peng et al. 2024) provides 13,746 stylized character RGB renderings, it lacks the corresponding high-quality normal maps essential for detailed 3D character generation. To address this, we developed the Character3D Dataset by first collecting 15,000 anime-style 3D models from VRoid-Hub (VRoid 2022), then applying rigorous quality control to remove low-fidelity models and those unsuitable for character disentanglement, resulting in a set of 12,153 professionally standardized characters. Each model was adjusted to a canonical A-pose with 45 ° downward arm rotation and rendered with comprehensive buffers including RGB, screen-space normals, depth maps, and alpha mattes using a physically-based anime shader pipeline adapted from (Blender Foundation 2023).

Hyperparameters. For generation based on character images, in the initial model generation stage, we optimize the DMTet representation with 3000 steps, with $\lambda_{mse} = 1$, $\lambda_{mask} = 0.2$, $\lambda_{smooth} = 0.5$. For sketch-based editing, in the layered editing stage, we optimize the geometry for 2500 steps, with $\lambda_{match} = 0.1$, $\lambda_{cloth} = 1$, $\lambda_{cp} = 1$. Specifically, alternate training is used in the layered editing stage, and the training ratio of the nth layer to the combination of the first n layers is $1:5$. In the Gaussian refinement stage, the Gaussian representation is optimized for 10000 steps, with $\lambda_{scale} = 1$, $\lambda_{\mathbf{N}} = 1$, $\lambda_n^{reg} = 0.25$. In the part-level editing stage, the edited object is optimized for 2000 steps, with $\lambda_{loc} = 1$, $\lambda_{glob} = 1$, $\lambda_p^{reg} = 1e3$. In particular, alternate training is used in the part-level editing stage, and the ratio of local editing to global editing training is $2:1$. Among them, the generation case takes 5 minutes to optimize, while the editing case takes about 6-10 minutes to optimize, on a single NVIDIA RTX4090 GPU. Specifically, our method can improve the generation speed by adjusting the parameters of the Gaussian refinement module.

User Study

We perform a user study comparing the character generation results of our method with those of other SoTA methods (Tang et al. 2024; Peng et al. 2024; He et al. 2025). The final evaluation results are provided in Tab. 1. We generate 3D characters for different methods based on 50 Anime-style character images, selected from the VRoid dataset (VRoid 2022) and online character images. Fifty volunteers (including 24 males and 26 females, aged between 18 and 50 years) are invited to rank the methods in terms of (1) geometric quality (GQ), (2) appearance quality (AQ), and (3) consistency with the reference image (CR). Volunteers score each comparative indicator for each method from 1 (worst) to 5 (best). Our method achieves optimal scores across all three metrics, indicating superior generative quality for geometry and appearance based on image inputs.

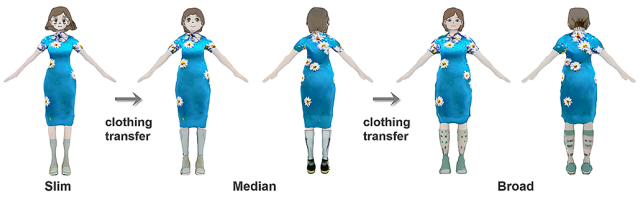

Figure 1: Clothing Transfer. Our method supports transferring generated 3D clothing layers between bodies of different shapes by using our 3D Matching Module. Our clothing transfer functionality facilitates the reuse of 3D generated content for AIGC, gaming, and VR/AR applications.

Application

Benefiting from layered and part-level generation and editing capabilities, our method can exchange generated clothing between different body shapes, enable physical collisions of multi-layer clothing, enable virtual try-on for 3D digital characters, and edit 3D characters using simple

