As-Plausible-As-Possible: Plausibility-Aware Mesh Deformation Using 2D Diffusion Priors - Supplementary Material

Seungwoo Yoo[*1] Kunho Kim[*1] Vladimir G. Kim[2] Minhyuk Sung[1]
[1]KAIST [2]Adobe Research

In this supplementary material, we first describe implementation details of our main pipeline (Sec. S1) and details of APAP-BENCH construction (Sec. S2). We also provide the exact question given to user study participants, as well as an example questionnaire (Sec. S3). Furthermore, we summarize additional experimental results including additional qualitative results for 3D shape deformation (Sec. S4), more complex 3D shape deformations achieved by leveraging classical deformation techniques (Sec. S5), human evaluation on the plausibility of 3D deformations via user study (Sec. S6), and an ablation study for 2D mesh editing based on qualitative results (Sec. S7).

S1. Implementation Details

We provide additional implementation details of Alg. 1 in the main paper. We used a modified version of the differentiable Poisson solver from [1], denoted by g in Alg. 1, and `nvdiffrast` [6] when implementing the differentiable renderer $\mathcal{R}$ in our pipeline. We render 2D/3D meshes at a resolution of 512×512.

When editing 2D meshes, we optimize $\mathcal{L}_h$ for $M = 300$ iterations in the `FirstStage` and jointly optimize $\mathcal{L}_h$ and $\mathcal{L}_{\text{SDS}}$ for $N = 700$ iterations in the `SecondStage`. For experiments involving the optimization of 3D meshes with increased geometric complexity, we use $M = 300$ and $N = 1000$ for each stage, respectively. We use ADAM [5] with a learning rate $\gamma = 1 \times 10^{-3}$ throughout the optimization. We use the Classifier-Free Guidance (CFG) scale of 100.0 and randomly sample $t \in [0.02, 0.98]$ when evaluating $\mathcal{L}_{\text{SDS}}$ following DreamFusion [8].

We use a script from *diffusers* [3] to finetune Stable Diffusion [9] with LoRA [4]. We employ `stabilityai/stable-diffusion-2-1-base` as our base model and augment its cross-attention layers in the U-Net with rank decomposition matrices of rank 16. For the task of 2D mesh editing, we train the injected parameters for 60 iterations, utilizing a rendering of a mesh as a training image. In the 3D shape deformation, where renderings from 4 canonical viewpoints (front, back, left, and right) are available, we finetune the model for 200 iterations. In both cases, we use the learning rate $\gamma = 5 \times 10^{-4}$.

S2. Details of APAP-BENCH

Image Generation. For evaluation purposes, we build APAP-BENCH 2D by generating 2 images of real-world objects for each of the 20 categories using Stable Diffusion-XL [7] as noted in the main paper. We segment the foreground objects from the generated images and run Delaunay triangulation to populate a collection of 2D meshes. When generating the images, we use the following template prompt `"a photo of [category name] in a white background"` for all categories to facilitate foreground object segmentation. Tab. S1 summarizes the list of categories. Note that the list includes both human-made and organic objects that can be easily found in the daily environment to test the generalization capability of a deformation technique to various object types.

Handle and Anchor Assignment. We manually assign two handle and anchor pairs to each mesh to imitate user instructions. Specifically, we choose vertices on the shape boundaries instead of internal vertices to induce deformations that alter object silhouettes. For instance, users would try to drag the bottom of a backpack downward to enlarge the shape, instead of dragging an interior point which may flip triangles, distorting the appearance. As an anchor, we use the vertex closest to the center of mass of each mesh.

In experiments using APAP-BENCH 3D and APAP-BENCH 2D, we note that utilization of neighboring vertices of the given handles and anchors during deformation helps retain smooth geometry near the handle. Therefore, we additionally

Human-Made	Organic
backpack	flying bird
bike	side view of cat
chair	side view of dog
high-heeled shoes	runway model
purse	sitting bird
side view of car	standing cheetah
sneakers	standing dragon
table	standing raccoon
airplane	standing sheep
	standing white duck
	starfish

Table S1. **Object categories of 2D meshes in APAP-BENCH 2D.** APAP-BENCH 2D includes 2D triangle meshes depicting various objects, including both human-made and organic objects.

sample vertices near the handles and anchors that lie in the sphere of radius $r = 0.01$ and denote the extended sets of handles and anchors *region handles* and *region anchors*, respectively. We use region anchors and a single handle for 3D experiments and region anchors and region handles for 2D cases. Note that we use the same sets of handles and anchors when deforming shapes with our baselines for fair comparisons.

