Supplementary Material: MVDiffusion: Enabling Holistic Multi-view Image Generation with Correspondence-Aware Diffusion

Anonymous Author(s)
Affiliation
Address
email

The supplementary document provides 1) Error correction of the main paper 2) the full architecture specification of correspondence attention; 3) the impelementation details of MVDiffusion system; and 4) additional experimental results in the same format as the figures in the main paper.

1 ERRATA

We list several typos or unclarities in the main paper, and provide the corresponding corrections:

• **Equation (3)**: The correction equation should be:

$$\bar{\mathbf{F}}(\mathbf{s}) = \mathbf{F}(\mathbf{s}) + \boldsymbol{\gamma}(0), \quad \bar{\mathbf{F}}(t_*^l) = \mathbf{F}^l(t_*^l) + \boldsymbol{\gamma}(\mathbf{s}_*^l - \mathbf{s}) \tag{1}$$

2 Network Architecture of correspondence-aware attention block

The correspondence-aware attention block, depicted in Figure 1, comprises a transformer block and a ResNet block. The architecture of the transformer block is similar to vision transformers [3], with the inclusion of zero convolutions as suggested in ControlNet [16] and GELU [5] activation function. C, H, W are channel numbers, height and width respectively.

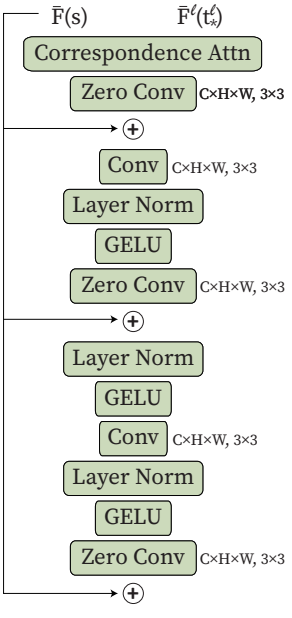

Figure 1: The architecture of the correspondence-aware attention block.

3 Implementation details of MVDiffusion

3.1 Homographic image generation

Data processing. Matterport3D dataset consists of 10,912 panorama images, each containing six skybox perspective images that can be converted into panoramic RGB visualizations. To ensure geometric consistency, we project each panorama into eight perspective images with a resolution of 1024 x 1024, a field of view (FoV) of 90 degrees, and a rotation angle of 45 degrees, resulting in eight images with known correspondences. In the first stage of training, the images are downsampled to a resolution of 256 x 256 to fit into the memory of a single GPU. We allocate 9,820 panoramas for training and reserve 1,092 panoramas for evaluation purposes.

Generation model. The generation model in our approach is built upon Stable-diffusion-v2 [13]. In the initial phase, we train the model on perspective images with a resolution of 256×256 for 20 epochs.

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

The training is performed using the AdamW optimizer with a batch size of 256 and a learning rate of $1e^{-5}$, utilizing four A6000 GPUs. In the second stage, we introduce the correspondence-aware attention block. Each image set consists of eight homographic images with a field of view (FOV) of 90 degrees and a rotation angle of 45 degrees. The correspondence-aware attention block is trained for 20 epochs with a batch size of eight and a learning rate of $1e^{-4}$ on four A6000 GPUs. During inference, we utilize the DDIM sampler with a step size of 50 to perform parallel denoising of the eight generated images. Additionally, we employ blip2 [7] to generate texts for each perspective image, and during both training and inference, we use the corresponding prompts.

Super resolution model. Our super-resolution model is derived from the publicly available Stable-diffusion-x4-upscaler[15] framework. In the first stage, we fine-tune the stable diffusion model on perspective images at a resolution of 1024×1024 for 20 epochs. This process uses the AdamW optimizer [8] with a learning rate of $1e^{-6}$ and a batch size of 64, utilizing four A6000 GPUs. In the second stage, we focus on multi-view homographic images. Each image set consists of eight homographic images, which are centrally cropped to a resolution of 512×512. We then train the correspondence-aware attention block for 20 epochs, using a batch size of four and a learning rate of $1e^{-4}$.

3.2 Implementation details of baselines

We introduce implementation details of baseline in the following.

- *Text2Light* [1] We combine the prompts of each perspective image and use the released pretrained model to generate the panorama.
- *Stable Diffusion (panorama)*[10] We fine-tuned Stable Diffusion using the panorama images within our training dataset, which contains 9820 panorama images at resolusion 512×1024. We fine-tuned the UNet layer of the Stable diffusion while keeping VAE layers frozen. We use AdamW optimizer with a learning rate of $1e^{-6}$ and batch size is 4, utilizing four A6000 GPUs.
- *Inpainting methods* [4, 6] In our approach, we employ Stable-diffusion-v2 [13] to generate the first image in the sequence based on the corresponding text prompt. For each subsequent image in the sequence, we utilize image warping to align the previous image with the current image. The warped image is then passed through Stable-diffusion-inpaint [14] to fill in the missing regions and generate the final image.
- *Stable diffusion (perspective)* In our approach, we utilize the model trained in the first stage of the generation module to generate the perspective images. During testing, each perspective image is associated with its own text prompt.

3.3 Multi-view depth to image generation

Data processing. ScanNet is an RGB-D video dataset containing over 1,500 indoor scenes with known camera parameters. We selected 200 scenes as training data. To construct our training sequence, we first select keyframes and ensure that each consecutive keyframe pair have an overlap of approximately 85%. Ultimately, we obtained 29,136 training keyframes. We use 26,222 randomly selected keyframes for training, while the rest serves for evaluation. Each training sample contains 6 sequential keyframes. The test set contains 486 non-overlapping image samples.