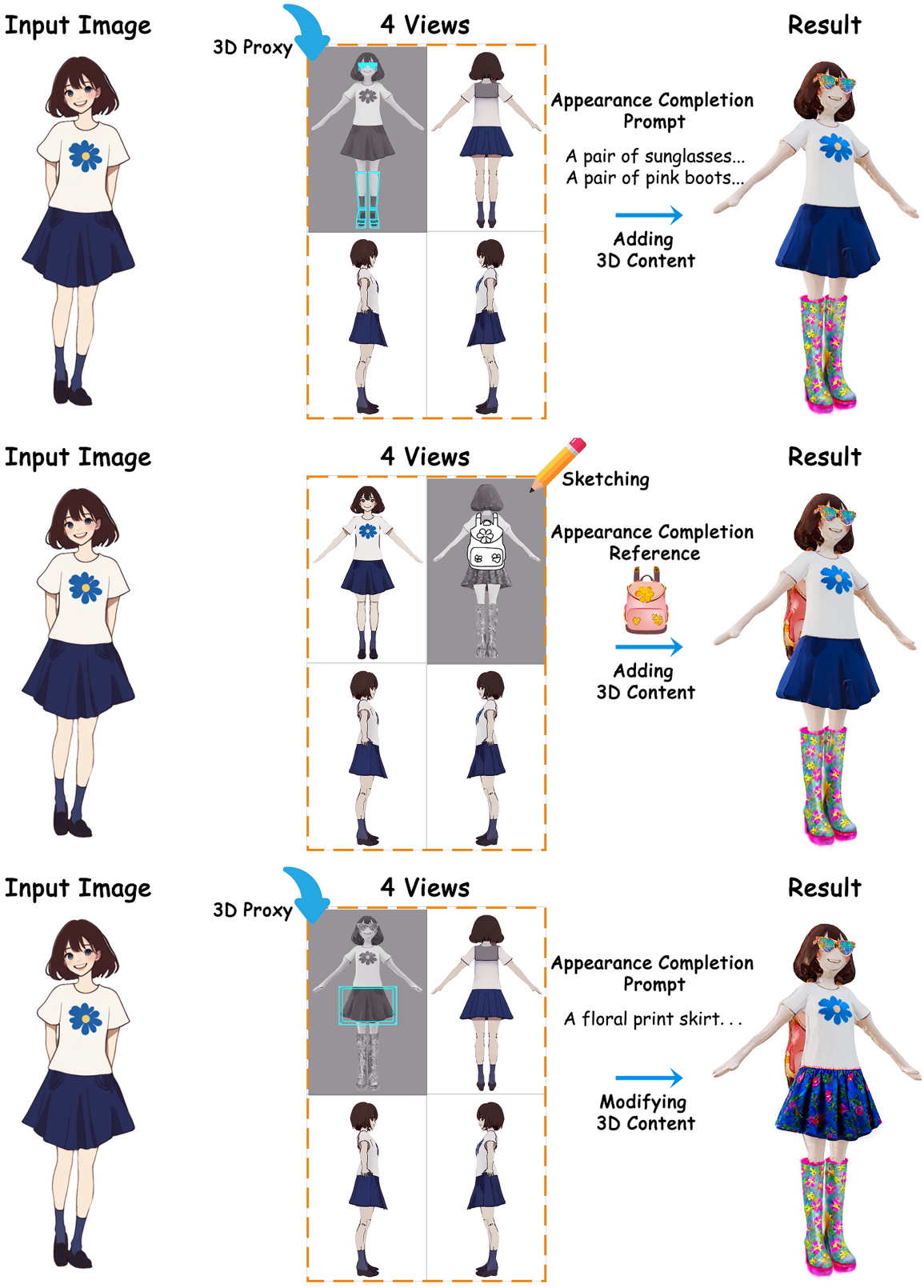

Figure 2: Interactive Editing Results. Our method enables users to modify 3D digital characters' clothing and accessories through direct sketch-based editing on character image or via intuitive 3D proxy manipulation for local fine-grained modifications and additions. Our approach achieves 3D virtual clothing changes and equipment/prop replacements for 3D digital characters. This convenient 3D interactive generation and editing facilitates AIGC, gaming, and AR/VR applications.

