ARTIST: *A*daptive Humanoid *R*igging by *T*ransferring *I*ndividual *S*tyle with Optimal *T*ransport

Jeanne-Emma Lefèvre[1,3]*, Théo Cheynel[1,2]*, Omar El Khalifi[1], Thomas Daniel[1], Baptiste Bellot-Gurlet[1]

[1]Kinetix
[2]LIX, École Polytechnique, CNRS, IP Paris
[3]Université Claude Bernard Lyon 1, INSA Lyon, CNRS, LIRIS, UMR5205, Villeurbanne, France

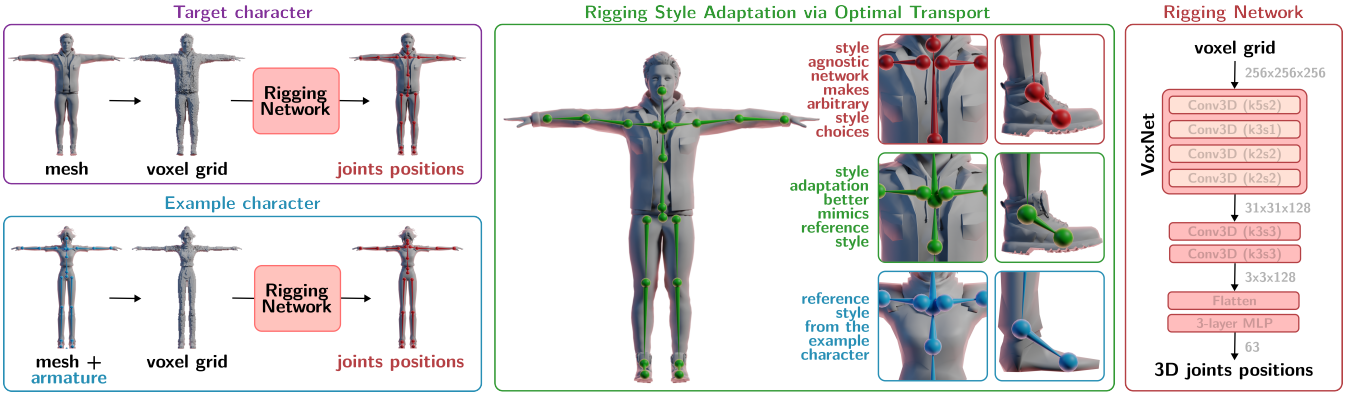

Figure 1: *Architecture of ARTIST: our rigging network automatically rig the target character. However, if the artist provides an example rigged character, our optimal transport method can copy its rigging style. The resulting rig better respects the artist's rigging conventions.*

Abstract

Automatic rigging transforms static meshes into articulated characters by predicting skeletal structure. However, rigging is inherently subjective: artists develop personal preferences for joint placement. Current approaches omit this aspect, learning only the average "style" of their training data. We quantify inter-artist variance through a user study and dataset analysis, demonstrating this notion of "rigging style". We propose a voxel-based model leveraging pretrained 3D backbones that outperforms state-of-the-art methods. We also introduce a one-shot style adaptation method based on volumetric optimal transport: given a single artist-rigged example, we transfer its stylistic joint placements to any new character. This improves any rigging model and supports different bone counts or hierarchies, reconciling automatic rigging with artistic variability.

Keywords: Rigging, Computer Graphics, Geometric Deep Learning, Optimal Transport, Volumetric Shape Matching

1. Introduction

Recent rigging methods estimate skeleton structures directly from meshes [BJD*12, XZKS19, BP07], yet none provide artists with control over joint placement — crucial for artistic and industrial use. Whether targeting arbitrary characters [XZK*20, LXY*25] or humanoid characters constrained to a strict template [GXM*25, CXL*25, MZ23], these models learn only the average rigging style of their training data and cannot adapt to individual preferences. Due to the lack of varied data, our approach leverages the robustness of pretrained 3D backbones for the rigging task. After comparing several backbones for 3D shape processing, we train a voxel-based neural rigging network, which predicts the position of the joints of the character from a single mesh in T-pose or A-pose, outperforming state-of-the-art rigging models.