Generation model. Our generation model is derived from the stable-diffusion-2-depth framework [12]. In the initial phase, we train the model on a dataset of 290421 perspective images at a resolution of 192×256 for 50 epochs. This training process employs the AdamW optimizer [8] with a learning rate of $1e^{-5}$ and a batch size of 256, utilizing four A6000 GPUs. In the second stage, we introduce the correspondence-aware attention block. We preprocess the perspective images, yielding 29136 training sets of multi-view images, each comprising six perspective images. The correspondence-aware attention block is subsequently trained for 20 epochs, with a batch size of eight and a learning rate of $1e^{-4}$, using the same four A6000 GPUs. During the inference stage, we deploy the DDIM [11] sampler with a step size of 50 to perform parallel denoising on eight images.

Super resolution model. Our super-resolution model, same as for homographic image generation, is based on the publicly available Stable-diffusion-x4-upscaler framework [15]. Additionally, we enrich the model with depth information. The output is then summed with the latent features. The training process is also split into two stages. In the first stage, we fine-tune the stable diffusion model on perspective images at a resolution of 768×1024 for 20 epochs. This process employs the AdamW

optimizer [8] with a learning rate of $1e^{-6}$ and a batch size of 64, utilizing four A6000 GPUs. In the second stage, we work with multi-view homographic images. Each image set is composed of 6 multi-view images, which we central crop to a resolution of 384×512. The correspondence-aware attention block is then trained for an additional 20 epochs, with a batch size of four and a learning rate of $1e^{-4}$. During inference, we utilize the DDIM sampler with a step size of 50 for generating images.

3.4 Implementation details of baselines

We introduce the implementation details of baselines in the following.

• *RePaint*[9]: In our method, we utilize depth-conditioned Stable-diffusion-v2 [13] to generate the first image in the sequence. For each subsequent image, we condition it on the previous image by applying latent warping. This helps align the generated image with the previous one. To complete the remaining areas of the image, we employ the Repaint technique [9] for inpainting.

• *Depth-conditioned ControlNet*: We use the same method to generate the first image as the above method. Next, we warp the generated images to the current frame and use Stable-inpainting model [14] to fill the hole. To incorporate depth information into the inpainting model, we utilize a method from a public codebase [2], which adds the feature from depth-conditioned ControlNet [16] into each UNet layer of the inpainting model. For more detailed information, please refer to their code repository. In order to reduce the domain gap, the Stable-inpainting model has been fine-tuned on our training dataset. Similar to other fine-tuning procedures, we only fine-tuned UNet layers while keeping VAE part fixed. The fine-tuning was conducted on a machine with four A6000 GPUs. The batch size is 4 and the learning rate is $1e^{-6}$. We used AdamW as the optimizer. During inference, we utilize the DDIM sampler with a step size of 50 for generating images.

3.5 Visualization results

Figures 2-14 present supplementary results for panorama generation. In these figures, we showcase the output panorama images generated by both Stable diffusion (panorama) and Text2light methods. To compare the consistency between the left and right borders, we apply a rotation to the border regions, bringing them towards the center of the images. These additional visualizations provide further insights into the quality and alignment of the generated panorama images.

Figures 15-20 show additional results with two baseline methods (depth-conditioned ControlNet [16] and Repaint [9]).

Figure 21 shows additional results of interpolated frames. The keyframes are at the left and the right, the middle frames are generated by applying our Interpolation module (see Sec. 4.2 in the main paper). The consistency is maintained throughout the whole sequence.

References

[1] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. Text2light: Zero-shot text-driven hdr panorama generation. *ACM Transactions on Graphics (TOG)*, 41(6):1–16, 2022.

[2] Mikolaj Czerkawski. Controlnetinpaint. `https://github.com/mikonvergence/ControlNetInpaint`, 2023.

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[4] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation. *arXiv preprint arXiv:2302.01133*, 2023.

[5] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[6] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. *arXiv preprint arXiv:2303.11989*, 2023.

[7] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[9] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022.

[10] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[11] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[12] StabilityAI. Stable-diffusion-2-depth. `https://huggingface.co/stabilityai/stable-diffusion-2-depth`, 2023.

[13] StabilityAI. Stable-diffusion-2. `https://huggingface.co/stabilityai/stable-diffusion-2`, 2023.

[14] StabilityAI. Stable-diffusion-impaint. `https://huggingface.co/runwayml/stable-diffusion-inpainting`, 2023.

[15] StabilityAI. Stable-diffusion-upscalerx4. `https://huggingface.co/stabilityai/stable-diffusion-x4-upscaler`, 2023.

[16] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023.

A kitchen counter with a vase, marble and wooden countertops, hallway with wooden door and tiled floor, stainless steel refrigerator/oven, and a kitchen with stove, oven, and sink.

