Auto-Connect: Connectivity-Preserving RigFormer with Direct Preference Optimization

Jingfeng Guo[1*], **Jian Liu**[2,7*], **Jinnan Chen**[3†], **Shiwei Mao**[4,7], **Changrong Hu**[5,7], **Puhua Jiang**[4,7]
Junlin Yu[6,7], **Jing Xu**[7], **Qi Liu**[1†], **Lixin Xu**[7], **Zhuo Chen**[7], **Chunchao Guo**[7]
[1]South China University of Technology
[2]Hong Kong University of Science and Technology [3]National University of Singapore
[4]Tsinghua Shenzhen International Graduate School [5]University of Science and Technology of China
[6]Beijing Normal University [7]Tencent Hunyuan
https://autoconnectrig.github.io/

Abstract

We introduce Auto-Connect, a novel approach for automatic rigging that explicitly preserves skeletal connectivity through a connectivity-preserving tokenization scheme. Unlike previous methods that predict bone positions represented as two joints or first predict points before determining connectivity, our method employs special tokens to define endpoints for each joint's children and for each hierarchical layer, effectively automating connectivity relationships. This approach significantly enhances topological accuracy by integrating connectivity information directly into the prediction framework. To further guarantee high-quality topology, we implement a topology-aware reward function that quantifies topological correctness, which is then utilized in a post-training phase through reward-guided Direct Preference Optimization. Additionally, we incorporate implicit geodesic features for latent top-k bone selection, which substantially improves skinning quality. By leveraging geodesic distance information within the model's latent space, our approach intelligently determines the most influential bones for each vertex, effectively mitigating common skinning artifacts. This combination of connectivity-preserving tokenization, reward-guided fine-tuning, and geodesic-aware bone selection enables our model to consistently generate more anatomically plausible skeletal structures with superior deformation properties.

1 Introduction

The creation of highly detailed 3D shapes [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11] and digital avatars [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22] has become increasingly accessible through advancements in generative modeling technologies. Despite these impressive capabilities in static content generation, these models remain fundamentally limited for dynamic applications. Some early works address this limitation by predicting dynamics through per-vertex deformation or physical simulation [23, 24, 25, 26, 27, 28, 29, 30, 31]. In contrast, rigging offers a parametric representation that establishes a skeleton for a 3D model and defines surface deformation in response to skeletal movement which is compatible to graphics pipeline. Despite its critical importance in animation pipelines, auto-rigging

[*]Equal contributions. Work primarily done during an internship at Tencent Hunyuan
[†]Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

remains a challenging problem due to the complexity of accurately modeling skeletal structures, ensuring proper joint and bone connectivity, and preventing artifacts during animation.

Early approaches to auto-rigging relied on fixed topology templates [32, 25], which limited their applicability across diverse character morphologies. Later developments, such as RigNet [33], employed more flexible strategies including clustering for joint position acquisition and Minimum Spanning Tree (MST) algorithms for topology construction. Additionally, various optimization-based methods [34, 35, 36, 37] have been proposed to generate character-specific rigs, but often require additional optimization cost and suffer from generalization issues. Recent advancements in generative models and the expansion of 3D datasets have made auto-rigging more scalable.

Current learning-based approaches typically fall into two categories: methods [38, 39] that predict bone positions without explicitly considering the connectivity between joints, and methods [40] that first predict skeletal joint points before determining their connectivity. As a result, these approaches frequently produce suboptimal topology quality, primarily because their representations fail to effectively capture the inherent topological relationships within skeletal structures. This deficiency frequently leads to unrealistic deformations in animations.

To address these limitations, we introduce Auto-Connect, a novel approach that explicitly preserves skeletal connectivity. By incorporating a connectivity-preserving tokenization scheme, Auto-Connect ensures that joints and bones are connected in a way that maintains the inherent structure of the skeleton, overcoming the topological errors common in previous methods. Our method defines endpoints for each joint's children and for each hierarchical layer, effectively automating connectivity relationships within the prediction framework itself. This fundamental redesign significantly enhances topological accuracy by integrating connectivity information directly into the model's representation. Based on this representation, we design an autoregressive training framework for skeleton trees, RigFormer, which incorporates level position embedding and a set of data augmentation strategies.

Furthermore, purely next-token prediction with cross-entropy loss focuses solely on local conditional distribution modeling, failing to adequately capture joint distribution properties. To address this limitation, we introduce a joint distribution constraint during post-training through a DPO loss framework. Our approach primarily emphasizes topology improvement by incorporating carefully designed topology-aware reward functions. We evaluate rig quality based on joint position accuracy and topological quality. For position accuracy, we calculate the chamfer distance between predicted joint positions and ground truth, providing a robust measure of geometric fidelity. For topological quality, we employ two complementary metrics: Tree Edit Distance, which measures the cost of transforming the predicted skeleton topology into the ground truth through a series of edit operations, and Hierarchical Jaccard Similarity, which quantifies the overlap between hierarchical structures while considering parent-child relationships. This reward function is then utilized in a post-training phase through our reward-guided Direct Preference Optimization (DPO), guiding the model toward generating both geometrically accurate and topologically sound skeletal structures. Finally, we incorporate implicit geodesic features for latent top-k bone selection, which substantially improves skinning quality by leveraging spatial relationships within the character's geometry. This approach implicitly determines the most influential bones for each vertex, effectively mitigating common skinning artifacts, particularly stretching phenomena that occur during extreme deformations.

Through extensive experiments on public benchmarks, we demonstrate that Auto-Connect substantially outperforms previous state-of-the-art methods, achieving superior results in joint location accuracy, topological consistency, and skinning quality. We summarize our contributions as follows:

- We introduce Auto-Connect, a novel automatic rigging pipeline that explicitly preserves skeletal connectivity through a connectivity-preserving tokenization scheme coupled with an enhanced pre-training framework, RigFormer.
- We develop a topology-aware reward function tailored for skeleton tree structures, and build upon this, we present a rigging post-training phase through our reward-guided DPO to further improve topology quality. To the best of our knowledge, this is the first work to combine reinforcement learning with the rigging task.
- We present a plug-and-play geodesic-aware bone probability prediction module that incorporates implicit geodesic features to dynamically determine the top-k bone for each vertex, effectively mitigating common skinning artifacts.

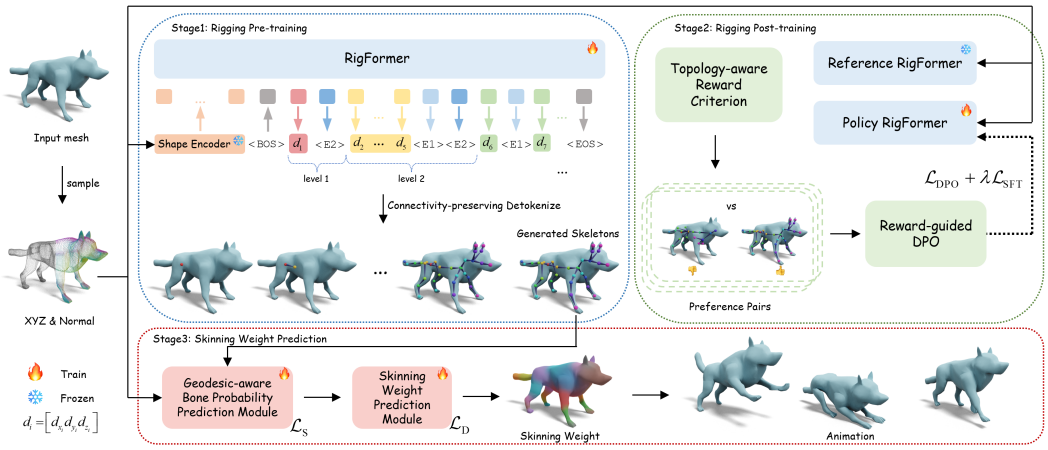

Figure 1: **Overview of the Auto-Connect.** The pipeline consists of three main stages. In the **Rigging Pre-training** stage, a point cloud sampled from the input 3D mesh is processed by the shape encoder to extract geometric features, which are subsequently fed into our autoregressive RigFormer to generate a token sequence. The generated sequence is then processed using our connectivity-preserving detokenization to construct the skeleton tree. In the **Rigging Post-training** stage, preference pairs are constructed using our topology-aware reward criterion, and RigFormer is fine-tuned with our reward-guided DPO for preference-driven optimization. Finally, in the **Skinning Weight Prediction** stage, the generated skeleton and mesh vertices serve as input. Our geodesic-aware bone probability prediction module is employed to implicitly determine the most influential bones to predict the skinning weights, enabling mesh animation.