We carry out a user study with professional riggers annotating the same characters which reveals significant inter-artist variability, confirming that rigging is subjective and that "rigging style" exists. This motivates us to formulate a mechanism that allows for stylistic adaptation in auto-rigging tools: given a single example of a character rigged by an artist (mesh + skeleton), we transfer the stylistic choices in bone positions onto any new character mesh.

*Equal contribution

DOI: 10.2312/egs.20261007

Our style adaptation relies on volumetric shape correspondence. Most existing shape matching methods rely on optimal transport on surfaces, relying on geometric information [SWC*22, SWS*15]. Neural Localizer Fields [SPM24] achieve a dense volumetric correspondence with neural networks, but require training data on deformable body templates. In contrast, our learning-free correspondence leverages the rigging network's joint predictions to build a volumetric segmentation, which allows us to establish correspondence between the two characters, and transfer the artist-defined joint positions from the example character to the target mesh. ARTIST thus combines the robustness of voxel-based backbones with controllable outputs for production-ready auto-rigging.

2. Placement of 3D joints

2.1. Choosing a 3D representation

Previous academic works use various 3D representations for the rigging task: voxels [KSBK21, XZKS19], mesh vertices [BJD*12], point clouds [XZK*20, CXL*25, ZPG*25], or gaussian splats [GXM*25]. However, given the limited quantities of data available for this task, we argue that fine-tuning pretrained generalist neural backbones allows for increased robustness on out-of-distribution characters. We compare three modalities and their respective backbones (point clouds, voxels, and multi-view images). Using characters from HumanRig [CXL*25], RigNet [XZK*20] and Mixamo [Ado24], we create a multimodal dataset where each mesh is converted into the three formats. Then, each format is encoded with its corresponding backbone, frozen during training: PointNet [QSMG17] for point clouds, VoxNet [MS15] for voxels, and ResNet-50 [HZRS16] for multi-view images. These embeddings are linearly projected to a 1024-dimensional latent vector, and for each modality, we train a 3-layer perceptron head to predict the 21 3D joints coordinates. All heads are trained under comparable conditions and evaluated using Chamfer distance–based metrics [XZKS19].

2.2. Joint regression model

As shown in the first lines of Table 1, voxel-based backbones consistently outperform alternatives. As such, we develop a rigging model based on voxels, built upon the VoxNet backbone. For improved accuracy, we use an occupancy grid of resolution 256, which is fed through the pretrained VoxNet despite the shape mismatch. We add two convolutional layers, after which the embeddings are processed by a MLP head predicting 21 joints positions. The whole network (head + backbone) is trained end-to-end using the mean-squared error on joint positions (Figure 1, right).

Method	MAE ↓	MSE ↓	CD-J2J ↓	CD-J2B ↓	CD-B2B ↓
ResNet	0.0444	0.0013	0.0357	0.0262	0.0258
PointNet	0.0532	0.0022	0.0418	0.0323	0.0315
VoxNet	0.0296	0.0007	0.0248	0.0166	0.0156
RigNet	0.0302	0.0025	0.0302	0.0249	0.0245
TARig	0.0277	0.0020	0.0277	0.0222	0.0221
Ours	**0.0166**	**0.0005**	**0.0216**	**0.0143**	**0.0123**
Artists	0.0197	0.0002	0.0170	0.0162	0.0080

Table 1: *Evaluation metrics on our test set (1382 characters) and on our user study (bottom row, 10 characters).*

Figure 2: *Some examples of our data augmentation (original characters on the left).*

2.3. Data augmentation procedure

To improve robustness across diverse body shapes and surface styles, we perform augmentation on character meshes through volumetric deformations. Our dataset combines 9000 AI-generated characters from HumanRig [CXL*25], 108 characters from Mixamo [Ado24] and a subset of 249 humanoids from RigNet-v1 [XZKS19]. Each character is deformed by a lattice deformation, with randomized scale factors applied to each anatomical regions (head, legs, torso, and arms), thereby generating plausible variations in body shape, height, and limb length. Figure 2 highlights the variety of these augmented characters.