S3. Details of User Study for 2D Mesh Editing

In Sec. 4 of the main paper, we reported the preference statistics collected from 102 user study participants who passed the vigilance tests. We provide additional details of the user study in the following. We instructed participants to select the most anticipated outcome when the displayed source image is edited by the dragging operation visualized as an arrow with the question: `"A visual designer wants to modify the object by clicking on a red point and dragging it in the direction of the arrow. Please choose a result that best satisfies the designer's edit, while retaining the characteristics and plausibility of the object."`

Fig. S1 (left) shows an example of a questionnaire provided to the participants. For vigilance tests, we included an editing result from DragDiffusion [10] depicting an object irrelevant to the source image in each question. The participants were asked to answer the same question. We illustrate an example questionnaire of a vigilance test in Fig. S1 (right).

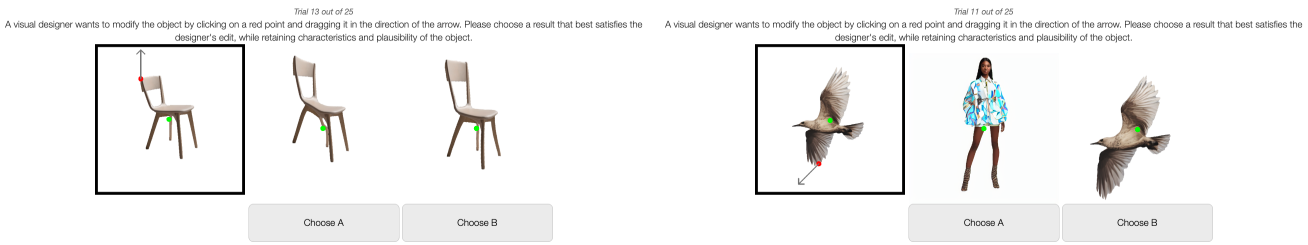


Figure S1. Examples of questionnaires displayed during the user study (2D mesh editing). During the user study, we asked the participants to evaluate 20 different result pairs from ARAP [11] and ours as shown on the left. To check whether a participant is focusing on the user study, we included 5 items for the vigilance test. As shown on the right, a question for the vigilance test includes an image of an object that is not related to the source image.

S4. Additional Qualitative Results for 3D Shape Deformation

Fig. S2 summarizes outputs of 3D shape deformation with additional results. As reported in the main paper, ARAP [11] only enforces local rigidity and hence cannot produce smooth deformations intended by users. In the ninth row, ARAP [11] introduces a pointy end given an editing instruction that drags the bottom of a doll downward. Ours, however, elongates the entire geometry smoothly, producing a more visually plausible deformation. Another example displayed in the tenth row shows similar behaviors of ARAP [11] and ours, respectively. Here, unlike ARAP [11], the proposed method adjusts the