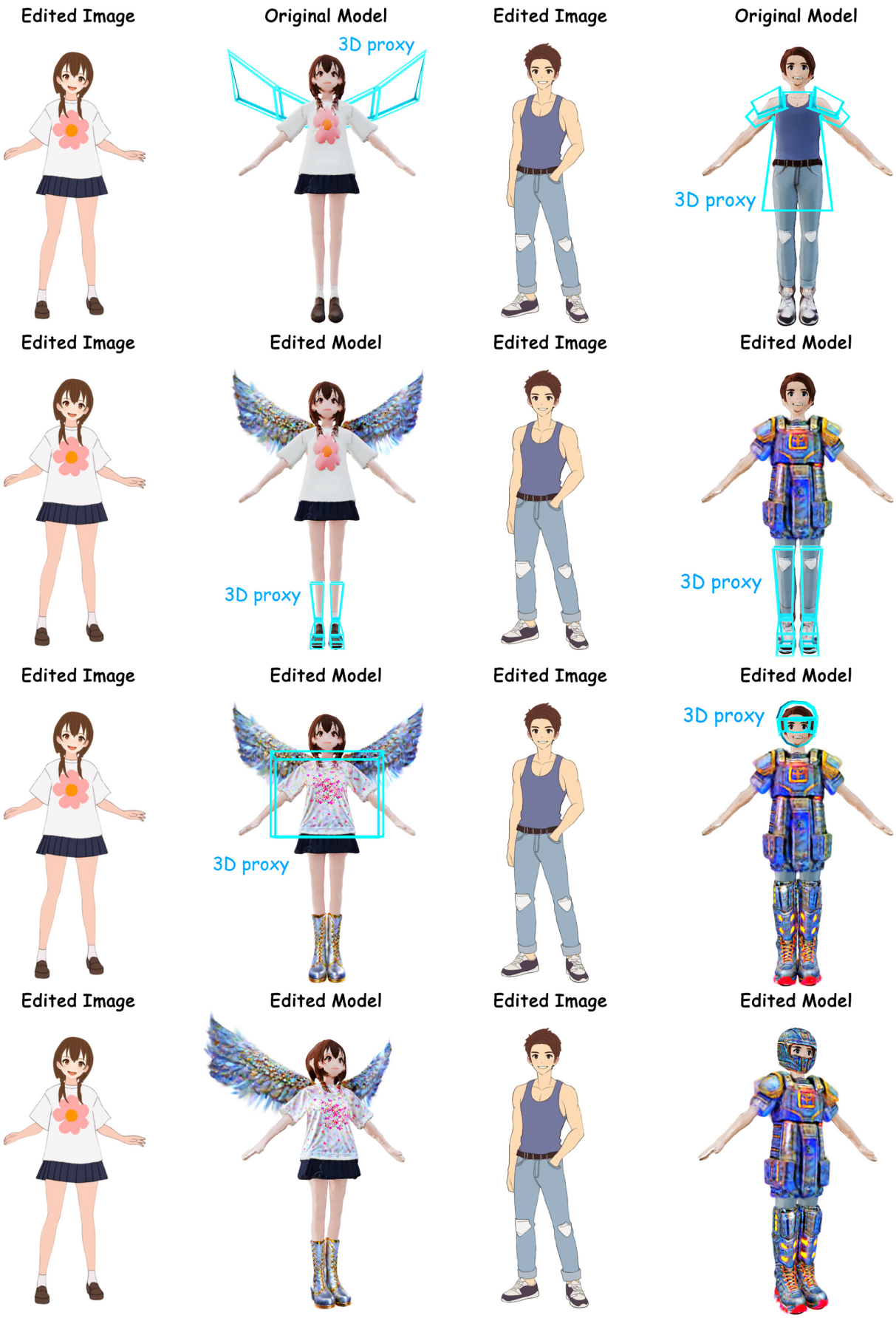

Figure 3: Interactive Editing Results. Our method enables users to modify 3D digital characters' clothing and accessories through direct sketch-based editing on character image or via intuitive 3D proxy manipulation for local fine-grained modifications and additions. Our approach achieves 3D virtual clothing changes and equipment/prop replacements for 3D digital characters. This convenient 3D interactive generation and editing facilitates AIGC, gaming, and AR/VR applications.