2.4. Results

Our network performs better than both RigNet [XZK*20] and TARig [MZ23] on all commonly accepted metrics, as shown in the middle rows of Table 1. Moreover, thanks to the pretrained voxel backbone, our network demonstrates better robustness on out-of-distribution characters. Qualitative results on unseen characters are shown on Figure 4.

3. Example-based style adaptation

3.1. Necessity of style adaptation

We conducted a user study in which 10 professional rigging artists annotated the same set of 10 characters. Pairwise Chamfer distances between the annotations on the same character, provided in the last row of Table 1, reveal significant inter-artist differences. Figure 3 illustrates these differences in regions such as the collarbones, spines, and toes, due to individual habits or learned idiosyncrasies. Furthermore, we demonstrate that an artist's style is consistent across all characters. The offset between each artist's annotation and the mean joint placement across all riggers can be visualized using a t-SNE (Figure 5), which shows that the characters annotated by each artist are clustered together, with a Silhouette coefficient of 0.168.

Similar clusters appear in the RigNet-v1 dataset [XZKS19], when grouping characters by video game of origin. Each character having a single ground truth, compare it to the predictions of TARig [MZ23]. The presence of clusters (Silhouette coefficient: 0.101) on the t-SNE visualization on Figure 6 demonstrates that

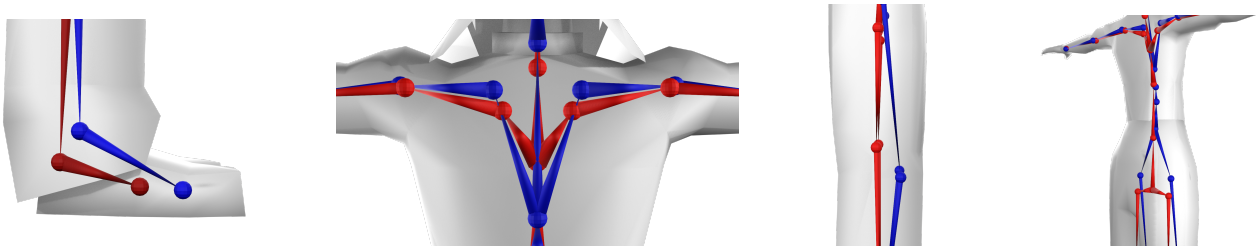

Figure 3: *Highlights of the stylistic differences in rigging between two specific artists (left to right: feet, clavicles, knees, spines).*

stylistic choices exist in the dataset, and that the network's performance could be substantially increased by using this information. To take into account the variability in rigging styles, we allow artists to provide an example rigged character of their choice, and adapt our method to reproduce the example's stylistic preferences onto any new target character.

3.2. Shape matching

We perform volumetric shape matching to transfer example bone positions p_b^E to target positions p_b^T. We first sample points inside the volume of the example and target characters, using our voxel grid. To increase surface coverage, we dilate the voxel grid and project the additional voxel centers back onto the mesh (shrinkwrap). This gives us two 3D distributions of points: $Q^E = \left\{q_i^E\right\}_{i\in[\![1,M_E]\!]}$ and $Q^T = \left\{q_i^T\right\}_{i\in[\![1,M_T]\!]}$ respectively. Our aim is to find an optimal *transport plan* $\mathbf{T} \in \mathbb{R}_+^{M_E\times M_T}$, that maps Q^E on Q^T:

$$\min_{\mathbf{T}} \sum_{i=1}^{M_E}\sum_{j=1}^{M_T} \mathbf{T}_{i,j}\, d(q_i^E, q_j^T) \quad \text{s.t.} \begin{cases} \forall j,\ \sum_{i=1}^{M_E} \mathbf{T}_{i,j} = w_j^T \\ \forall i,\ \sum_{j=1}^{M_T} \mathbf{T}_{i,j} = w_i^E \end{cases} \tag{1}$$

where $d(\cdot,\cdot)$ is an ℓ_2 norm, and where $w^E \in \mathbb{R}_+^{M_E}$, $w^T \in \mathbb{R}_+^{M_T}$ are predefined marginal probability density for our samples. For computational efficiency, we actually solve the entropy-regularized version of this problem [Cut13], by adding $\varepsilon \mathrm{KL}\left(\mathbf{T},\ \left(w^E\right)^\top w^T\right)$ to the cost function, where KL is the Kullback-Leibler divergence and ϵ is a regularization factor. This allows for a faster, GPU-compatible computation using the Sinkhorn-Knopp matrix scaling algorithm [FSV*19].

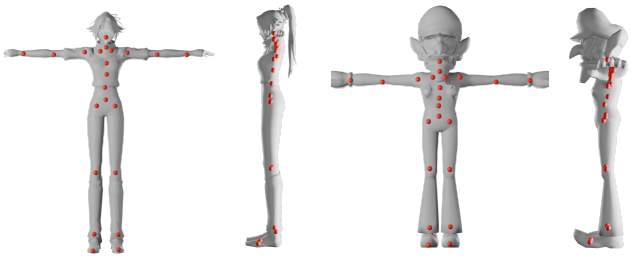

Figure 4: *Prediction of our rigging network on two unseen characters from the test set.*

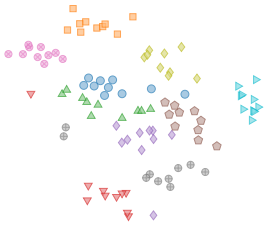

Figure 5: *t-SNE visualization of the difference of each annotation with respect to the mean annotation across all artists. One color per artist.*

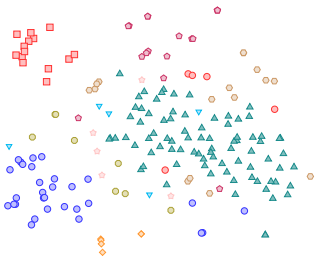

Figure 6: *t-SNE visualization of the difference of TARig's estimation with respect to the ground truth. One color per video game of origin.*

3.3. Marginal distributions

The choice of the marginal distributions w^E and w^T plays an important role in the final solution. Uniform marginals would cause misalignment when body proportions differ (e.g., mapping head mass to neck). Instead, we ensure each limb contains equal total mass in both distributions. To this end, we use the positions of the bones in each character estimated by the rigging network described in Section 2, $\left\{\hat{p}_b^E\right\}_{b\in[\![1,N]\!]}$ and $\left\{\hat{p}_b^T\right\}_{b\in[\![1,N]\!]}$. This works as follows:

- We first compute skinning weights $s_{i,b}^E$ (resp. $s_{j,b}^T$) that skin each point q_i^E (resp. q_i^T) to the bone b of the example (resp. target) characters, based on the positions $\hat{p}_b^E$ (resp. $\hat{p}_b^T$). For this, we use a voxel-based heat diffusion method [DdL13].
- We then compute the values of w_i in order to obtain a similar "quantity of mass" skinned to each bone. For that, we solve the following optimization problem:

$$\min_{w} \sum_{i=1}^{M} w_i^2 \quad \text{s.t.} \quad \forall b \in [\![1,N]\!], \quad \sum_{i=1}^{M} s_{i,b}\, w_i = 1 \tag{2}$$

Our problem being very sparse (points are rarely skinned to more than 3 bones) and involving large amounts of points, we devise a procedure (Algorithm 1) which runs faster than a convex optimization solver.