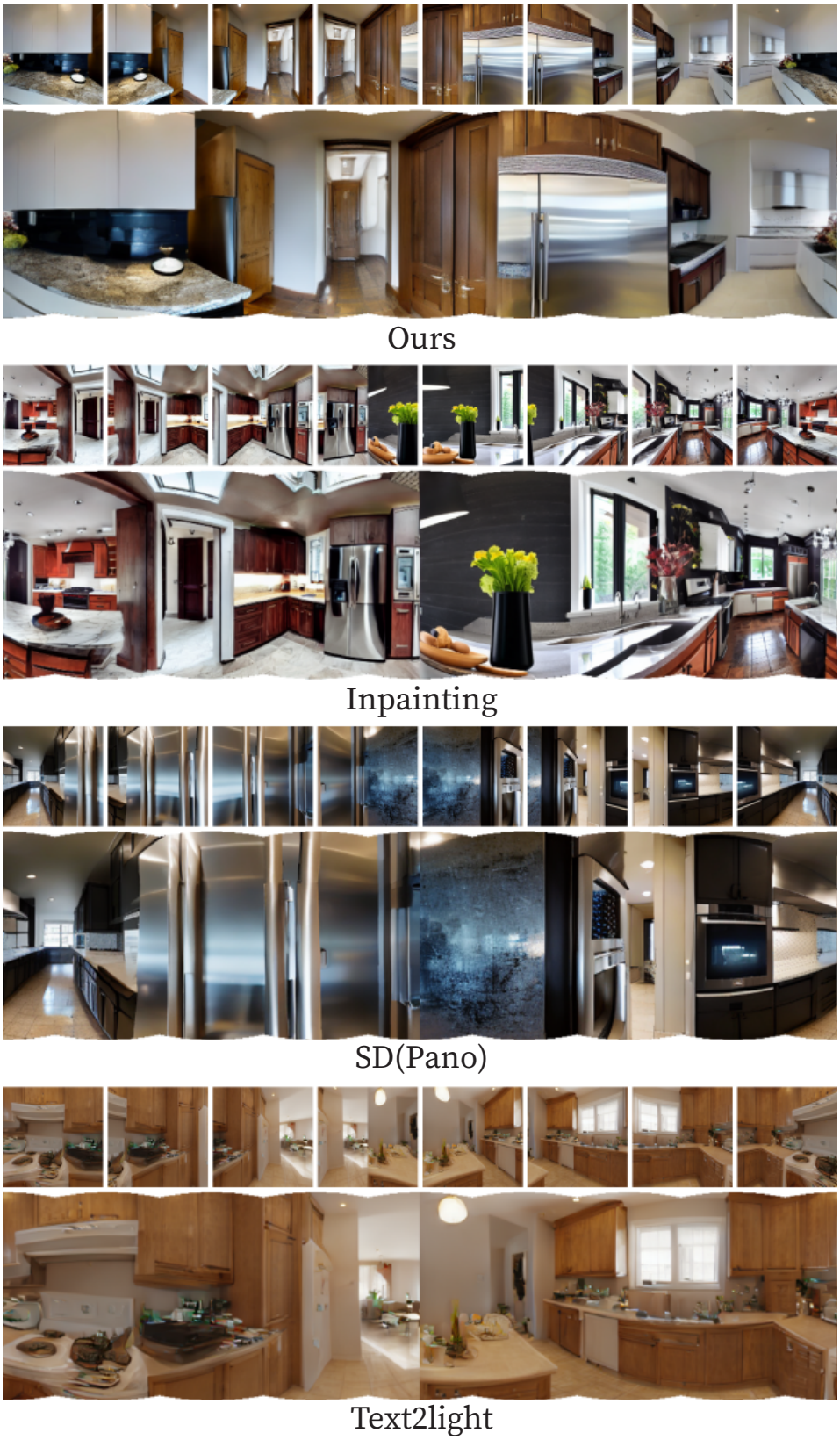

Figure 2: Addition results for panorama generation

A living room with a large glass table, white chairs and a giant window. A TV is attached to the wall and a fancy chandelier is hanging on the ceiling.