2 Related Work

2.1 Automatic Rigging and Skinning

Automatic Rigging can be categorized into template-based [32, 41, 42, 43, 44] methods and template-free [33, 45, 40, 38, 39] paradigms. Template-based methods mainly focus on humanoid characters and are limited to fixed skeleton topologies, restricting their use for diverse character types. Template-free methods like RigNet [33] and AnimSkelVolNet [45] use regression and clustering for joint prediction, along with MST for connectivity. However, their hybrid architectures face optimization challenges due to non-differentiable clustering and MST operations. RigAnything [40] and MagicArticulate [38] use autoregressive rigging but neglect the skeleton's hierarchical structure and parent-child joint connections. This forces them to either rely on additional modules for joint connectivity or re-encode parent joints multiple times, which increases sequence length and computational cost. UniRig [39] attempts to address these shortcomings by extracting bone chains through Depth-First Search. However, the connections between bone chains rely on heuristic connection rules, which merge joints within a predefined distance threshold, making it non-end-to-end and prone to compounding errors. Additionally, it often predicts the termination token prematurely, leading to incomplete skeletal structures with missing bone chains. In contrast, our method overcomes these limitations by integrating topology information directly into the model's representation. Current skinning methods [33, 46, 47] statically select the k-nearest bones for each vertex and assume that only these bones influence the vertex, then use Graph Neural Networks to predict skinning weights. However, complex mesh-skeleton topologies often render distance calculations unreliable, leading to critical binding errors. To address this, we present a plug-and-play geodesic-aware module that dynamically identify the k most probable influencing bones conditioned on geodesic feature cues.

2.2 Autoregressive Models for 3D Generation

Autoregressive transformers [48, 49] have radically transformed visual generation [50, 51, 52, 53, 54, 55] through their sophisticated sequential approach of synthesizing images using discrete tokens derived from image tokenizers. This paradigm has achieved remarkable success by decomposing the complex image generation task into a series of manageable token prediction steps, enabling more coherent and controllable outputs. Building upon this foundation, recent work [4, 56, 57, 58, 59, 60, 61] has introduced specialized mesh tokenizers that extend the autoregressive framework to 3D mesh

generation. These methods effectively discretize 3D geometry into sequential tokens that can be predicted in an autoregressive manner, similar to language modeling. Our method builds upon these advances, introducing a novel connectivity-preserving tokenizer that enables more accurate, diverse, and artist-intuitive skeleton generations.

2.3 RLHF with Direct Preference Optimization

With the rapid advancement of Large Language Models (LLMs) [62, 63] and Vision-Language Models (VLMs) [64, 65, 66], aligning policy models with human preferences has become increasingly critical. Post-training techniques such as Reinforcement Learning with Human Feedback (RLHF) and Direct Preference Optimization (DPO) aim to improve model performance by reflecting user intentions. Early RLHF methods [67] trained on manually labeled preference pairs and optimized policies using Proximal Policy Optimization (PPO) [68], but faced instability and high computational costs. DPO addresses this by removing the need for an explicit reward model and using an implicit reward function based on PPO optimality, optimizing policies via maximum likelihood estimation with the Bradley-Terry model [69]. Building upon this foundation, we design a topology-aware reward function for skeleton trees and propose a reward-guided DPO to encourage the generation of topologically accurate skeletal structures. To the best of our knowledge, this is the first successful implementation of DPO in rigging tasks.

3 Method

Our Auto-Connect comprises three core innovations, as illustrated in Figure 1. First, Section 3.1 introduces the rigging pre-training stage with a novel connectivity-preserving tokenization scheme and RigFormer training framework. Building upon this foundation, Section 3.2 presents the rigging post-training stage using our reward-guided DPO with the proposed topology-aware reward criteria. Finally, Section 3.3 details the proposed geodesic-aware bone probability prediction module that enables plug-and-play integration with existing skinning methods for precise deformation control.

3.1 Rigging Pre-training

Unlike template-based methods [41, 42, 43, 44] that rely on fixed topologies for generating specific skeleton categories, or prior template-free methods [33, 40, 38, 39] that overlook skeleton topology encoding, we propose a connectivity-preserving tokenizer that efficiently encodes both skeleton hierarchical structure and parent-child joint connections. This tokenizer enables the automation of joint connectivity in an autoregressive paradigm, allowing for the generation of diverse skeletons.

Connectivity-preserving Tokenization. Given an input mesh and skeleton, we first normalize them into a unit cube space $[-0.5, 0.5]^3$ and apply n-bit quantization to the joint coordinates via

$$d_k = \lfloor (j_k + 0.5) \times 2^n \rfloor \; k \in \{x, y, z\} \quad (1)$$

where j_k and d_k denote the original and discretized coordinates, respectively. Next, we traverse the skeleton tree in a breadth-first (BFS) order, as shown in Figure 2. Specifically, based on the standard BFS traversal, we insert a special token <E1> after visiting all the child joints of a parent joint to indicate the endpoints for the current parent's children. Note that leaf joints not at the last level still need an <E1> to signify this. Similarly, after traversing all the joints in the current level, we insert another special token <E2> to mark the completion of that level. Additionally, we incorporate a height-aware spatial prior by sorting child joints under each parent joint according to their z-axis coordinates, which reduces the difficulty for the model in regressing joint positions. Finally, we obtain $3J + M + L$ tokens, where J is the number of joints, L is the number of levels in the skeleton tree, and M is the total number of joints in the first $L - 1$ levels.

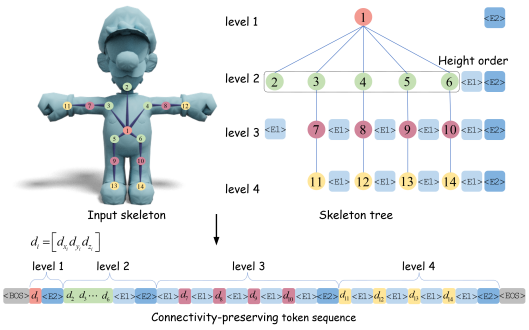

Figure 2: **Connectivity-preserving tokenization process.** The number indicates the joint indices.

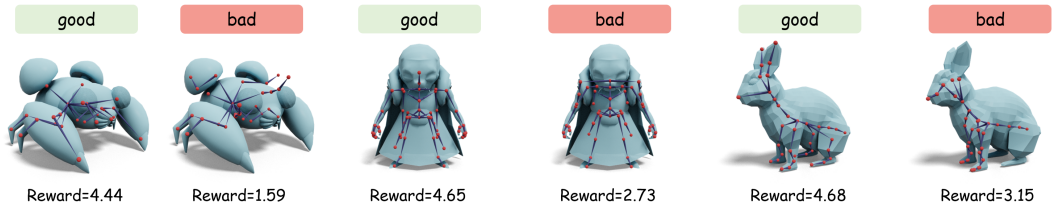

Figure 3: **Examples of the collected preference pairs.** Skeleton trees with higher reward exhibit superior topology and better align with human intuition, making them the preferred choice.

Shape-conditioned generation. we randomly sample $N = 20000$ surface points from the input mesh to construct a point cloud representation $\mathcal{P} \in \mathbb{R}^{N\times 3}$ with corresponding surface normals $\mathcal{N} \in \mathbb{R}^{N\times 3}$. These geometric primitives are encoded through a pre-trained Michelangelo [70] encoder $\mathcal{E}_g$, which captures both local geometric details and global shape semantics. The generates shape condition $\mathbf{c_g} = \mathcal{E}_g(\mathcal{P}, \mathcal{N})$, serving as the context for the autoregressive generation process.

RigFormer. We adopt a standard transformer with parameter θ to model the skeleton sequence and leverage cross-attention for shape conditioning. The training process is achieved using the next token prediction loss:

$$\mathcal{L}_{\text{stage1}} = -\prod_{i=1}^{T} p\left(\tau_i \left| \tau_{1:i-1} \oplus \mathbf{e}_{\ell_{1:i-1}}, \mathbf{c}_g; \theta\right.\right) \tag{2}$$

where T denotes the total sequence length. To explicitly model skeletal hierarchy, we additionally inject level position embedding $\mathbf{e}_\ell$ into the transformer. Here, $p\left(\tau_i \left| \tau_{1:i-1} \oplus \mathbf{e}_{\ell_{1:i-1}}\right.\right)$ denotes the conditional probability of token τ_i given the preceding tokens and level embeddings in the sequence.

During inference, the generation process starts with only the shape tokens as input and progressively generates skeleton tokens. The resulting token sequence is then converted into the final skeleton using our connectivity-preserving detokenization. With the proposed special tokens `<E1>` and `<E2>`, the hierarchical structure and parent-child joint connections can be automatically determined.

3.2 Rigging Post-training

Next-token prediction method focuses only on local conditional distributions, neglecting the critical aspects of joint distribution modeling, especially for topology preserving. We implement a joint distribution constraint during the post-training phase, utilizing DPO loss to further improve the topology quality. Specifically, we introduce a topology-aware reward function to evaluate the quality of skeletons. Based on this criterion, we construct preference pairs and propose a reward-guided DPO to fine-tune our RigFormer in a preference-driven manner. Moreover, to prevent overfitting, we add the SFT auxiliary constraint loss during DPO training.