3.4. Transfer of bone positions

The transportation plan maps the example character's dense volumetric distribution to that of the target character. We use it to transfer the example character's bone position p_b^E into an estimation $\bar{p}_b^T$

Algorithm 1: Marginal weight computation

Input: Skinning weights $\{s_{i,b}\}_{i\in[\![1,M]\!],\, b\in[\![1,N]\!]}$
Output: Marginal vector $w = \{w_i\}_{i\in[\![1,M]\!]}$
$w \leftarrow 1_M$
for $l \in [\![1,K]\!]$ **do**
  for $b \in [\![1,N]\!]$ **do** $w^b := \frac{w}{\sum_{i=1}^{M} s_{i,b}\, w_i}$
  for $i \in [\![1,M]\!]$ **do** $w_i \leftarrow \sum_{b=1}^{N} s_{i,b} w_i^b$

inside the target character. For each bone b, we first find the k points of Q^E that are the closest to p_b^E – let us denote them by $q_{i_1}^E, \dots, q_{i_k}^E$.

We then use inverse distance weighting to interpolate the transport plan at this position. For each neighbor $q_{i_j}^E$ ($j \in [\![1,k]\!]$), we compute a Shepard interpolation weight α_j based on inverse distances. The target position $\bar{p}_b^T$ is then computed as a weighted combination of the transport destinations:

$$\bar{p}_b^T = \sum_{j=1}^{k} \alpha_j \cdot \left(\sum_{m=1}^{M_T} \frac{\mathbf{T}_{i_j,m}}{w_{i_j}^E} \cdot q_m^T \right) \tag{3}$$

3.5. Results

We report in Table 2 the improvements caused by our optimal transport method on the RigNet-v1 dataset (average improvement over each identified cluster, over all example / target character pairs). We also report the improvements on the characters from our user study, averaged over all pairs of characters from the same artist. Figure 1 illustrates our style-adaptive method on a character from our user study. Interestingly, our method could be used regardless of the number of joints and their hierarchy in the artist-provided example character.

Method	MAE ↓	MSE ↓	CD-J2J ↓	CD-J2B ↓	CD-B2B ↓
TARig	0.0248	0.0003	0.0218	0.0144	0.0121
TARig+OT	**0.0220**	**0.0002**	**0.0206**	**0.0127**	**0.0108**
TARig	0.0234	0.0003	0.0219	0.0127	0.0099
TARig+OT	**0.0219**	**0.0002**	**0.0210**	**0.0105**	**0.0081**
Ours	0.0201	0.0004	0.0188	0.0126	0.0110
Ours+OT	**0.0182**	**0.0002**	**0.0171**	**0.0101**	**0.0085**

Table 2: *Improvement brought by our optimal transport (OT) style adaptation on the RigNet-v1 dataset (top) and the characters from our user study (bottom). Results of Ours on RigNet-v1 are purposely not reported, as it was part of the training set.*

4. Conclusion and future works

We show that VoxNet extracts rigging-related features more effectively than point cloud or image backbones, and our proposed voxel-based network outperforms prior mesh-based methods. We identify and address the problem of personal rigging style via optimal transport adaptation.Future work includes octree acceleration and conditioning rigging networks on style embeddings. It would also be worth investigating whether this approach could be adapted to non-humanoid morphologies, such as quadrupeds.