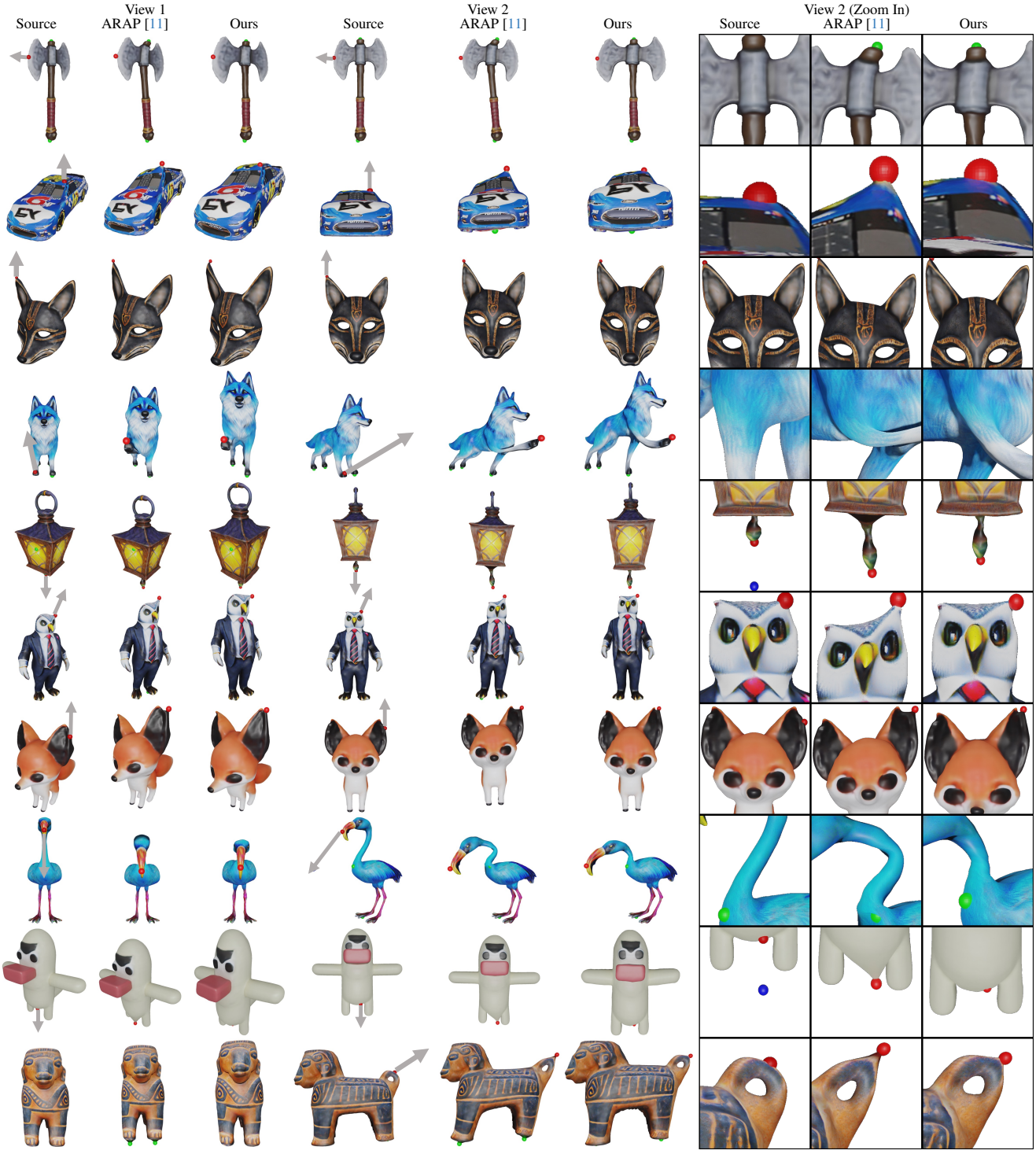


Figure S2. Additional qualitative results from 3D shape deformation. We visualize the source shapes and their deformations made using ARAP [11] and ours by following the instructions each of which specifies a handle (*red*), an edit direction denoted with an arrow (*gray*), and an anchor (*green*). We showcase the rendered images captured from two different viewpoints, as well as one zoom-in view highlighting local details.

overall proportion of the statue as the handle located at the tail is translated, while preserving the smooth and round geometry near the handle.

S5. Complex 3D Deformation Examples

In addition to the ability to optimize Jacobian fields using diffusion priors offered by the linearity of Poisson solvers, we can directly propagate local transforms, additionally defined at handle vertices, to Jacobians of neighboring faces by employing geodesic distances as weights [2]. This allows for more dramatic deformations illustrated in Fig. S3, involving limb articulations, large bending, and the use of multiple handles and anchors. As represented in the *Panda* (the seventh, eighth, ninth columns) example, our framework can handle large pose variations, useful in downstream applications, such as animation.

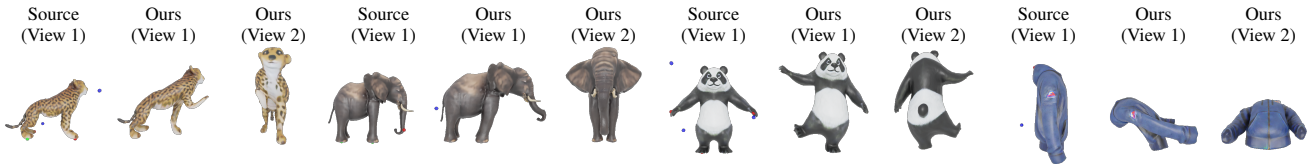


Figure S3. Examples of deforming source meshes using multiple handles and anchors. Best viewed in Zoom-in.