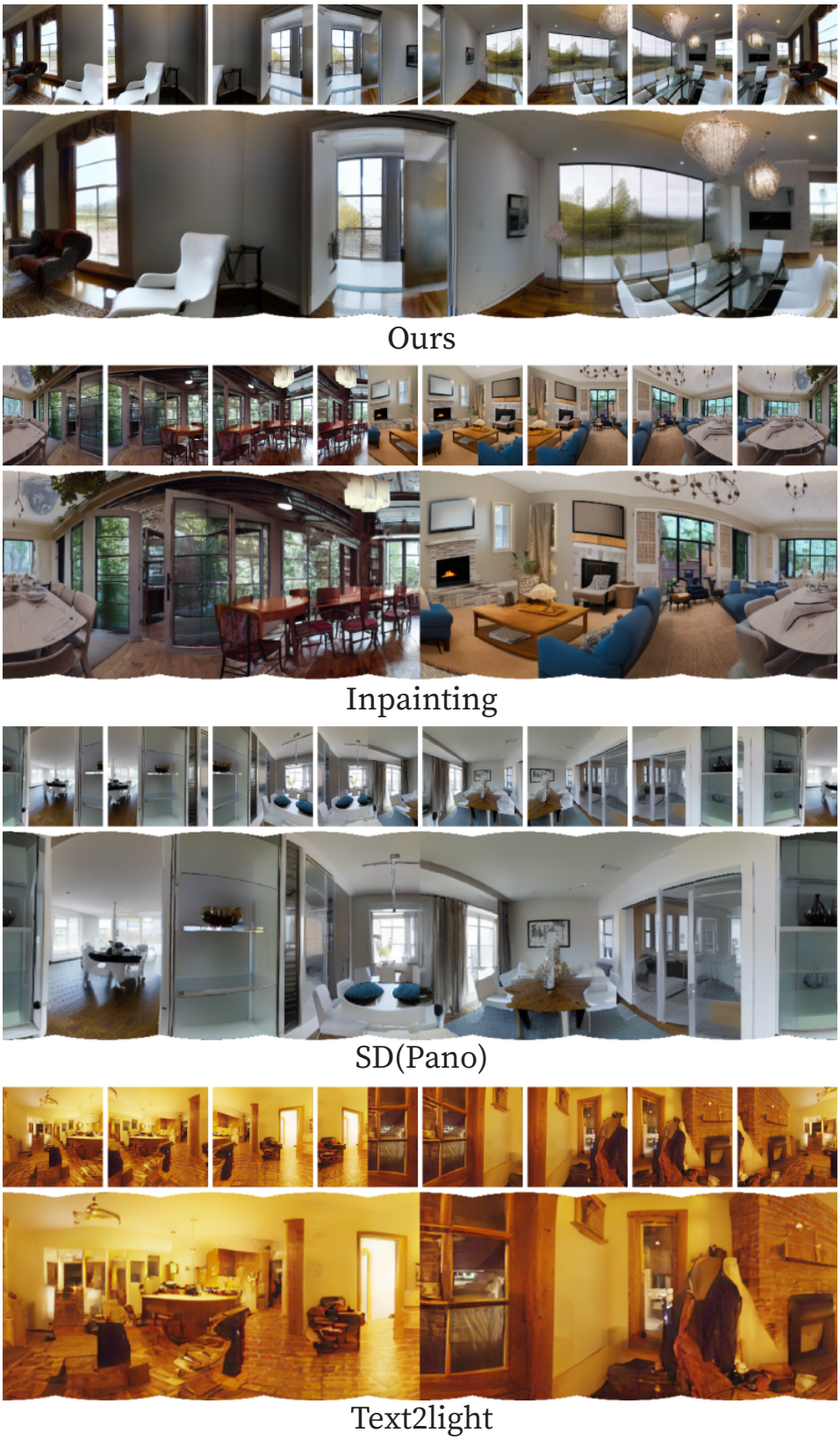

Figure 3: Addition results for panorama generation

A bedroom with a large bed and sliding glass doors. An open door to a patio with an ocean view. A room with a chair and a lamp.

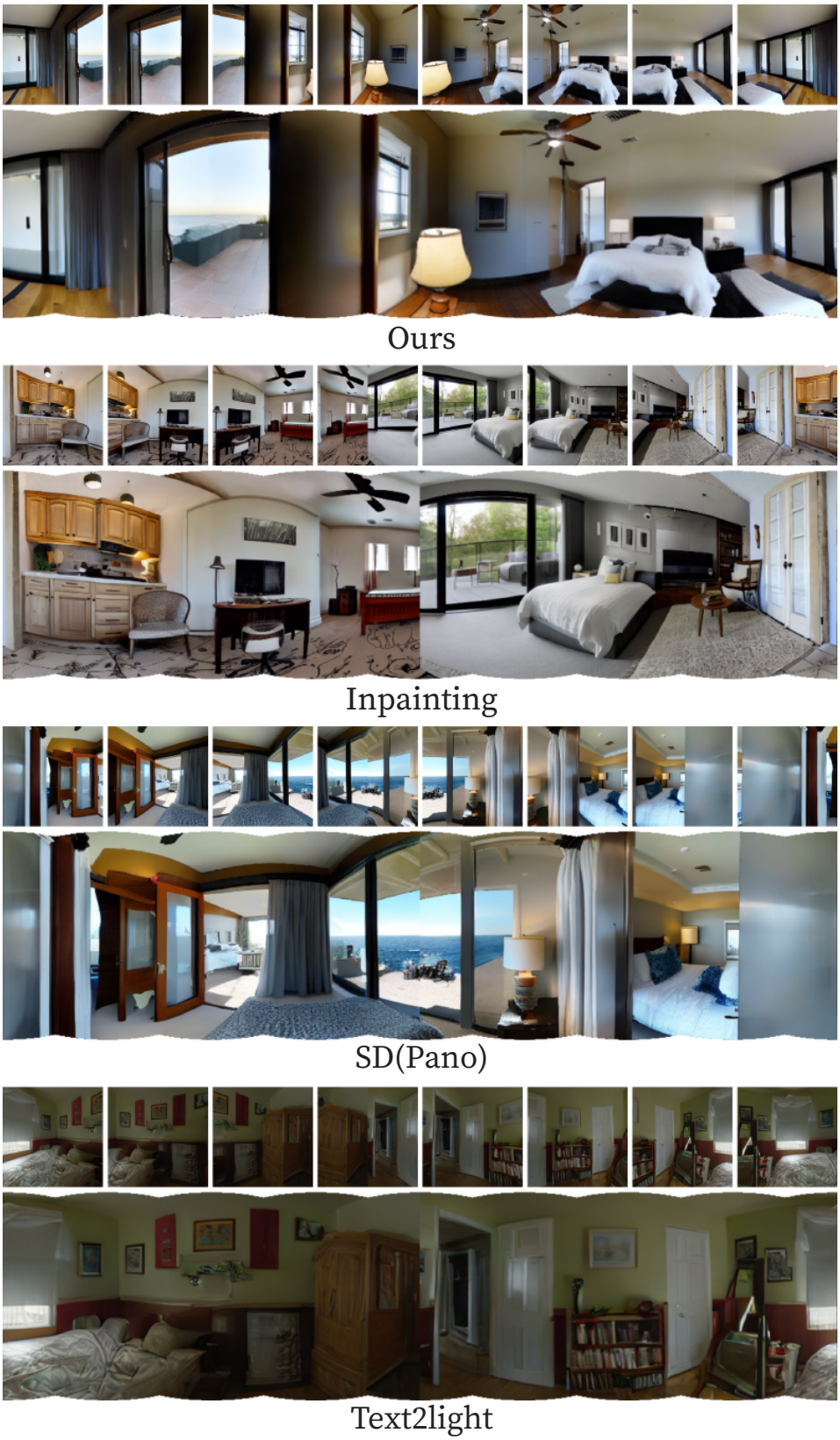

Figure 4: Addition results for panorama generation

A room with white cabinets and a wooden floor. A walk in closet with a lot of shelves. An empty room with a closet and shelves. A hallway with white walls and a white floor.

