Alchemist: Parametric Control of Material Properties with Diffusion Models

Prafull Sharma[1,2] Varun Jampani[2] Yuanzhen Li[2] Xuhui Jia[2] Dmitry Lagun[2]

Fredo Durand[2] Bill Freeman[1,2] Mark Matthews[2]

In this supplementary document, we provide the description of the configurations used for baseline comparison and the description of the NeRF training on images edited using our method enabling material edited NeRF models. Please refer to `index.html` in the zip file for the video results of our method and video comparison to InstructPix2Pix prompt-only version trained on our data.

1. Baseline configurations

In the paper, we show comparison against GAN-based material editing [6], Null-text inversion with prompt-to-prompt [4], InstructPix2Pix [1], and InstructPix2Pix prompt-only version trained on our data. Following is the testing configuration for each of the methods.

GAN-based material editing [6]. The method takes image as input along with a scalar as input to perform relative material editing for glossiness and metallic. The method requires an input mask for localizing the object in the image. We generated the mask using the recommended website `www.remove.bg`. We input the image with the required transformation and tested it with scalar of 0.9. Note that the change in the material appearance is in the perceptual domain which is different than the shader based material change.

Null-text inversion with prompt-to-prompt [4]. This method is built on top of DDIM inversion, taking advantage of the unconditional textual embedding usually used for classifier-free guidance. For the given image, we first perform inversion which results in recovery of the input noisy latent and the unconditional textual embedding. Using this recovered latent which expresses the input image, prompt-to-prompt is employed to add an adjective in front of the object name to change the material attribute along with a scalar to define the weight for the adjective used in the cross-attention reweighting. Specifically, for albedo change, we add the word "gray" with a reweighting scalar of 5. For roughness we use the term "shiny" with 10 to make the object more shiny. To change the metallic component, we added the term "metal" in front of the object name with 8 as the scalar value. We add the term "transparent" with a scalar value as 10 to make the object transparent. This method requires a per-image optimization step, exploration of the prompt, and manual tuning of the scalar for cross-attention reweighting, while our method is feed-forward. We use the text conditioning weight as 7.5 as used in the original paper.

InstructPix2Pix [1]. We used pre-trained InstructPix2Pix based on Stable Diffusion 1.5 as presented by Brooks et al. [1]. We use the following instructions for each attribute as follows:

Roughness:"Make the `<object_name>` more/less shiny."

Metallic:"Make the `<object_name>` more/less metallic."

Albedo:"Make the color of the `<object_name>` gray."

Transparency:"Make the `<object_name>` transparent."

We use a text conditioning weight of 7.5 and image-conditioning weight of 1.5 for all experiments. Note that this model cannot be controlled using a scalar as the model is not trained using scalar inputs in the prompt.

InstructPix2Pix prompt-only trained on our data. We use prompts in the format of "Change the `<attribute>` of the `<object_name>` to `<scalar_strength>`" for `<attribute>` $\in$ {roughness, metallic, albedo, transparency}. Given that numerical values are hard to embed and use in the generation process results in non-smooth transitions as the values are changed.

2. Material editing in NeRFs

Our NeRF experiments use a vanilla NeRF configuration based on [5]. We use an 8 layer MLP of 256 channels, and separate density and color heads of 2 layers of 128 channels each. We use a single network sampled at 192 stratified positions and 32 importance sampled positions along each ray. We use 9 frequency bands of positional encoding, and 4 bands for view direction encoding. We train for $250k$ steps

with a learning rate of 10^{-4} using Adam optimizer [3] with $\beta_1 = 0.9, \beta_2 = 0.99$.

We hold out every 8^{th} view from DTU MVS dataset [2] as a "test" split, keeping the remaining views as the "train" split. Supplemental orbit animation viewpoints are outside of the test–train distribution, thus the large amount of floaters are not unexpected. We set near–far bounds to encompass only the foreground object, thus background artifacts are not unexpected either. Please refer to the attached html for the video results.

3. Qualitative comparison on synthetic data

We present qualitative comparison between our method and InstructPix2Pix finetuned with a prompt-only approach on held-out synthetic data. Please refer to the results for roughness (Figure 1), metallic (Figure 2), albedo (Figure 3), and transparency (Figure 4) attached below (on the next page).