S6. User Study Results for 3D Shape Deformation

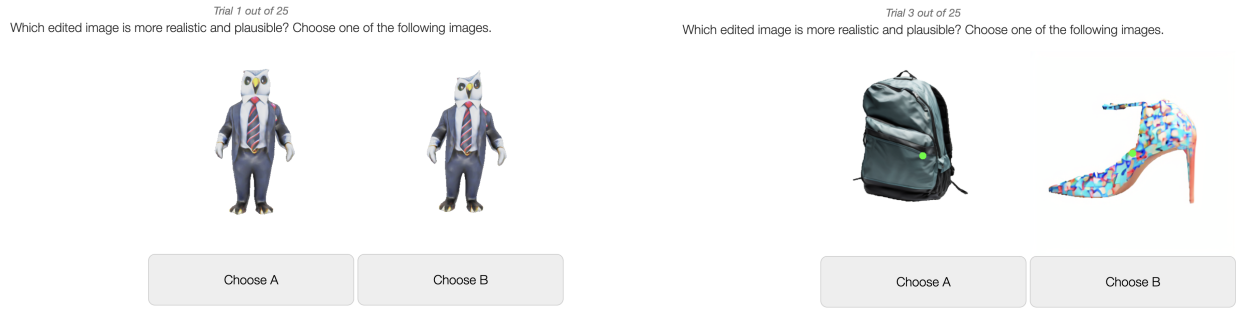


Figure S4. Examples of questionnaires displayed during the user study (3D shape deformation). During the user study, we asked the participants to evaluate 20 different result pairs from ARAP [11] and ours as shown on the left. To check whether a participant is focusing on the user study, we included 5 items for the vigilance test. As shown on the right, a vigilance test asks a participant to compare two images, with one of them containing noticeable artifacts.

Assessing the visual plausibility of 3D deformations is particularly challenging due to the difficulty in populating large-scale reference sets as we did for 2D meshes in Sec. 4 in the main paper. We further note that, unlike 3D generative models, computing image-based metrics such as CLIP-R score is non-trivial since it is hard to describe handle-based deformations solely using text prompts.

Therefore, we conduct a user study similar to the one presented in Sec. 4 in the main paper. We asked 47 user study participants on Amazon Mechanical Turk (MTurk) to compare rendered images of meshes deformed using ARAP [11] and ours. Each participant is provided with 20 image pairs and asked to select one image at each time given the question: `"Which edited image is more realistic and plausible? Choose one of the following images."` An example of a questionnaire displayed to the participants is shown in Fig. S4 (left). We provide an example of vigilance tests, similar to the user study for 2D mesh editing, in Fig. S4 (right). As summarized in Tab. S2, the deformation produced by our method is preferred over the results from the baseline.

Methods	Preference (%) ↑
ARAP [11]	41.7
Ours	**58.3**

Table S2. **User study preference for 3D mesh deformation.** In a user study targeting users on Amazon Mechanical Turk (MTurk), the results produced using ours were preferred over the outputs from the baseline.

S7. Ablation Study for 2D Mesh Editing

In this section, we provide qualitative results from the ablation study to validate the impact of each component on the plausibility of editing results. In Fig. S5, we summarize the results obtained by (1) optimizing only $\mathcal{L}_h$, (2) $\mathcal{L}_h$ and $\mathcal{L}_{\text{SDS}}$ without LoRA finetuning, (3) skipping the `FirstStage`, (4) using ARAP initialization, (5) using Poisson initialization, and (6) Ours. As mentioned in the main paper, optimizing only $\mathcal{L}_h$ (the second column) either distorts texture (the fifth row) or inflates or shrinks other parts of the given shape (the seventh and twelfth row). This demonstrates the necessity of a visual prior during deformation. Also, we observe the cases where skipping the `FirstStage` (the fourth column) does not lead to intended deformation as our diffusion prior is reluctant to modify shapes from their original states (the first, second, and fifth row). On the other hand, deformations initialized with the meshes produced by ARAP [11] (the fifth column) or Poisson solve (the sixth column) suffer from distortions that could not be resolved by optimizing $\mathcal{L}_{\text{SDS}}$ in the `SecondStage`.

Figure S5. Ablation study for 2D mesh editing. We examine the impact of each design choice on deformation outputs, including the use of diffusion prior (the second column), LoRA finetuning (the third column), two-stage pipeline (the fourth column), and initialization strategies during the `FirstStage` (the fifth and sixth column).