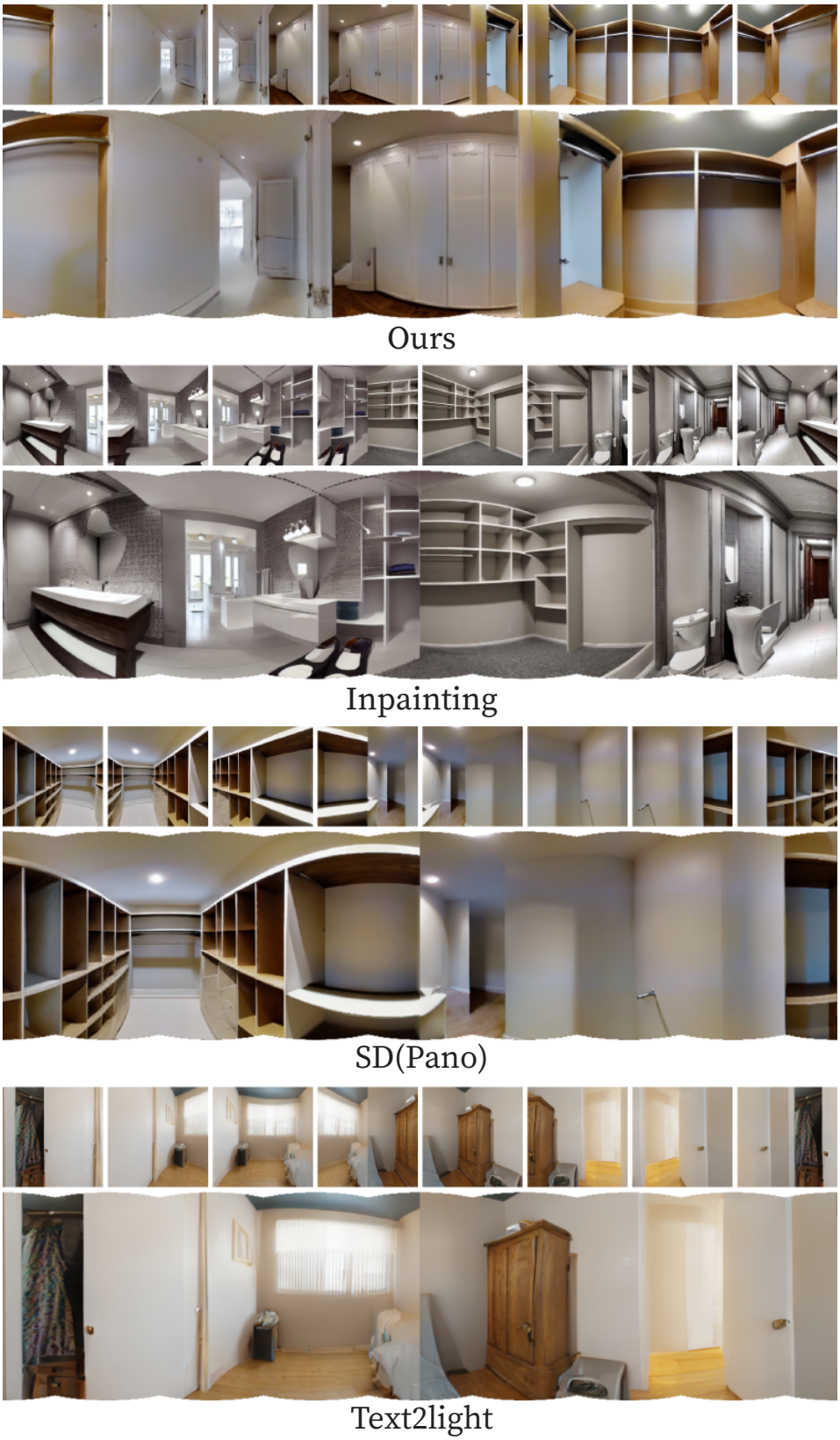

Figure 5: Addition results for panorama generation

A living room filled with furniture, a grand piano, a fire place, a painting, a large window. A living room with a couch and a ceiling fan.