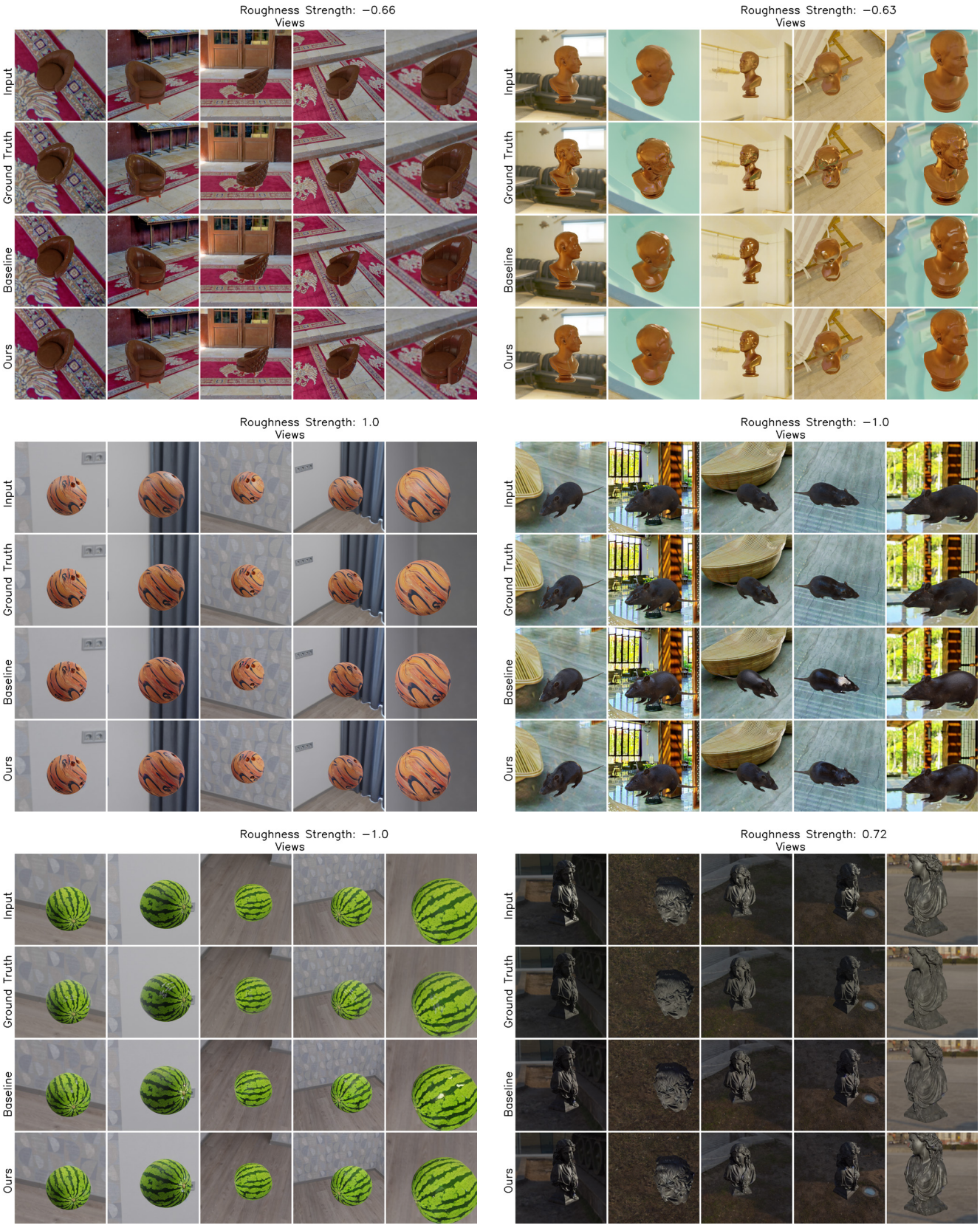

Figure 1. **Roughness test results.** Comparison of our method to InstructPix2Pix prompt-only method trained on our data (Baseline) for editing roughness on held-out synthetic data for different strengths.