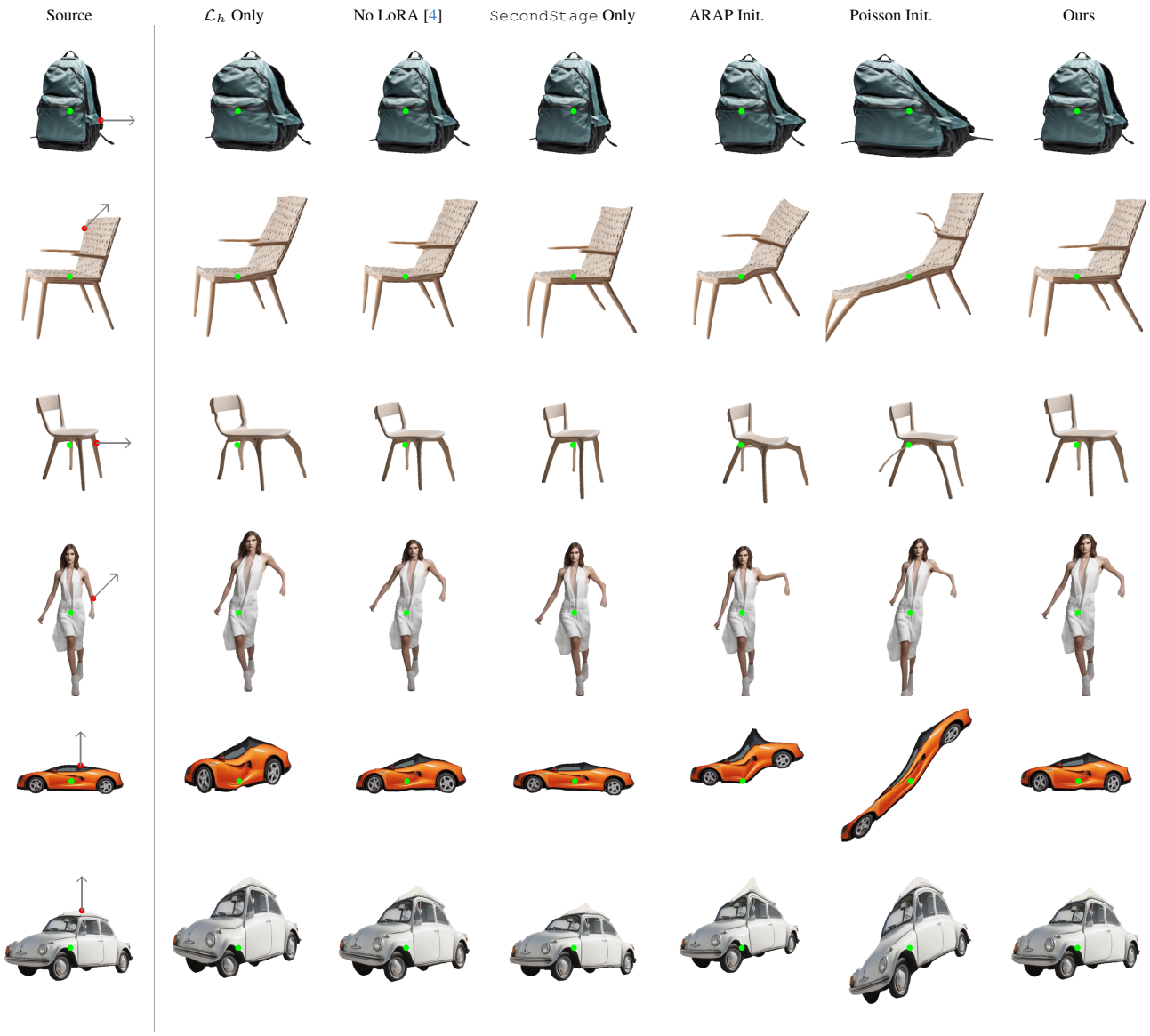
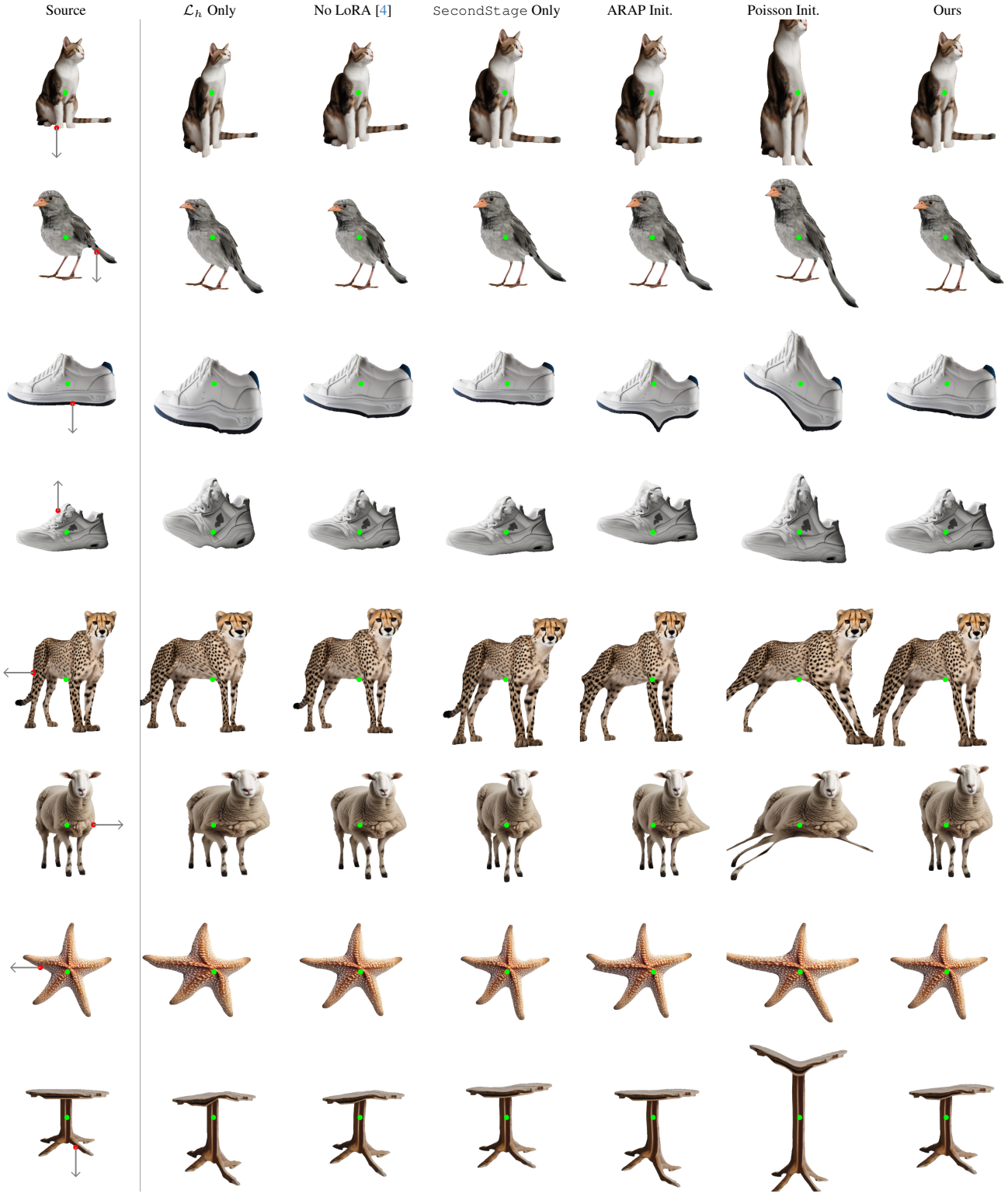