Topology-aware Reward. Given the predicted skeleton $\mathcal{S}_p$ and ground truth skeleton $\mathcal{S}_g$, we evaluate the quality of the predicted skeleton in terms of spatial accuracy and topological fidelity. Spatial accuracy is measured by the Chamfer Distance (CD) between the generated and ground truth skeletons, which is calculated as the sum of CD-J2J, CD-J2B, and CD-B2B. Each reward decays linearly from 1 to 0 points as the CD value increases from 0 to 10%, which can be formalized as:

$$R_{\text{CD}} = \sum_{i\in\{\text{J2J,J2B,B2B}\}} \max\left(1 - \frac{\text{CD}_i}{10\%}, 0\right) \in [0, 3] \tag{3}$$

For topological fidelity, we first use the Tree Edit Distance (TED) to measure the minimum edit operations required to transform the predicted skeleton into the ground truth skeleton. This value is normalized by the total number of joints in both trees to eliminate scale bias:

$$R_{\text{TED}} = 1 - \frac{\text{TED}\left(\mathcal{S}_p, \mathcal{S}_g\right)}{|\mathcal{J}_p| + |\mathcal{J}_g|}, R_{\text{TED}} \in [0, 1] \tag{4}$$

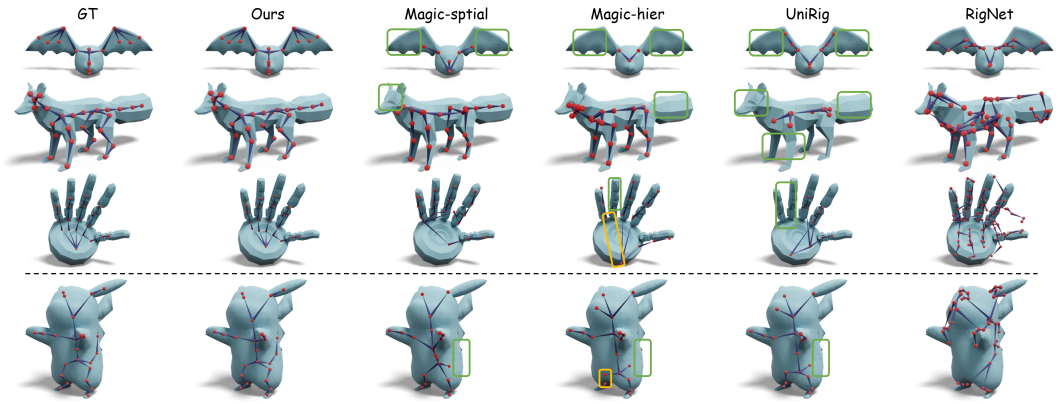

Figure 4: **Qualitative comparison of rigging result on Art-XL2.0 (top) and MR (bottom).** Our connectivity-preserving representation effectively captures intrinsic skeletal topology, and reward-guided fine-tuning enables the generated skeletons to better align with artistic aesthetics. Additional results are provided in the appendix B.1.

where $|\mathcal{J}_p|$ and $|\mathcal{J}_g|$ represent the number of joints. Then, we calculate the Hierarchical Jaccard Similarity (HJS), which evaluates the local topological accuracy of the shared joints $\mathcal{J}_{\text{common}}$:

$$R_{\text{HJS}} = \frac{1}{|\mathcal{J}_{\text{common}}|} \sum_{j \in \mathcal{J}_{\text{common}}} \frac{|\mathcal{C}_{p(j)} \cap \mathcal{C}_{g(j)}|}{|\mathcal{C}_{p(j)} \cup \mathcal{C}_{g(j)}|} \in [0, 1] \tag{5}$$

where $\mathcal{C}_{p(j)}$ and $\mathcal{C}_{g(j)}$ represent the children set of joint j in the predicted and ground truth skeletons respectively. The composite reward aggregates these as a 5-point scale:

$$\text{Reward} = \underbrace{R_{\text{CD}}}_{\substack{\text{Spatial} \\ \text{(3pts)}}} + \underbrace{R_{\text{TED}} + R_{\text{HJS}}}_{\substack{\text{Topological} \\ \text{(2pts)}}} \in [0, 5] \tag{6}$$

Preference Pair Construction. We generate four candidate skeletons for each input and select preference pairs using our topology-aware reward criterion. Specifically, pairs in which both skeletons reward below a predefined threshold (set to 3) are discarded. When a skeleton outperforms its counterpart by more than 0.5 points, the superior skeleton is chosen as the preferred one. Figure 3 shows some selection cases of our collected preference pairs.

Reward-guided DPO. Unlike standard DPO, which leverages only the binary preference order between two responses, our approach incorporates reward differences to exploit richer comparative information. By multiplying the DPO loss with the reward difference, we amplify gradients for pairs with larger disparities, guiding the model to better distinguish between high-fidelity and low-fidelity skeletons and enhance the discriminative gap during optimization. By training on triplets of inputs x, high-reward outputs y_g, and low-reward outputs y_b, the model learns to prioritize generating high-reward samples:

$$\mathcal{L}_{\text{DPO}}(x, y_g, y_b) = -\mathbb{E}_{(x,y_g,y_b)\sim\mathcal{D}}\left[\log \sigma\left(\beta \log \frac{\pi(y_g|x)}{\pi_{\text{ref}}(y_g|x)} - \beta \log \frac{\pi(y_b|x)}{\pi_{\text{ref}}(y_b|x)}\right) \cdot \left(r^\star(x, y_g) - r^\star(x, y_b)\right)\right] \tag{7}$$

where β is a hyperparameter balancing the distance to the reference policy π_{ref}, set to 0.3 in our experiments, π denotes the policy model being optimized, $r^\star(x, y_g)$ and $r^\star(x, y_b)$ are the rewards assigned by our topology-aware reward function. Furthermore, while high-reward skeletons exhibit better quality, they may still fall short of the perfect ground truth y_{gt}, which obtains the maximum 5-point reward. To address this limitation and ensure the model not only discriminates between good and bad cases but also retains foundational generative capabilities, we introduce an auxiliary SFT loss, where the ground truth y_{gt} is used to compute the next-token prediction loss. This loss $\mathcal{L}_{\text{SFT}}$ mitigates excessive deviation from the pre-trained knowledge base, ensuring stability during preference alignment. Finally, the total loss for the post-training stage is:

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{DPO}}(x, y_g, y_b) + \lambda \mathcal{L}_{\text{SFT}}(x, y_{\text{gt}}) \tag{8}$$

Table 1: **Quantitative comparison on rigging result.** * denotes models trained on Art-XL2.0 and tested on MR. MagicArticulate and UniRig cannot be trained on the MR dataset as their training script is not provided. indicate the best, second best, and third best performance respectively.

Method	Dataset	CD-J2J↓	CD-J2B↓	CD-B2B↓	IoU↑	Prec.↑	Rec.↑
RigNet	Art-XL2.0	7.587%	6.347%	6.366%	21.055%	21.015%	33.135%
Magic-hier		3.435%	2.757%	2.393%	76.364%	78.121%	77.567%
UniRig		3.232%	2.540%	2.124%	75.571%	77.334%	77.323%
Magic-spatial		3.041%	2.479%	2.099%	78.675%	80.026%	80.085%
Ours		**2.572%**	**2.030%**	**1.683%**	**82.806%**	**85.254%**	**82.918%**
RigNet	MR	6.375%	5.115%	5.245%	31.034%	24.034%	50.002%
Magic-hier*		4.119%	3.155%	2.780%	63.382%	57.889%	73.680%
UniRig*		3.797%	2.888%	2.437%	62.832%	56.230%	76.419%
Magic-spatial*		3.920%	3.138%	2.712%	63.831%	57.735%	75.667%
Ours*		3.735%	2.816%	2.362%	66.488%	62.655%	75.059%
Ours		**3.203%**	**2.436%**	**2.046%**	**73.108%**	**73.965%**	**76.795%**

3.3 Skinning Weight Prediction

The fundamental limitation of existing skinning methods [33, 46, 71] lies in their static bone selection paradigm—they precompute vertex-bone distances to permanently select the k-nearest bones, and assume that the vertex is influenced only by these bones. In contrast, we present a plug-and-play geodesic-aware bone probability prediction module that dynamically identifies the k most probable influencing bones conditioned on implicit geodesic features.

Geodesic-aware Bone Probability Prediction Module. The inputs include the vertex positions, the vertex normal, the coordinates of joints, and the vertex-bone geometric distances computed from both raw and laplacian-smoothed meshes. These attributes are processed through a three-layer MLP to predict the influence probabilities of each bone for every vertex, and the k highest-probability bones are selected. To optimize bone selection, we reframe the issue of whether the bone b_j impacts the vertex v_i as a binary classification problem. Here, a label of 1 signifies influence, while a label of 0 indicates no influence. Thus, this module minimizes the discrepancy between the chosen bone and the actual bone using Binary-Cross-Entropy loss: $\mathcal{L}_{\text{S}} = \sum_{i=1}^{N} \left(-\hat{b} \log\left(b\right) - \left(1 - \hat{b}\right) \log\left(1 - b\right) \right)$, where $\hat{b}$ represents the ground truth labels, b denotes the predicted probabilities, and N is the number of vertices. For more details, please refer to the appendix A.1.

Skinning Weight Prediction Module. To highlight the plug-and-play nature of our bone probability prediction module, we integrate it with existing skinning methods. We consider the skinning weight matrix as the probability of each vertex binding to each bone. Thus, this module minimizes the discrepancy between the predicted skinning weights distribution and the actual distribution using Kullback-Leibler divergence loss: $\mathcal{L}_{\text{D}} = \sum_{i=1}^{N} \sum_{j=1}^{B} w_{ij} \left(\log \frac{w_{ij}}{\hat{w}_{ij}} \right)$, where $\hat{w}_{ij}$ is the ground truth, w_{ij} is the predicted skinning weights, and B is the number of bones. Finally, the total loss for the skinning weight prediction stage is: $\mathcal{L}_{\text{stage3}} = \mathcal{L}_{\text{S}} + \mathcal{L}_{\text{D}}$.