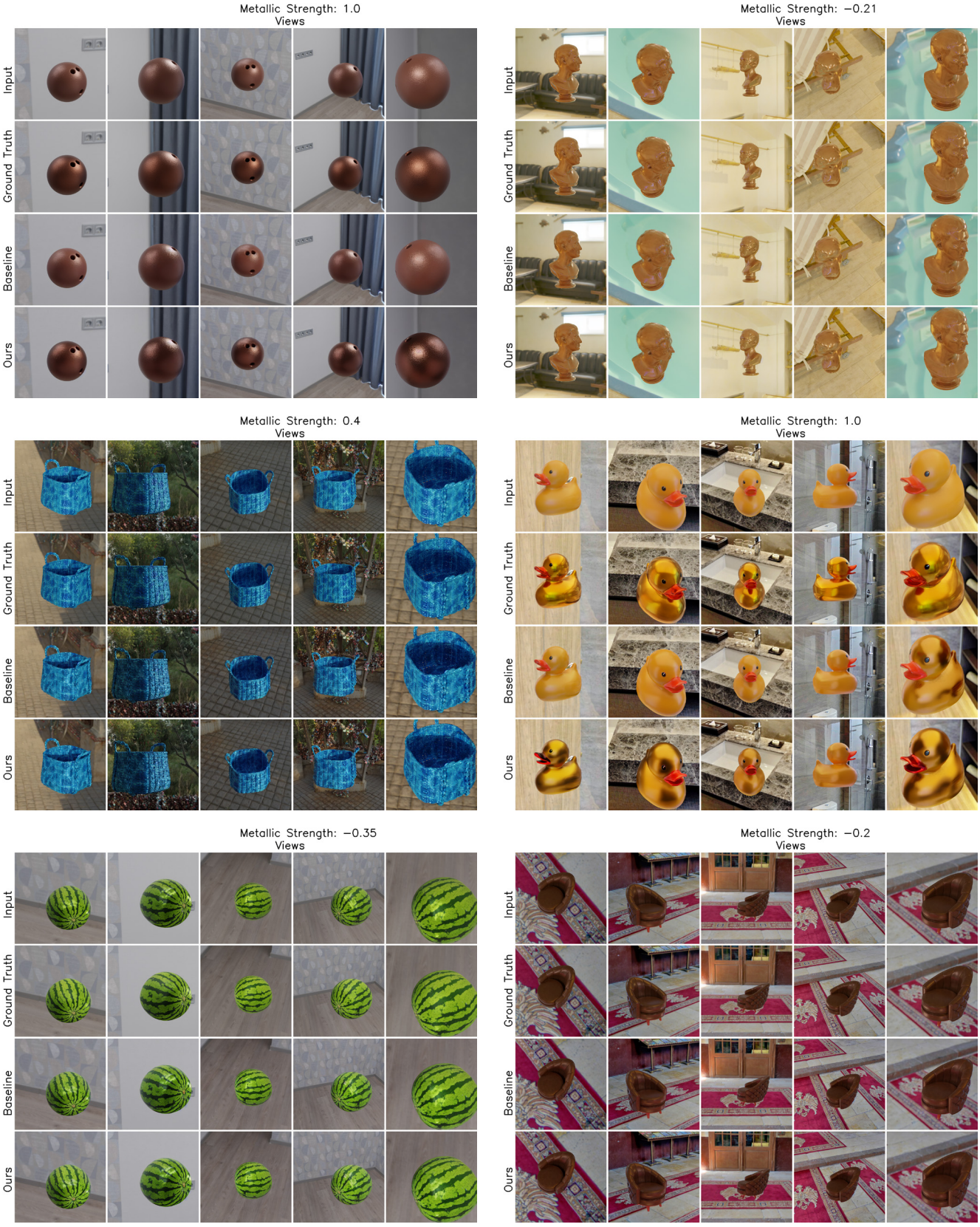

Figure 2. **Metallic test results.** Comparison of our method to InstructPix2Pix prompt-only method trained on our data (Baseline) for editing metallic attribute on held-out synthetic data for different strengths.