Figure S5. Ablation study for 2D mesh editing. We examine the impact of each design choice on deformation outputs, including the use of diffusion prior (the second column), LoRA finetuning (the third column), two-stage pipeline (the fourth column), and initialization strategies during the `FirstStage` (the fifth and sixth column).



References

[1] Noam Aigerman, Kunal Gupta, Vladimir G Kim, Siddhartha Chaudhuri, Jun Saito, and Thibault Groueix. Neural Jacobian Fields: Learning Intrinsic Mappings of Arbitrary Meshes. *ACM TOG*, 2022. 1

[2] Yu et al. Mesh editing with poisson-based gradient field manipulation. *ACM Trans. Graph.*, 23(3), 2004. 4

[3] Hugging Face. Diffusers: State-of-the-art diffusion models for image and audio generation in PyTorch. 1

[4] Yelong Hu, Edward J.and Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*, 2022. 1, 5, 6

[5] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1

[6] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular Primitives for High-Performance Differentiable Rendering. *ACM TOG*, 2020. 1

[7] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. *arXiv*, 2023. 1

[8] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. In *ICLR*, 2023. 1

[9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*, 2022. 1

[10] Yujun Shi, Chuhui Xue, Jiachun Pan, Wenqing Zhang, Vincent YF Tan, and Song Bai. DragDiffusion: Harnessing Diffusion Models for Interactive Point-based Image Editing. *arXiv*, 2023. 2

[11] Olga Sorkine and Marc Alexa. As-Rigid-As-Possible Surface Modeling. *Proceedings of EUROGRAPHICS/ACM SIGGRAPH Symposium on Geometry Processing*, pages 109–116, 2007. 2, 3, 4, 5