4 Experiments

4.1 Implementation Details

Dataset. We evaluated our model on Articulation-XL2.0 (Art-XL2.0) [38] and ModelsResource (MR) [33]. Art-XL2.0 provides 46k samples for training and 2k samples for testing, while MR dataset contains 2.1k training samples and 540 testing samples.

Data augmentation. To enhance model robustness and generalization, we applied a comprehensive data augmentation strategy during the rigging pre-training stage. This includes the following components: 1) random mesh translations within $[-0.3, 0.3]$, 2) random axial rotations, 3) non-uniform scaling along each axis to introduce variations in proportions, and 4) bone perturbation, where a randomly selected bone is rotated by an angle sampled from a gaussian distribution $N\left(0, 25^{\circ}\right)$.

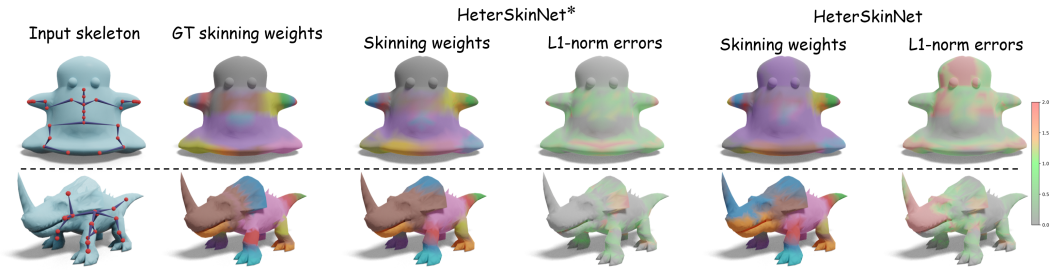

Figure 5: **Qualitative comparison of skinning result on Art-XL2.0 (top) and MR (bottom).** Models marked with * were trained using our geodesic-aware bone probability prediction module, which effectively mitigates the L1-norm error and enhances skinning performance. Additional results are provided in the appendix B.2.

Table 2: **Quantitative comparison on skinning result.** Models marked with * were trained using our geodesic-aware bone probability prediction module. Results for the MR dataset are provided in the appendix B.2.

Method	Dataset	Prec.↑	Rec.↑	avg L1↓	avg Dist↓
RigNet	Art-XL2.0	87.47%	56.04%	0.52	0.0090
RigNet*		**87.67%**	**59.98%**	**0.44**	**0.0079**
NeuroSkinning		87.05%	55.51%	0.52	0.0089
NeuroSkinning*		**88.00%**	**61.31%**	**0.45**	**0.0080**
HeterSkinNet		88.16%	60.18%	0.43	0.0079
HeterSkinNet*		**89.38%**	**61.13%**	**0.42**	**0.0075**

Training details. The rigging pre-training stage uses a batch size of 192, lasting 2 days on the Art-XL2.0 dataset and 10 hours on the MR dataset, while the post-training stage runs 5 epochs on 14k curated preference pairs. For the skinning weight prediction stage, we set $k = 6$ follow the baseline. Training uses a batch size of 80, lasting 1 day for the Art-XL2.0 dataset and 10 hours for the MR dataset. See the appendix A.2 for more details.

4.2 Metrics and Baselines

Metrics. Consistent with RigNet [33], we evaluate rigging results using CD-J2J (Chamfer Distance between Joints), CD-J2B (Chamfer Distance between Joints and Bones), CD-B2B (Chamfer Distance between Bones), IoU (Intersection over Union), Precision, and Recall. For skinning results, we adopt Precision, Recall, L1-norm error, and distance error to comprehensively assess bone identification accuracy, skinning weight precision, and deformation quality.

Baselines. For rigging results, we compare our approach against state-of-the-art approaches, including RigNet [33], MagicArticulate [38], and UniRig [39]. MagicArticulate is evaluated using both its proposed hierarchical (Magic-hier) and spatial (Magic-spatial) sequence orders. Since RigAnything [40] does not share its code, we cannot compare to it. For skinning results, we integrate our geodesic-aware bone probability prediction module with three top-k-based skinning methods—RigNet [33], NeuroSkinning [71], and HeterSkinNet [46]—to demonstrate its compatibility and effectiveness. All vertex-to-bone geodesic distance computations adhere to the HollowDist proposed in the HeterSkinNet, with GPU-accelerated 256-resolution voxelization [72].

4.3 Comparison

Rigging Comparison. As quantitatively shown in 1, our method achieves comprehensive improvements across all metrics on both datasets. For skeleton location accuracy, we outperform Magic-spatial by +4.2% IoU, +5.2% Precision, and +2.9% Recall on Art-XL2.0, while reducing 15% by CD-J2J, 18% by CD-J2B, and 20% by CD-B2B. The improvements over RigNet are even more pronounced. In terms of topological integrity, as qualitatively shown in Figure 4, MagicArticulate's representation fails to capture the inherent topological relationships within skeletal structures, often

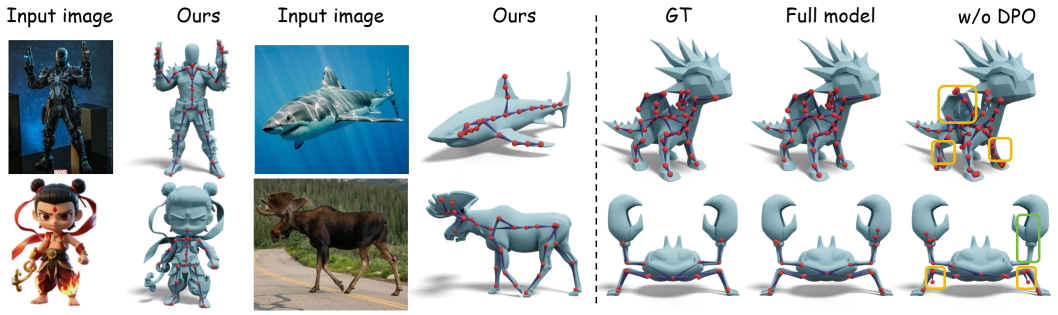

Figure 6: **(Left) Rigging results on mesh from in-the-wild images.** We use off-the-shelf image-to-3D model Hunyuan3D 2.5 [3] to generated meshes from input images. **(Right) Ablation study on DPO.** The proposed reward-guided DPO learns human preferences to produce skeletons that better align with artistic aesthetics.

Table 4: **Ablation study on DPO**. Results for the MR dataset are provided in the appendix B.3.

Method	Dataset	CD-J2J↓	CD-J2B↓	CD-B2B↓	IoU↑	Prec.↑	Rec.↑
w/o DPO	Art-XL2.0	2.695%	2.109%	1.750%	82.183%	84.234%	82.734%
Ours		**2.572%**	**2.030%**	**1.683%**	**82.806%**	**85.254%**	**82.918%**

resulting in unconnected joints and discontinuous skeletons (highlighted in orange boxes). UniRig struggles with insufficient spatial continuity between sequentially generated bone chains. Specifically, the large gap between the terminal joint of a completed chain and the expected starting position of the next chain creates initialization barriers, leading to incomplete skeletal structures with missing bone chains (highlighted in green boxes), particularly for characters with tails or wings. MagicArticulate also suffers from similar shortcomings. RigNet, on the other hand, frequently generates an excessive number of joints. In contrast, our method consistently produces accurate, well-structured, and coherent skeletons that closely align with the shapes across diverse categories.

Skinning Comparison. Table 2 quantitatively compares baseline performance with and without incorporating the proposed geodesic-aware bone probability prediction module. The results demonstrate the effectiveness of our method in accurately identifying influential bones, reducing L1-norm errors, and preventing incorrect deformations during motion. Furthermore, Figure 5 presents a qualitative comparison of the per-vertex L1-norm errors and predicted skinning weights. The method with the module shows its capability to produce more reliable skinning weights that closely match the ground truth and improve animation fidelity.

4.4 Ablation Studies

To validate the effectiveness of the key components of our method, we conducted a series of ablation studies targeting: 1) the proposed connectivity-preserving tokenization strategy, 2) the data augmentation strategy, and 3) the reward-guided DPO post-training strategy.

Effectiveness of the tokenization strategy. Table 3 compares the average token sequence length between our method against baselines. Compared to MagicArticulate, our method reduces sequence length by 26% by eliminating redundant parent joint encoding. For UniRig, despite similar compression efficiency, it ignores skeletal topology and relies on manually designed heuristic rules to determine the connections between different bone chains, limiting its generalization and resulting in suboptimal topology quality. In contrast, our method explicitly encodes skeletal topology and connectivity, ensuring better generalization.

Table 3: **Comparison of different tokenization strategies.**

Method	Dataset	avg Tokens↓
MagicArticulate	Art-XL2.0	201.00
UniRig		**140.26**
Ours		142.01

Effectiveness of the DPO post-training. We compared the post-trained model with the pre-trained model to assess the impact of reward-guided DPO. Table 4 quantitatively demonstrates that DPO-enhanced results have achieve higher skeleton location accuracy. Additionally, the right side of Figure 6 qualitatively highlights the topological connectivity improvements introduced by our

topology-aware reward function. Models without DPO often suffer from missing skeletal elements, such as the crab's right claw (highlighted in green boxes), or exhibit disorganized structures, such as wings, feet, and the crab's legs (highlighted in orange boxes). In contrast, the full model produces more complete and structured results, aligning well with artistic preferences.