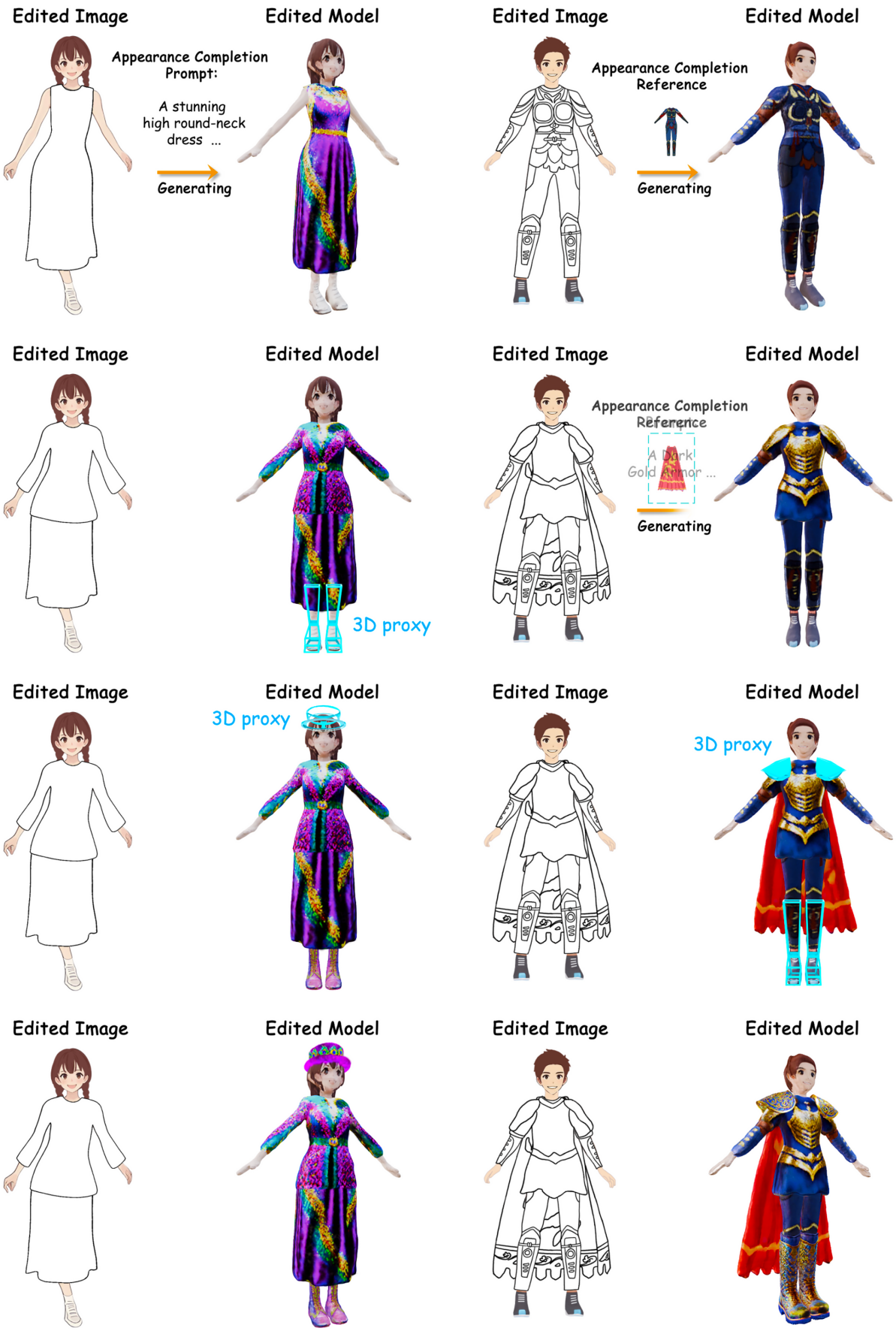

Figure 4: Interactive Editing Results. Our method enables users to modify 3D digital characters' clothing and accessories through direct sketch-based editing on character image or via intuitive 3D proxy manipulation for local fine-grained modifications and additions. Our approach achieves 3D virtual clothing changes and equipment/prop replacements for 3D digital characters. This convenient 3D interactive generation and editing facilitates AIGC, gaming, and AR/VR applications.

Figure 5: Animation and Cloth Collision Simulation for 3D Characters. Benefiting from the disentangled 3D representation and canonical pose in our method, we can easily obtain animatable 3D layered dressed characters and achieve physical simulation of clothing fabrics in physics or game engines.