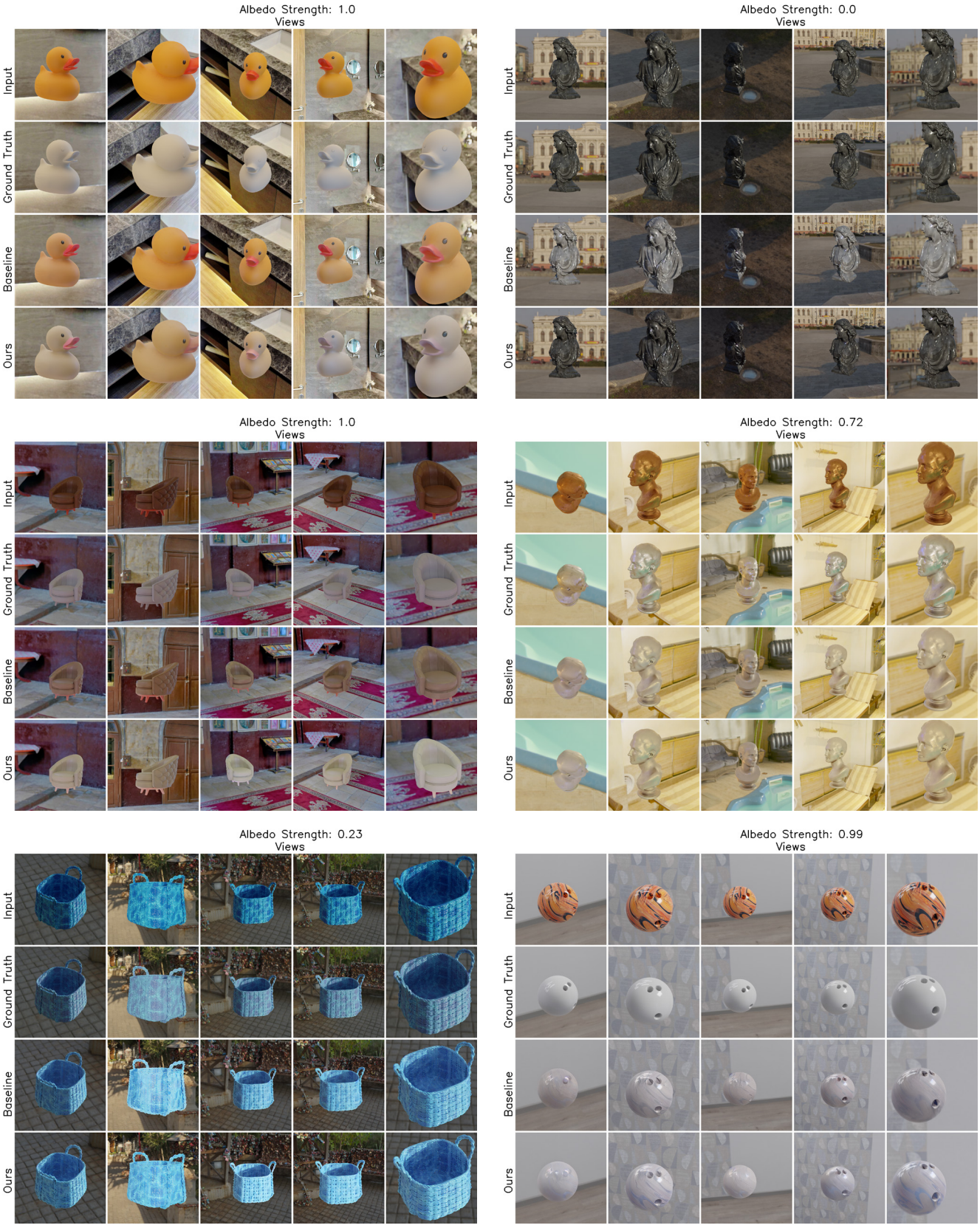

Figure 3. **Albedo test results.** Comparison of our method to InstructPix2Pix prompt-only method trained on our data (Baseline) for editing albedo on held-out synthetic data for different strengths.