Effectiveness of the data augmentation. Figure 7 compares results with and without our data augmentation strategy on a character in a walking pose not present in the dataset. Our full model generates skeletons that are better aligned with the character's shape, whereas the model trained without augmentation struggles. Furthermore, as shown on the left side of Figure 6, our model effectively handles cartoon characters and in-the-wild data, highlighting the augmentation's role in improving generalization to unseen poses and data types.

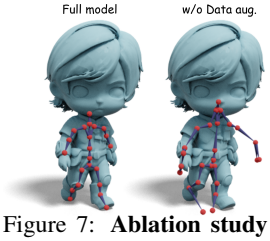

Figure 7: **Ablation study on data augmentation.**

5 Conclusion

We present Auto-Connect, a method that transforms static 3D meshes into animation-ready assets. By integrating connectivity-preserving tokenization, reward-guided fine-tuning, and geodesic-aware bone selection, our approach achieves exceptional rigging and skinning performance across diverse categories of 3D meshes. We believe Auto-Connect holds great potential to revolutionize 3D content creation, offering a more efficient solution for digital artists by streamlining animation workflows. Limitations and future work are discussed in appendix C.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62202174, in part by the GJYC program of Guangzhou under Grant 2024D01J0081, and in part by the ZJ program of Guangdong under Grant 2023QN10X455, and in part by the Fundamental Research Funds for the Central Universities under Grant 2025ZYGXZR053.

References

[1] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *arXiv preprint arXiv:2301.11445*, 2023.

[2] Xianglong He, Zi-Xin Zou, Chia-Hao Chen, Yuan-Chen Guo, Ding Liang, Chun Yuan, Wanli Ouyang, Yan-Pei Cao, and Yangguang Li. Sparseflex: High-resolution and arbitrary-topology 3d shape modeling. *arXiv preprint arXiv:2503.21732*, 2025.

[3] Tencent Hunyuan3D Team. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

[4] Jinnan Chen, Lingting Zhu, Zeyu Hu, Shengju Qian, Yugang Chen, Xin Wang, and Gim Hee Lee. Mar-3d: Progressive masked auto-regressor for high-resolution 3d generation. *arXiv preprint arXiv:2503.00000*, 2025.

[5] Xin Yu, Ze Yuan, Yuan-Chen Guo, Ying-Tian Liu, Jianhui Liu, Yangguang Li, Yan-Pei Cao, Ding Liang, and Xiaojuan Qi. Texgen: a generative diffusion model for mesh textures. *ACM Trans. Graph.*, 43(6), 2024.

[6] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *arXiv preprint arXiv:2502.06608*, 2025.

[7] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *arXiv preprint arXiv:2406.13897*, 2024.

[8] Zhengming Yu, Zhiyang Dou, Xiaoxiao Long, Cheng Lin, Zekun Li, Yuan Liu, Norman Müller, Taku Komura, Marc Habermann, Christian Theobalt, et al. Surf-d: High-quality surface generation for arbitrary topologies using diffusion models. *arXiv preprint arXiv:2311.17050*, 2023.

[9] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.

[10] Xiaoyu Xiang, Liat Sless Gorelik, Yuchen Fan, Omri Armstrong, Forrest Iandola, Yilei Li, Ita Lifshitz, and Rakesh Ranjan. Make-a-texture: Fast shape-aware texture generation in 3 seconds. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4872–4881. IEEE, 2025.

[11] Haoyu Wu, Meher Gitika Karumuri, Chuhang Zou, Seungbae Bang, Yuelong Li, Dimitris Samaras, and Sunil Hadap. Direct and explicit 3d generation from a single image. *arXiv preprint arXiv:2411.10947*, 2024.

[12] Yangyi Huang, Hongwei Yi, Yuliang Xiu, Tingting Liao, Jiaxiang Tang, Deng Cai, and Justus Thies. TeCH: Text-guided Reconstruction of Lifelike Clothed Humans. In *International Conference on 3D Vision (3DV)*, 2024.

[13] Tingting Liao, Hongwei Yi, Yuliang Xiu, Jiaxaing Tang, Yangyi Huang, Justus Thies, and Michael J. Black. Tada! text to animatable digital avatars. *arXiv preprint arXiv:2308.10899*, 2023.

[14] Jinnan Chen, Chen Li, Jianfeng Zhang, Lingting Zhu, Buzhen Huang, Hanlin Chen, and Gim Hee Lee. Generalizable human gaussians from single-view image. *arXiv preprint arXiv:2406.06050*, 2024.

[15] Jinnan Chen, Chen Li, and Gim Hee Lee. Dihur: Diffusion-guided generalizable human reconstruction. *arXiv preprint arXiv:2411.11903*, 2025.

[16] Shoukang Hu, Takuya Narihira, Kazumi Fukuda, Ryosuke Sawata, Takashi Shibuya, and Yuki Mitsufuji. Humangif: Single-view human diffusion with generative prior. *arXiv preprint arXiv:2502.12080*, 2025.

[17] 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 (CVPR)*, June 2023.

[18] Yiyu Zhuang, Jiaxi Lv, Hao Wen, Qing Shuai, Ailing Zeng, Hao Zhu, Shifeng Chen, Yujiu Yang, Xun Cao, and Wei Liu. Idol: Instant photorealistic 3d human creation from a single image. *arXiv preprint arXiv:2412.14963*, 2024.

[19] Shoukang Hu, Takuya Narihira, Kazumi Fukuda, Ryosuke Sawata, Takashi Shibuya, and Yuki Mitsufuji. Humangif: Single-view human diffusion with generative prior. *arXiv preprint arXiv:2502.12080*, 2025.

[20] Shoukang Hu, Fangzhou Hong, Tao Hu, Liang Pan, Haiyi Mei, Weiye Xiao, Lei Yang, and Ziwei Liu. Humanliff: Layer-wise 3d human generation with diffusion model. *arXiv preprint*, 2023.

[21] Yuhan Wang, Fangzhou Hong, Shuai Yang, Liming Jiang, Wayne Wu, and Chen Change Loy. Meat: Multiview diffusion model for human generation on megapixels with mesh attention. *arXiv preprint arXiv:2503.08664*, 2025.

[22] Peng Li, Wangguandong Zheng, Yuan Liu, Tao Yu, Yangguang Li, Xingqun Qi, Mengfei Li, Xiaowei Chi, Siyu Xia, Wei Xue, et al. Pshuman: Photorealistic single-view human reconstruction using cross-scale diffusion. *arXiv preprint arXiv:2409.10141*, 2024.

[23] Haoyu Chen, Hao Tang, Ehsan Adeli, and Guoying Zhao. Towards robust 3d pose transfer with adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2295–2304, 2024.

[24] Chaoyue Song, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. Unsupervised 3d pose transfer with cross consistency and dual reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10488–10499, 2023.

[25] Jinnan Chen, Chen Li, and Gim Hee Lee. Weakly-supervised 3d pose transfer with keypoints. *arXiv preprint arXiv:2307.13459*, 2023.

[26] Zijie Ye, Jia-Wei Liu, Jia Jia, Shikun Sun, and Mike Zheng Shou. Skinned motion retargeting with dense geometric interaction perception. *Advances in Neural Information Processing Systems*, 37:125907–125934, 2024.

[27] Inbar Gat, Sigal Raab, Guy Tevet, Yuval Reshef, Amit H Bermano, and Daniel Cohen-Or. Anytop: Character animation diffusion with any topology. *arXiv preprint arXiv:2502.17327*, 2025.

[28] Zhiyang Dou, Xuelin Chen, Qingnan Fan, Taku Komura, and Wenping Wang. C·ase: Learning conditional adversarial skill embeddings for physics-based characters. *arXiv preprint arXiv:2309.11351*, 2023.

[29] Haoyu Chen, Hao Tang, Radu Timofte, Luc V Gool, and Guoying Zhao. Lart: Neural correspondence learning with latent regularization transformer for 3d motion transfer. *Advances in Neural Information Processing Systems*, 36:43742–43753, 2023.

[30] Yiming Huang, Zhiyang Dou, and Lingjie Liu. Modskill: Physical character skill modularization. *arXiv preprint arXiv:2502.14140*, 2025.

[31] Rong Wang, Wei Mao, Changsheng Lu, and Hongdong Li. Towards high-quality 3d motion transfer with realistic apparel animation. In *European Conference on Computer Vision*, pages 35–51. Springer, 2024.

[32] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. *ACM Transactions on graphics (TOG)*, 26(3):72–es, 2007.

[33] Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. Rignet: Neural rigging for articulated characters. *arXiv preprint arXiv:2005.00559*, 2020.

[34] Hao Zhang, Fang Li, Samyak Rawlekar, and Narendra Ahuja. S3o: A dual-phase approach for reconstructing dynamic shape and skeleton of articulated objects from single monocular video. *arXiv preprint arXiv:2405.12607*, 2024.

[35] Hao Zhang, Di Chang, Fang Li, Mohammad Soleymani, and Narendra Ahuja. Magicpose4d: Crafting articulated models with appearance and motion control. *arXiv preprint arXiv:2405.14017*, 2025.

[36] Hao Zhang, Fang Li, Samyak Rawlekar, and Narendra Ahuja. Learning implicit representation for reconstructing articulated objects. *arXiv preprint arXiv:2401.08809*, 2024.