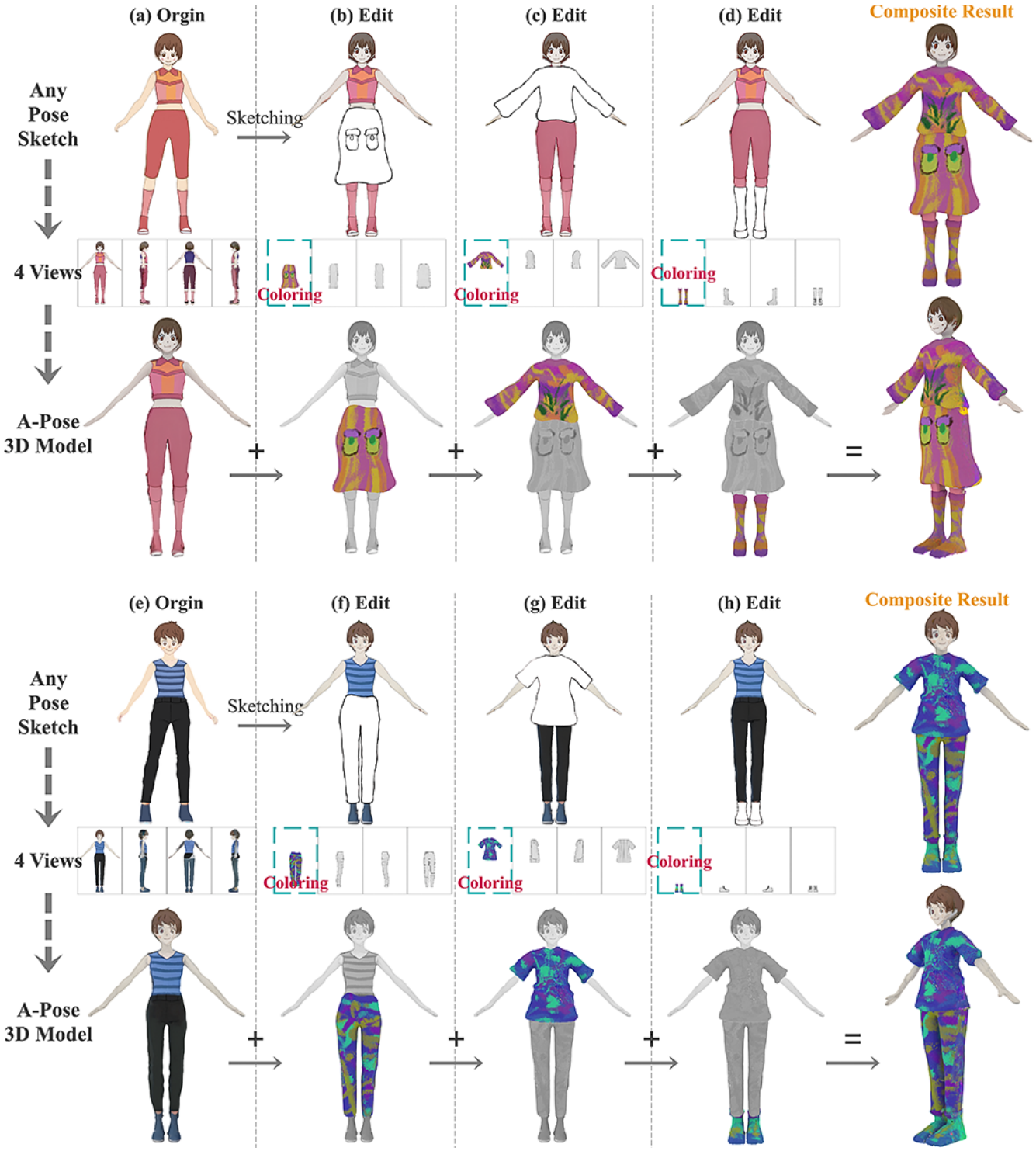

Figure 6: Results of Editing based on Simple Doodles. Our method can take simple and exaggerated hand-drawn shapes as input, edit them, and generate high-quality 3D content that is consistent with the input.

Figure 7: Single-Image-based Generation Comparison. Our method can obtain high-quality appearance and fine local details with smooth surfaces through appearance and geometry refinement based on 3D Gaussian splatting.

Figure 8: 3D Proxy-based Interactive Editing Interface. To facilitate users in generating and editing 3D character content based on 3D proxy, we built an interactive generation application. Users can generate and edit 3D characters through interface interactions. Meanwhile, our method supports two generation modes: quality-first and speed-first. The quality mode generates detailed models and appearance, while the speed-first mode accelerates the generation process to enhance user experience.

sketches. To achieve these goals, users only need intuitive sketch and 3D proxy editing.