References

[Ado24] ADOBE: Mixamo. https://www.mixamo.com/, 2024. Accessed: 2026-01-16. 2

[BJD*12] BOROSÁN P., JIN M., DECARLO D., GINGOLD Y., NEALEN A.: Rigmesh: automatic rigging for part-based shape modeling and deformation. *ACM TOG 31*, 6 (2012), 1–9. 1, 2

[BP07] BARAN I., POPOVIĆ J.: Automatic rigging and animation of 3d characters. *ACM TOG 26*, 3 (July 2007), 72–es. doi:10.1145/1276377.1276467. 1

[Cut13] CUTURI M.: Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS 26* (2013). 3

[CXL*25] CHU Z., XIONG F., LIU M., ZHANG J., SHAO M., SUN Z., WANG D., XU M.: Humanrig: Learning automatic rigging for humanoid character in a large scale dataset. In *IEEE CVPR* (2025), pp. 304–313. 1, 2

[DdL13] DIONNE O., DE LASA M.: Geodesic voxel binding for production character meshes. In *ACM SIGGRAPH/Eurographics SCA* (2013), pp. 173–180. 3

[FSV*19] FEYDY J., SÉJOURNÉ T., VIALARD F.-X., AMARI S.-I., TROUVE A., PEYRÉ G.: Interpolating between optimal transport and mmd using sinkhorn divergences. In *International Conference on AI and Statistics* (2019), pp. 2681–2690. 3

[GXM*25] GUO Z., XIANG J., MA K., ZHOU W., LI H., ZHANG R.: Make-it-animatable: An efficient framework for authoring animation-ready 3d characters. In *IEEE CVPR* (2025), pp. 10783–10792. 1, 2

[HZRS16] HE K., ZHANG X., REN S., SUN J.: Deep residual learning for image recognition. In *IEEE CVPR* (2016), pp. 770–778. 2

[KSBK21] KIM J., SON H., BAE J., KIM Y. M.: Auto-rigging 3D Bipedal Characters in Arbitrary Poses. In *Eurographics 2021 - Short Papers* (2021), Theisel H., Wimmer M., (Eds.), The Eurographics Association. 2

[LXY*25] LIU I., XU Z., YIFAN W., TAN H., XU Z., WANG X., SU H., SHI Z.: Riganything: Template-free autoregressive rigging for diverse 3d assets. *ACM TOG 44*, 4 (July 2025). doi:10.1145/3731149. 1

[MS15] MATURANA D., SCHERER S.: Voxnet: A 3d convolutional neural network for real-time object recognition. In *IROS* (2015), pp. 922–928. 2

[MZ23] MA J., ZHANG D.: Tarig: Adaptive template-aware neural rigging for humanoid characters. *Computers & Graphics 114* (2023), 158–167. 1, 2

[QSMG17] QI C. R., SU H., MO K., GUIBAS L. J.: Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017. 2

[SPM24] SÁRÁNDI I., PONS-MOLL G.: Neural localizer fields for continuous 3d human pose and shape estimation. *NeurIPS 37* (2024), 140032–140065. 2

[SWC*22] SALEH M., WU S.-C., COSMO L., NAVAB N., BUSAM B., TOMBARI F.: Bending graphs: Hierarchical shape matching using gated optimal transport, 2022. 2

[SWS*15] SU Z., WANG Y., SHI R., ZENG W., SUN J., LUO F., GU X.: Optimal mass transport for shape matching and comparison. *IEEE TPAMI 37*, 11 (2015), 2246–2259. doi:10.1109/TPAMI.2015.2408346. 2

[XZK*20] XU Z., ZHOU Y., KALOGERAKIS E., LANDRETH C., SINGH K.: Rignet: neural rigging for articulated characters. *ACM TOG 39*, 4 (Aug. 2020). doi:10.1145/3386569.3392379. 1, 2

[XZKS19] XU Z., ZHOU Y., KALOGERAKIS E., SINGH K.: Predicting animation skeletons for 3d articulated models via volumetric nets. In *3DV* (2019), IEEE, pp. 298–307. 1, 2

[ZPG*25] ZHANG J.-P., PU C.-F., GUO M.-H., CAO Y.-P., HU S.-M.: One model to rig them all: Diverse skeleton rigging with unirig. *ACM Trans. Graph. 44*, 4 (July 2025). doi:10.1145/3730930. 2