[37] Zimeng Wang, Zhiyang Dou, Rui Xu, Cheng Lin, Yuan Liu, Xiaoxiao Long, Shiqing Xin, Taku Komura, Xiaoming Yuan, and Wenping Wang. Coverage Axis++: Efficient Inner Point Selection for 3D Shape Skeletonization. *Computer Graphics Forum*, 2024.

[38] Chaoyue Song, Jianfeng Zhang, Xiu Li, Fan Yang, Yiwen Chen, Zhongcong Xu, Jun Hao Liew, Xiaoyang Guo, Fayao Liu, Jiashi Feng, et al. Magicarticulate: Make your 3d models articulation-ready. *arXiv preprint arXiv:2502.12135*, 2025.

[39] Jia-Peng Zhang, Cheng-Feng Pu, Meng-Hao Guo, Yan-Pei Cao, and Shi-Min Hu. One model to rig them all: Diverse skeleton rigging with unirig. *arXiv preprint arXiv:2504.12451*, 2025.

[40] Isabella Liu, Zhan Xu, Wang Yifan, Hao Tan, Zexiang Xu, Xiaolong Wang, Hao Su, and Zifan Shi. Riganything: Template-free autoregressive rigging for diverse 3d assets. *arXiv preprint arXiv:2502.09615*, 2025.

[41] Jing Ma and Dongliang Zhang. Tarig: Adaptive template-aware neural rigging for humanoid characters. *Computers & Graphics*, 114:158–167, 2023.

[42] Mingze Sun, Junhao Chen, Junting Dong, Yurun Chen, Xinyu Jiang, Shiwei Mao, Puhua Jiang, Jingbo Wang, Bo Dai, and Ruqi Huang. Drive: Diffusion-based rigging empowers generation of versatile and expressive characters. *arXiv preprint arXiv:2411.17423*, 2024.

[43] Zedong Chu, Feng Xiong, Meiduo Liu, Jinzhi Zhang, Mingqi Shao, Zhaoxu Sun, Di Wang, and Mu Xu. Humanrig: Learning automatic rigging for humanoid character in a large scale dataset. *arXiv preprint arXiv:2412.02317*, 2024.

[44] Zhiyang Guo, Jinxu Xiang, Kai Ma, Wengang Zhou, Houqiang Li, and Ran Zhang. Make-it-animatable: An efficient framework for authoring animation-ready 3d characters. *arXiv preprint arXiv:2411.18197*, 2024.

[45] Zhan Xu, Yang Zhou, Evangelos Kalogerakis, and Karan Singh. Predicting animation skeletons for 3d articulated models via volumetric nets. In *2019 international conference on 3D vision (3DV)*, pages 298–307. IEEE, 2019.

[46] Xiaoyu Pan, Jiancong Huang, Jiaming Mai, He Wang, Honglin Li, Tongkui Su, Wenjun Wang, and Xiaogang Jin. Heterskinnet: A heterogeneous network for skin weights prediction. In *Proceedings of the ACM on computer graphics and interactive techniques*, volume 4. Association for Computing Machinery, 2021.

[47] Albert Mosella-Montoro and Javier Ruiz-Hidalgo. Skinningnet: Two-stream graph convolutional neural network for skinning prediction of synthetic characters. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18593–18602, 2022.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[49] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[50] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[51] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11315–11325, 2022.

[52] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[53] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

[54] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024.

[55] Xu Ma, Peize Sun, Haoyu Ma, Hao Tang, Chih-Yao Ma, Jialiang Wang, Kunpeng Li, Xiaoliang Dai, Yujun Shi, Xuan Ju, Yushi Hu, Artsiom Sanakoyeu, Felix Juefei-Xu, Ji Hou, Junjiao Tian, Tao Xu, Tingbo Hou, Yen-Cheng Liu, Zecheng He, Zijian He, Matt Feiszli, Peizhao Zhang, Peter Vajda, Sam Tsai, and Yun Fu. Token-shuffle: Towards high-resolution image generation with autoregressive models. *arXiv preprint arXiv:2504.17789*, 2025.

[56] Shengwen Peng, Ronghui You, Hongning Wang, Chengxiang Zhai, Hiroshi Mamitsuka, and Shanfeng Zhu. Deepmesh: deep semantic representation for improving large-scale mesh indexing. *Bioinformatics*, 32(12):i70–i79, 2016.

[57] Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, et al. Meshanything: Artist-created mesh generation with autoregressive transformers. *arXiv preprint arXiv:2406.10163*, 2024.

[58] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19615–19625, 2024.

[59] Haohan Weng, Zibo Zhao, Biwen Lei, Xianghui Yang, Jian Liu, Zeqiang Lai, Zhuo Chen, Yuhong Liu, Jie Jiang, Chunchao Guo, et al. Scaling mesh generation via compressive tokenization. *arXiv preprint arXiv:2411.07025*, 2024.

[60] Jiaxiang Tang, Zhaoshuo Li, Zekun Hao, Xian Liu, Gang Zeng, Ming-Yu Liu, and Qinsheng Zhang. Edgerunner: Auto-regressive auto-encoder for artistic mesh generation. *arXiv preprint arXiv:2409.18114*, 2024.

[61] Stefan Lionar, Jiabin Liang, and Gim Hee Lee. Treemeshgpt: Artistic mesh generation with autoregressive tree sequencing. *arXiv preprint arXiv:2503.11629*, 2025.

[62] Zixuan Liu, Xiaolin Sun, and Zizhan Zheng. Enhancing llm safety via constrained direct preference optimization. *arXiv preprint arXiv:2403.02475*, 2024.

[63] Shuaijie She, Wei Zou, Shujian Huang, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. Mapo: Advancing multilingual reasoning through multilingual alignment-as-preference optimization. *arXiv preprint arXiv:2401.06838*, 2024.

[64] Lei Li, Zhihui Xie, Mukai Li, Shunian Chen, Peiyi Wang, Liang Chen, Yazheng Yang, Benyou Wang, and Lingpeng Kong. Silkie: Preference distillation for large visual language models. *arXiv preprint arXiv:2312.10665*, 2023.

[65] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*, 2023.

[66] Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. *arXiv preprint arXiv:2402.11411*, 2024.

[67] Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems*, 36:10935–10950, 2023.

[68] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[69] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[70] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *Advances in neural information processing systems*, 36:73969–73982, 2023.

[71] Lijuan Liu, Youyi Zheng, Di Tang, Yi Yuan, Changjie Fan, and Kun Zhou. Neuroskinning: Automatic skin binding for production characters with deep graph networks. *ACM Transactions on Graphics (ToG)*, 38(4):1–12, 2019.

[72] Jeroen Baert. Cuda voxelizer: A gpu-accelerated mesh voxelizer. https://github.com/Forceflow/cuda_voxelizer, 2017.

[73] Blender Foundation. Blender - a 3d modelling and rendering software, 2024. Version 3.6.

[74] Autodesk Inc. Autodesk maya, 2024. Version 2024.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have clearly stated the claims made in the abstract and introduction, accurately reflecting the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: We have discussed the limitations of our work in the appendix, which include the inability to generalize well to snake-like data due to the lack of such samples in the training dataset.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: Our paper does not include theoretical results, and therefore, this question is not applicable to our work.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

Justification: We provide a detailed description of the model and experimental settings in our paper, ensuring that readers have the necessary information to reproduce the main experimental results. Additionally, we plan to release the code to further enhance reproducibility. and facilitate verification of our results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: While we currently do not provide open access to the data and code, we plan to release the code along with sufficient instructions to reproduce the main experimental results after the paper has been accepted.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided all the necessary details regarding the training and testing process, including data splits, network structure, hyperparameters, and the type of optimizer used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We conducted our experiments and baseline experiments on the same training and testing datasets to ensure a fair comparison.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provided sufficient information on the computer resources needed to reproduce the experiments in the implementation details section.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: Our research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We discussed the social impact of this work in the introduction and conclusion sections.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We will release the code, data, and models publicly upon the acceptance of the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We did not submit any new assets at the time of submission. However, we plan to release well-documented code after the paper's acceptance.

    Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in our research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

Appendix

In this appendix, we provide additional content to complement the main manuscript, including:

- Further details of Auto-Connect (Section A).
- Additional experimental results on rigging and skinning (Section B).
- A discussion of the limitations of our work and future works (Section C).

A More Details of Auto-Connect

A.1 More Details of Geodesic-aware Bone Probability Prediction Module

This module combining heterogeneous features, including vertex positions p_v, vertex normal n_v, the coordinate of the bone start and end joints $\left(p_{b_j^s}, p_{b_j^e}\right)$, and the vertex-bone geometric distances computed from both raw and laplacian-smoothed meshes. These features are processed by an three-layer MLP to predict bone influence probabilities:

$$\tilde{b} = \text{top\_k}\Big(\text{MLP}\big(p_v, n_v, p_{b_j^s}, p_{b_j^e}, d_v^{b_j}, d_{v_{l5}}^{b_j}, d_{v_{l10}}^{b_j} \big| b_j \in \mathcal{B}\big)\Big) \tag{9}$$

where $d_{v_{l5}}^{b_j}$ and $d_{v_{l10}}^{b_j}$ denote distances after performing 5/10 laplacian smoothing iterations, respectively. The resulting selected bone set $\tilde{b}$ for each vertex v is used as the input to the skinning weight prediction module to compute the final skinning weights.