Clothing Transfer. As shown in Fig. 1, our method enables the exchange of generated clothing layers between bodies of different shapes, which facilitates the reuse of 3D generated content for AIGC, gaming, and VR / AR applications, while providing diverse clothing and equipment options for digital humans in downstream applications.

Virtual Try-On. As shown in Fig. 2, Fig. 3 and Fig. 4, our method can not only add layered clothing to 3D characters through direct sketch editing on the initial character image, but also perform fine-grained addition of accessories and local modifications to characters through intuitive 3D proxy operations. Combined with our clothing transfer function, our method can provide convenient and diverse virtual dressing and game equipment changing capabilities for 3D digital characters.

Clothing Collision Simulation. As shown in Fig. 5, our method can generate multi-layer clothed characters with high-quality appearence and geometric details, enabling collision simulation between clothing layers. We employ the Unreal game engine to rig the generated A-pose clothed characters for animation and simulate physical collisions between the clothing of 3D characters.

Editing Based on Simple Doodles. As shown in Fig. 6, our method can take simple and exaggerated hand-drawn shapes as input, edit them, and generate high-quality 3D content that is consistent with the input.

More Qualitative Comparison Results

To compare under a unified posture, we use a single character image in A-pose as input for a qualitative comparison between our generation method and SoTA methods (Tang et al. 2024; Peng et al. 2024; He et al. 2025). Fig. 7 shows that our generation results outperform SoTA results. LGM lacks fine textures and smooth geometric structures due to its lightweight asymmetric U-Net architecture, which sacrifices some texture details. CharacterGen loses local geometry such as hair or clothing, despite its introduction of multi-view pose normalization to improve handling of complex poses. While STDGEN uses multi-view normal maps for geometry, its 3D segmentation from sparse 2D representations yields geometric artifacts. In contrast, our method generates 3D results with fine geometric details and smooth surfaces through our geometry-prior-based reconstruction strategy.

Interactive Operation Demonstration

To facilitate users in generating and editing 3D character content based on sketches and 3D proxy, we built an interactive generation application, as shown Fig. 8. Users can generate and edit 3D characters through interface interactions. Meanwhile, our method supports two generation modes: quality-first and speed-first. The quality mode generates detailed models and appearance, while the speed-first mode accelerates the generation process to enhance user experience.

References

Blender Foundation. 2023. Blender - a 3D modelling and rendering package. Accessed: 2023-10-15.

Carver, H. C.; O'TOOLE, A.; and RAIFORD, T. 1930. *The annals of mathematical statistics*. Edwards Bros.

He, Y.; Zhou, Y.; Zhao, W.; Wu, Z.; Xiao, K.; Yang, W.; Liu, Y.-J.; and Han, X. 2025. Stdgen: Semantic-decomposed 3d character generation from single images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 26345–26355.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4): 139–1.

Lin, C.-H.; Gao, J.; Tang, L.; Takikawa, T.; Zeng, X.; Huang, X.; Kreis, K.; Fidler, S.; Liu, M.-Y.; and Lin, T.-Y. 2023. Magic3D: High-resolution text-to-3D content creation. 300–309.

Liu, R.; Wu, R.; Van Hoorick, B.; Tokmakov, P.; Zakharov, S.; and Vondrick, C. 2023. Zero-1-to-3: Zero-shot one image to 3D object. 9298–9309.

Peng, H.-Y.; Zhang, J.-P.; Guo, M.-H.; Cao, Y.-P.; and Hu, S.-M. 2024. Charactergen: Efficient 3d character generation from single images with multi-view pose canonicalization. *ACM Transactions on Graphics (TOG)*, 43(4): 1–13.

Shi, Y.; Wang, P.; Ye, J.; Long, M.; Li, K.; and Yang, X. 2023. MVDream: Multi-view Diffusion for 3D Generation. *CoRR*, abs/2308.16512.

Tang, J.; Chen, Z.; Chen, X.; Wang, T.; Zeng, G.; and Liu, Z. 2024. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, 1–18. Springer.

VRoid. 2022. VRoid Hub.

Wu, R.; Yan, S.; Shan, Y.; Dang, Q.; and Sun, G. 2015. Deep image: Scaling up image recognition. *arXiv preprint arXiv:1501.02876*.