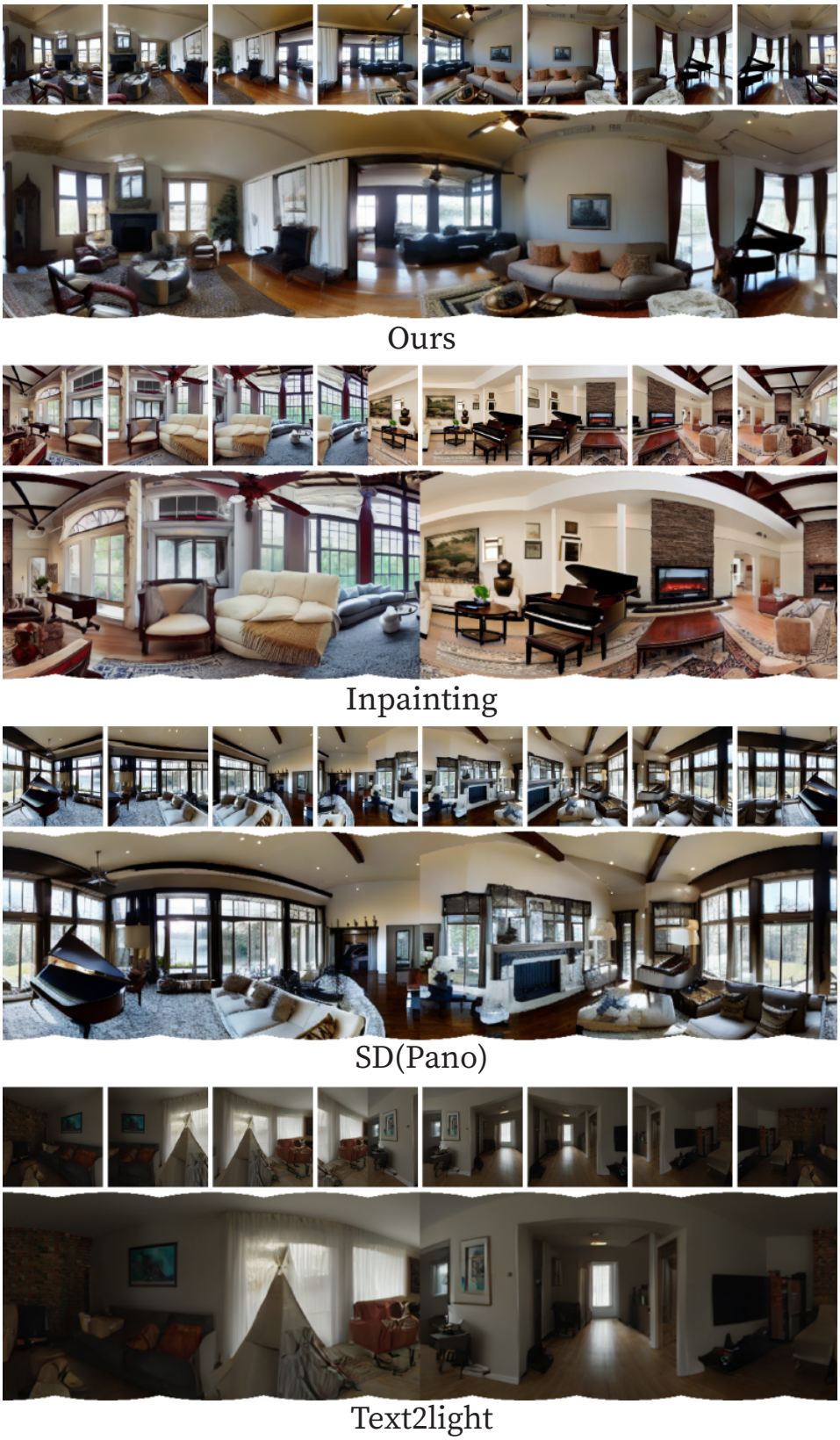

Figure 6: Addition results for panorama generation

A grand piano sitting in a living room next to a window. A living room filled with furniture and a fire place. A black piano sitting in a living room next to a window.

Ours

Inpainting

SD(Pano)

Text2light

Figure 7: Addition results for panorama generation

A dining room with a chandelier a table and chairs. A living room with white walls and wood floors. A room with a mirror and a mirror on the wall.

Ours

Inpainting

SD(Pano)

Text2light

Figure 8: Addition results for panorama generation

A bedroom with a bed and a mirror and a window. A vase of flowers on a shelf in a room. A hallway with two framed pictures on the wall.