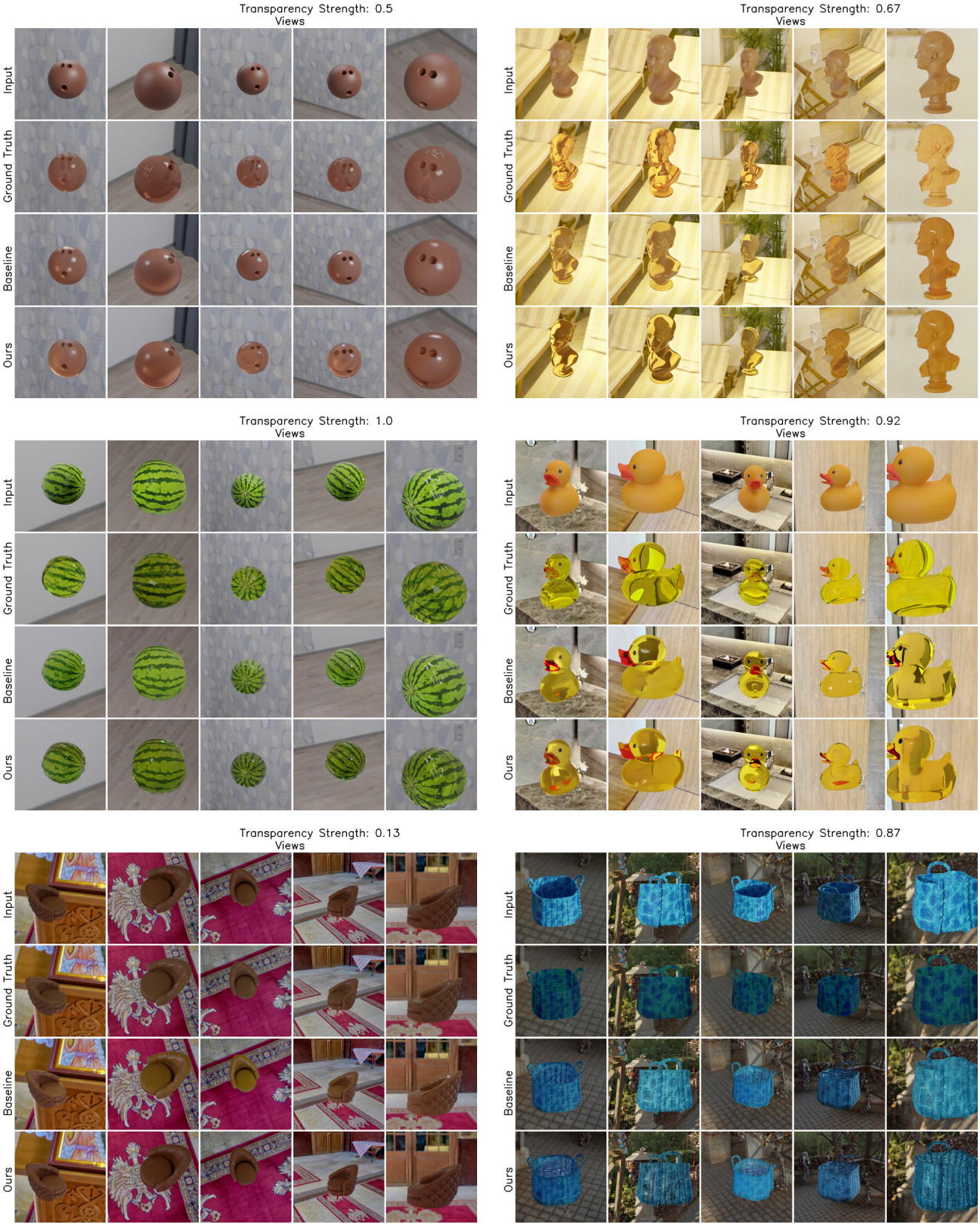

Figure 4. **Transparency test results.** Comparison of our method to InstructPix2Pix prompt-only method trained on our data (Baseline) for editing transparency on held-out synthetic data for different strengths.

References

[1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1

[2] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 2

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[4] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 1

[5] Daniel Rebain, Mark Matthews, Kwang Moo Yi, Dmitry Lagun, and Andrea Tagliasacchi. Lolnerf: Learn from one look. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1558–1567, 2022. 1

[6] J Daniel Subias and Manuel Lagunas. In-the-wild material appearance editing using perceptual attributes. In *Computer Graphics Forum*, volume 42, pages 333–345. Wiley Online Library, 2023. 1