A.2 More Training Details

The rigging pre-training stage is conducted with a global batch size of 192, lasting 2 days on the Art-XL2.0 dataset and 10 hours on the MR dataset. We use the Adam optimizer with a base learning rate of 5×10^{-5}, a weight decay of 0.001, and a linear warmup for the first 1,000 steps. For the DPO post-training stage, the optimizer remains unchanged, but the learning rate is reduced to 1×10^{-6}, and the coefficient for $\mathcal{L}_{\text{SFT}}$ is set to $\lambda = 1$. This stage performs 5 epochs on 14k curated preference pairs. The RigFormer model consists of 24 layers with a hidden dimension of 1024, and each transformer block incorporates a 16-head multi-head self-attention mechanism.

For skinning weight prediction, following the baseline, we set $k = 6$ nearest bones and prune non-influential joints during ground truth construction. Training is conducted on $8 \times$ H20 GPUs with a global batch size of 80, lasting 1 day for the Art-XL2.0 dataset and 10 hours for the MR dataset. The geodesic-aware bone probability prediction module is implemented as a three-layer MLP with a hidden dimension of 256 and ReLU activation.

For the baselines MagicArticulate [38] and UniRig [39], we utilize their publicly available pre-trained weights on the Art-XL2.0 dataset for comparison since their training scripts are not provided. All other baselines [33, 46, 71] are retrained on the same datasets for a fair comparison.

A.3 Animation Details

Auto-Connect provides an automated animation pipeline. The resulting animation-ready assets can be exported in standard formats such as FBX and GLB. These assets are directly compatible with popular animation software like Blender [73] and Autodesk Maya [74], enabling digital artists to edit and refine them. Animation videos are included in the supplementary materials.

B Additional Experimental Results

B.1 More Rigging Result

We provide additional qualitative rigging results on both Articulation-XL and ModelsResource datasets. As illustrated in Fig. 11, our method consistently generates high-quality skeletons, even

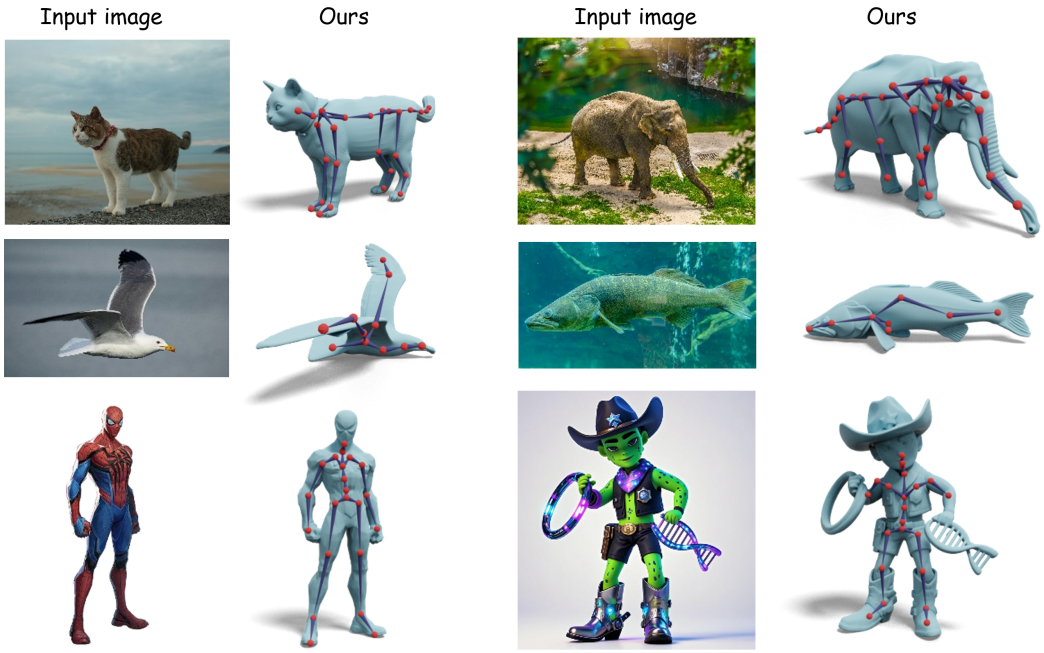

Figure 8: **More rigging results on mesh from in-the-wild images.** We use off-the-shelf image-to-3D model Hunyuan3D 2.5 [3] to generated mesh from the input images. The rigging results demonstrate that our model has strong generalization to unseen data, achieving artist-approved skeleton quality and transforming static 3D meshes into animation-ready assets.

Table 5: **Quantitative comparison of skinning result on ModelsResource dataset.** Models marked with * were trained using our geodesic-aware bone probability prediction module.

Method	Dataset	Prec.↑	Rec.↑	avg L1↓	avg Dist↓
RigNet	MR	86.03%	79.03%	0.36	0.0058
RigNet*		**87.85%**	**80.11%**	**0.33**	**0.0049**
NeuroSkinning		86.24%	78.31%	0.36	0.0057
NeuroSkinning*		**88.03%**	**79.27%**	**0.33**	**0.0049**
HeterSkinNet		87.31%	78.99%	0.34	0.0052
HeterSkinNet*		**89.09%**	**79.45%**	**0.28**	**0.0045**

for complex cases. In contrast, the baseline methods produce suboptimal results that are not directly suitable for animation pipelines. Fig. 8 showcases more rigging results on AI-generated 3D data, further validating the robustness and effectiveness of our approach.

B.2 More Skinning Results

Table 5 presents a quantitative comparison of baseline performance with and without incorporating this module on the ModelsResource dataset. The results demonstrate substantial improvements across all evaluation metrics, including higher precision, higher recall, lower L1-norm error, and reduced distance error. This highlights the effectiveness of our approach in enhancing skinning accuracy. In addition, Fig. 12 illustrates qualitative comparisons, where the integration of our module enables the baseline to produce smoother and more realistic skin deformations, particularly in regions with complex geometries. These results further highlight the module's ability to accurately identify influence bones, leading to more precise predictions of skinning weights with minimal L1-norm error.

B.3 More Ablation Studies

More Ablation study on DPO post-training. Table 6 presents a quantitative comparison of model performance with and without the proposed DPO post-training on the ModelsResource

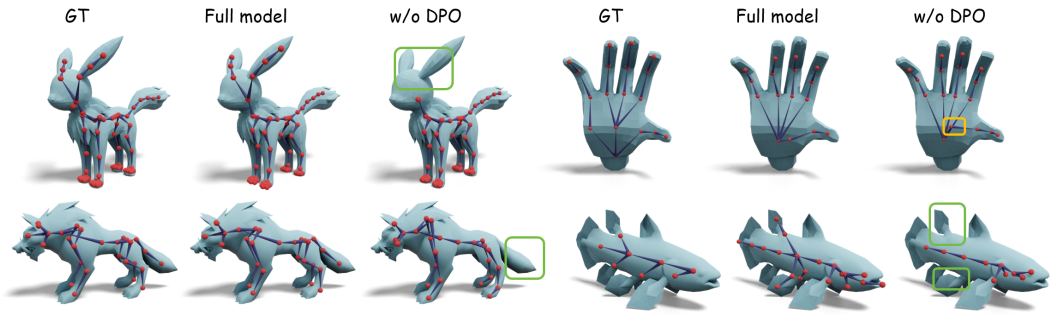

Figure 9: **More qualitative ablation study on DPO post-training.** Models trained without DPO often suffer from missing details—such as ears, tails, or fins (highlighted in green boxes)—or generate structural artifacts like crossing topologies (highlighted in orange boxes). In contrast, our full model effectively alleviates these issues, producing well-defined skeletons that better align with the artistic aesthetics expected by creators.

Table 6: **Quantitative ablation study of DPO post-training on ModelsResource dataset.**

Method	Dataset	CD-J2J↓	CD-J2B↓	CD-B2B↓	IoU↑	Prec.↑	Rec.↑
w/o DPO	MR	3.426%	2.576%	2.164%	72.213%	73.700%	72.822%
Ours		**3.203%**	**2.436%**	**2.046%**	**73.108%**	**73.965%**	**76.795%**

dataset. The results clearly demonstrate that our reward-guided DPO post-training substantially improves the model's accuracy in skeleton localization, highlighting the effectiveness of the proposed approach. In addition, Fig. 9 showcases additional qualitative evidence from ablation studies. Our full model produces more realistic topology connections and generates more complete skeletons, closely resembling the ground truth.

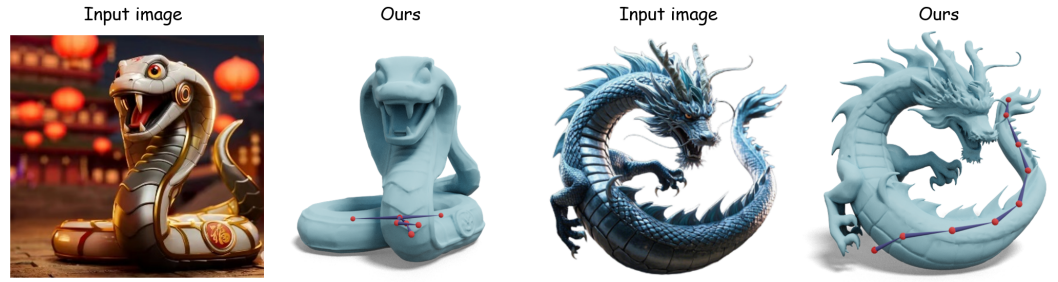

Figure 10: **Some failure cases.** Our method struggles with snake-like data, the input meshs are generated using Hunyuan3D 2.5 [3].