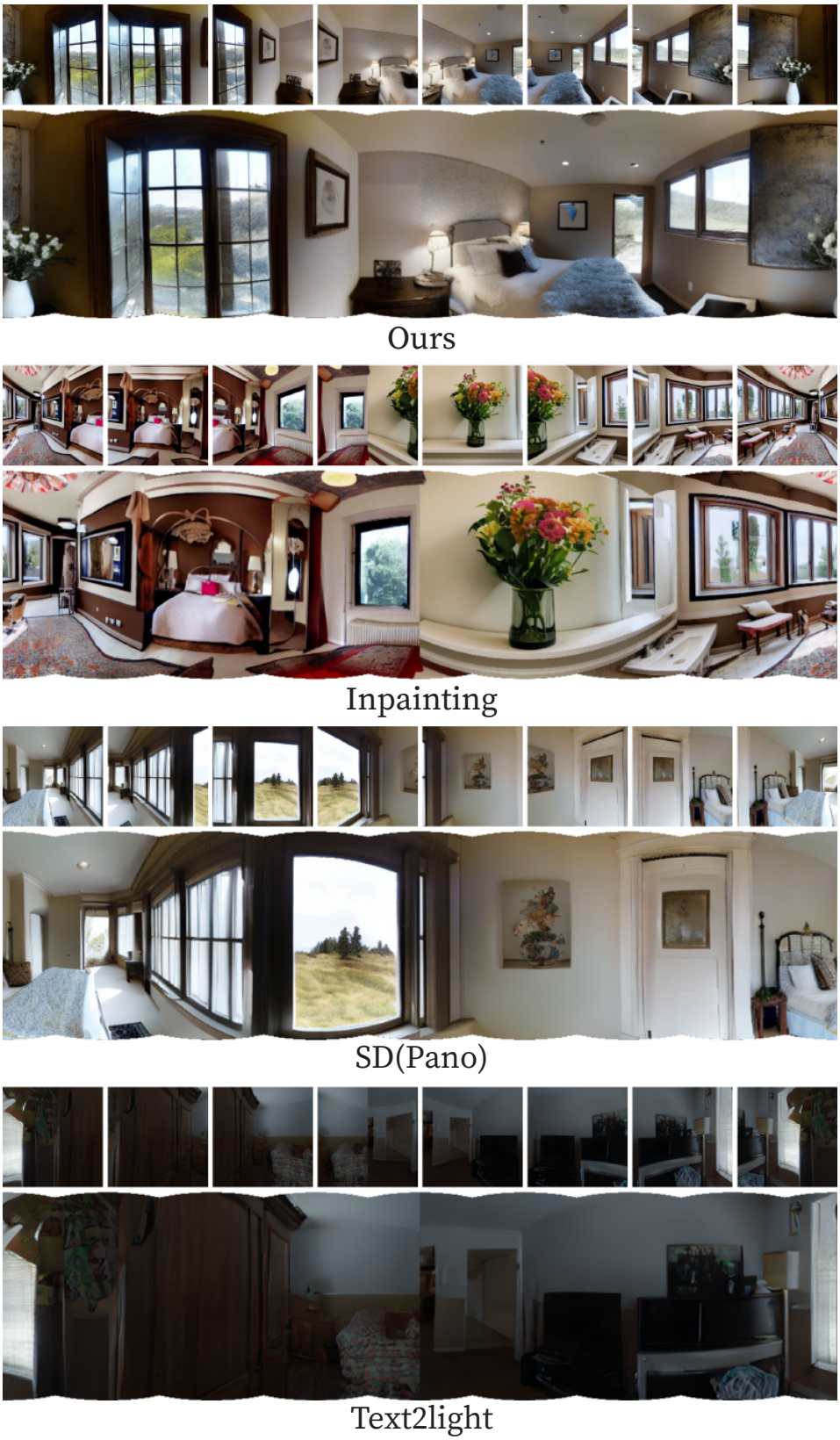

Figure 9: Addition results for panorama generation

A living room filled with furniture and a large mirror. A table with a lamp and a mirror on it. A living room filled with furniture and a chandelier.

Ours

Inpainting

SD(Pano)

Text2light

Figure 10: Addition results for panorama generation

A chandelier hanging from the ceiling of a house. A large room with a lot of windows. a staircase with a chandelier in a house. A room with a wooden floor and white walls.

Figure 11: Addition results for panorama generation

A large kitchen with a center island and white cabinets. A dining room with a table and chairs. A view of a pool through a glass door. A room with a lot of windows and a wooden floor.