C Limitations and Future Work

While our method achieves significant improvements in skeleton location accuracy, topological consistency, and skinning quality compared to prior approaches, it still has some limitations. As shown in Fig. 10, the main drawback is the inability to generalize well to snake-like data, due to the lack of such samples in the training dataset. Future work could address this by expanding the dataset with more snake-like examples or by developing more robust data augmentation techniques. Another promising direction is leveraging multimodal input, such as text, image, and video, to allow user-friendly editing of rigging results.

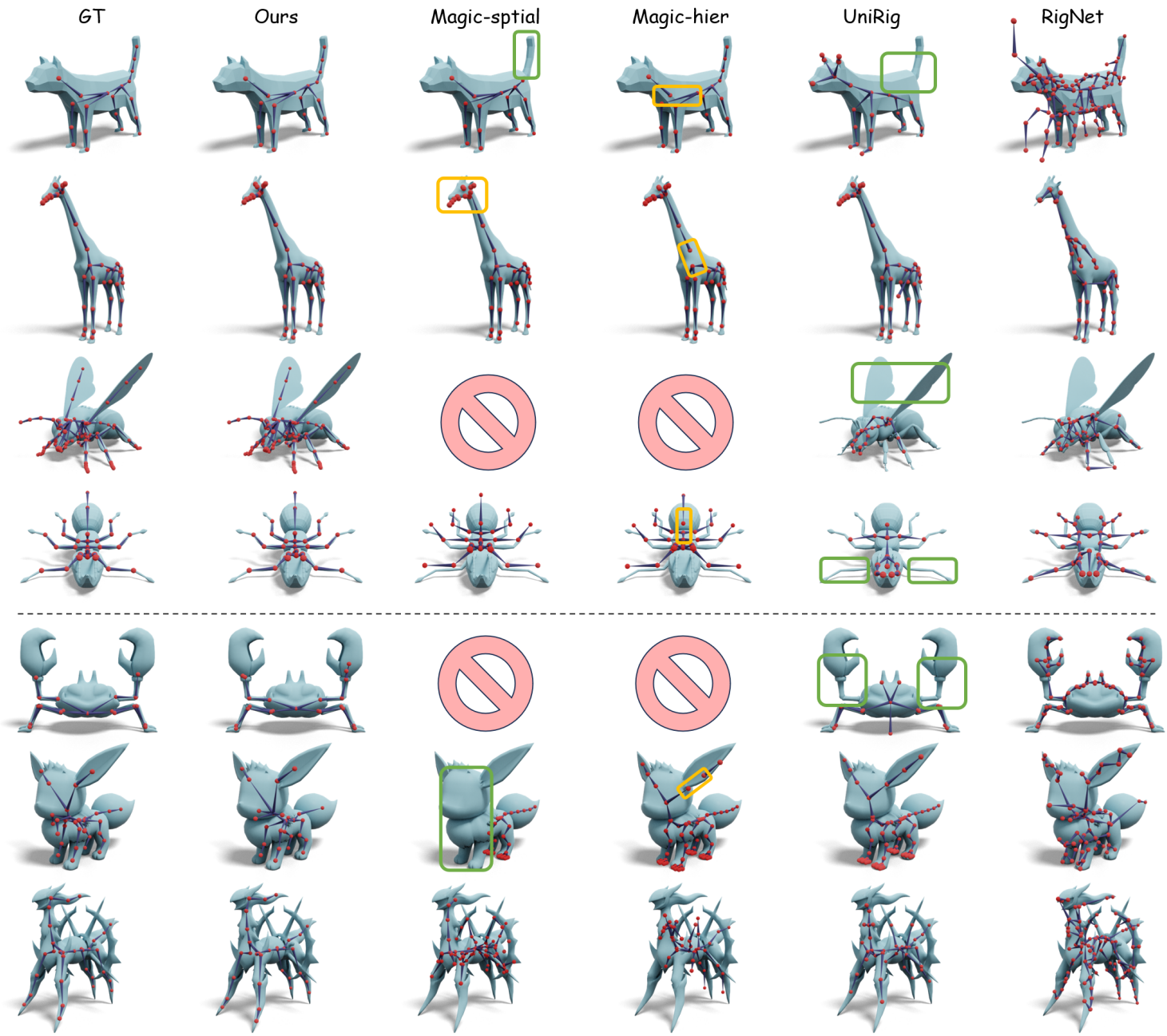

Figure 11: **More qualitative comparison of rigging results on Art-XL2.0 (top) and ModelsResource (bottom).** MagicArticulate often generates discontinuous skeletons (highlighted in green boxes) or even fails entirely. UniRig tends to predict the termination token prematurely, leading to incomplete skeletal structures with missing bone chains (highlighted in orange boxes), and RigNet frequently produces overly dense joints, resulting in disorganized skeleton topologies. In contrast, our method reliably generates coherent, accurate, and well-structured skeletons that closely conform to the shapes.

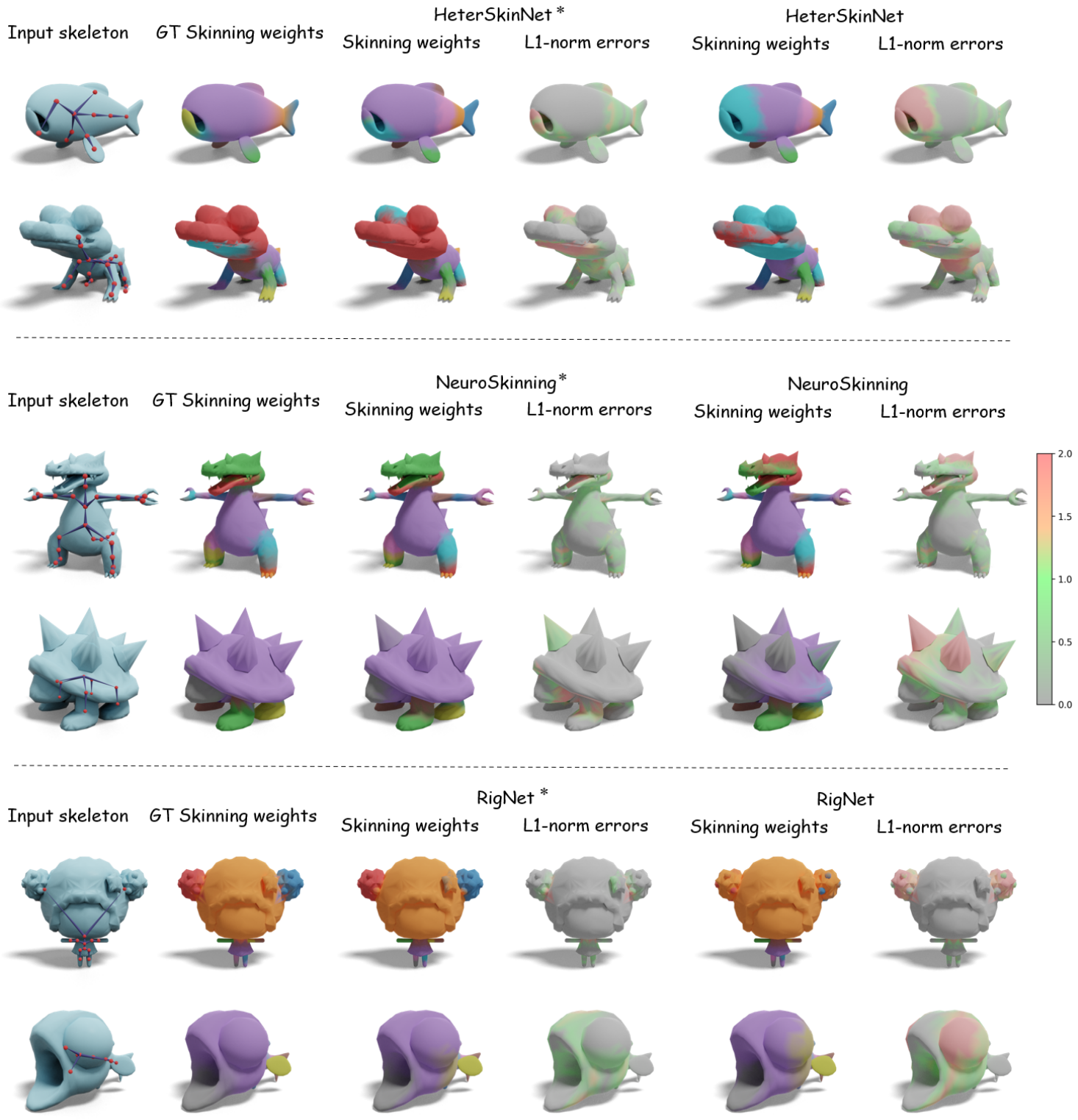

Figure 12: **More qualitative comparison of skinning result on ModelsResource dataset.** Models marked with * were trained using our geodesic-aware bone probability prediction module. By incorporating this module, the baseline method achieves lower L1-norm errors and more precise skinning weights, resulting in more accurate deformations during animation.