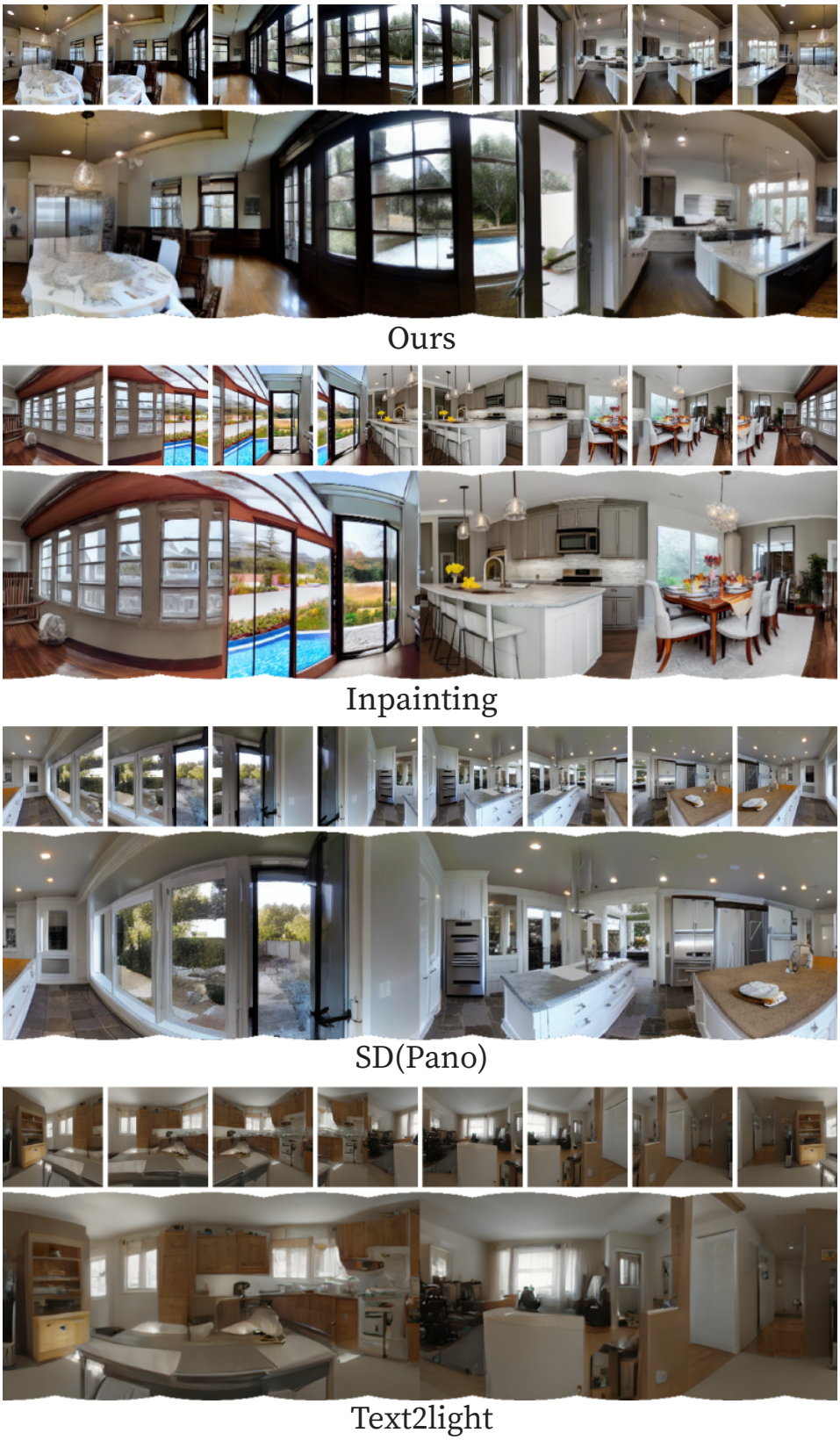

Figure 12: Addition results for panorama generation

A living room with hardwood floors and a ceiling fan. A living room filled with furniture and a chandelier. A living room with two white chairs and a painting on the wall.

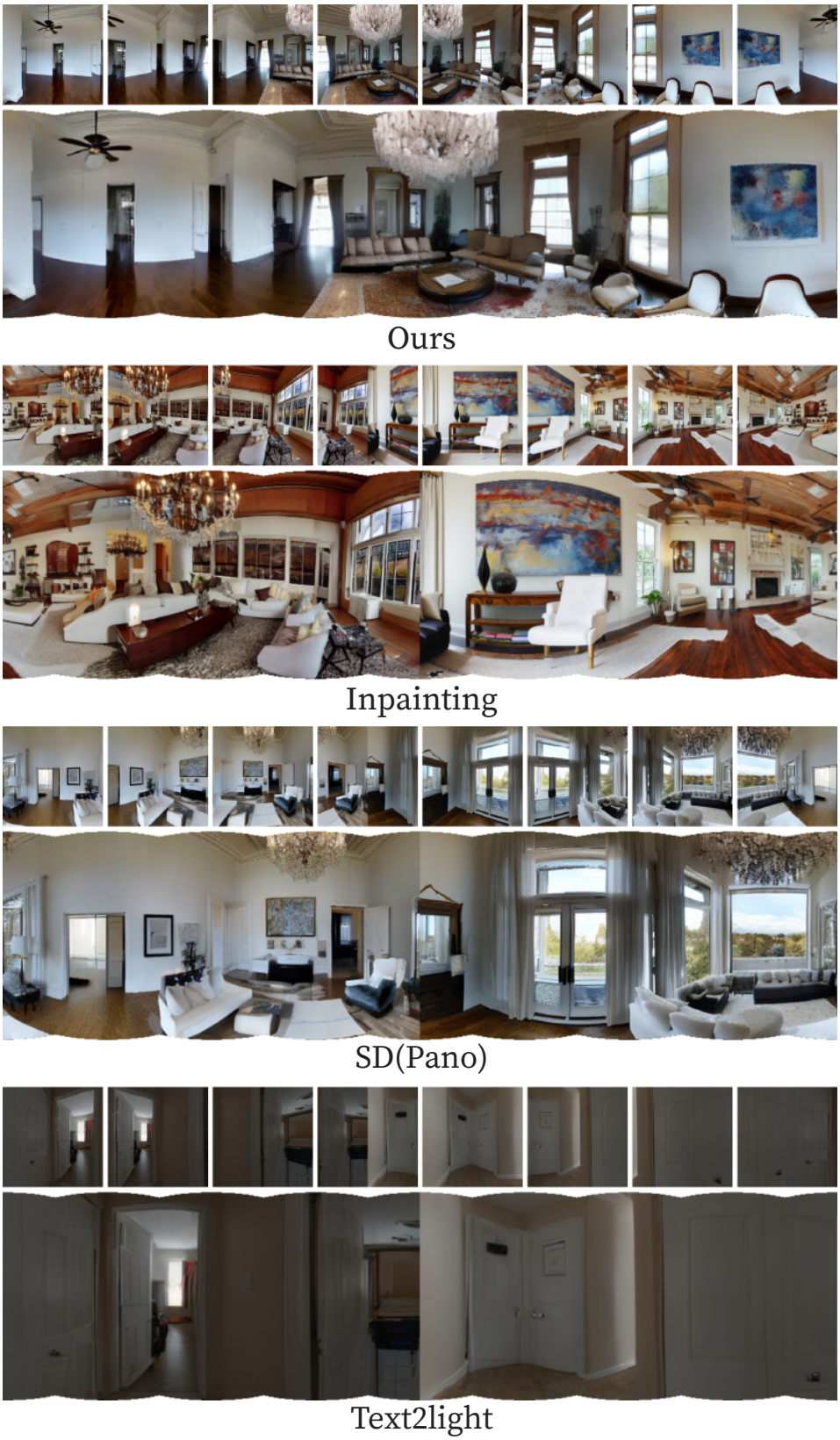

Figure 13: Addition results for panorama generation

A living room filled with furniture and a flat screen tv. A plant in a pot in a living room. A bedroom with a large bed and a piano.

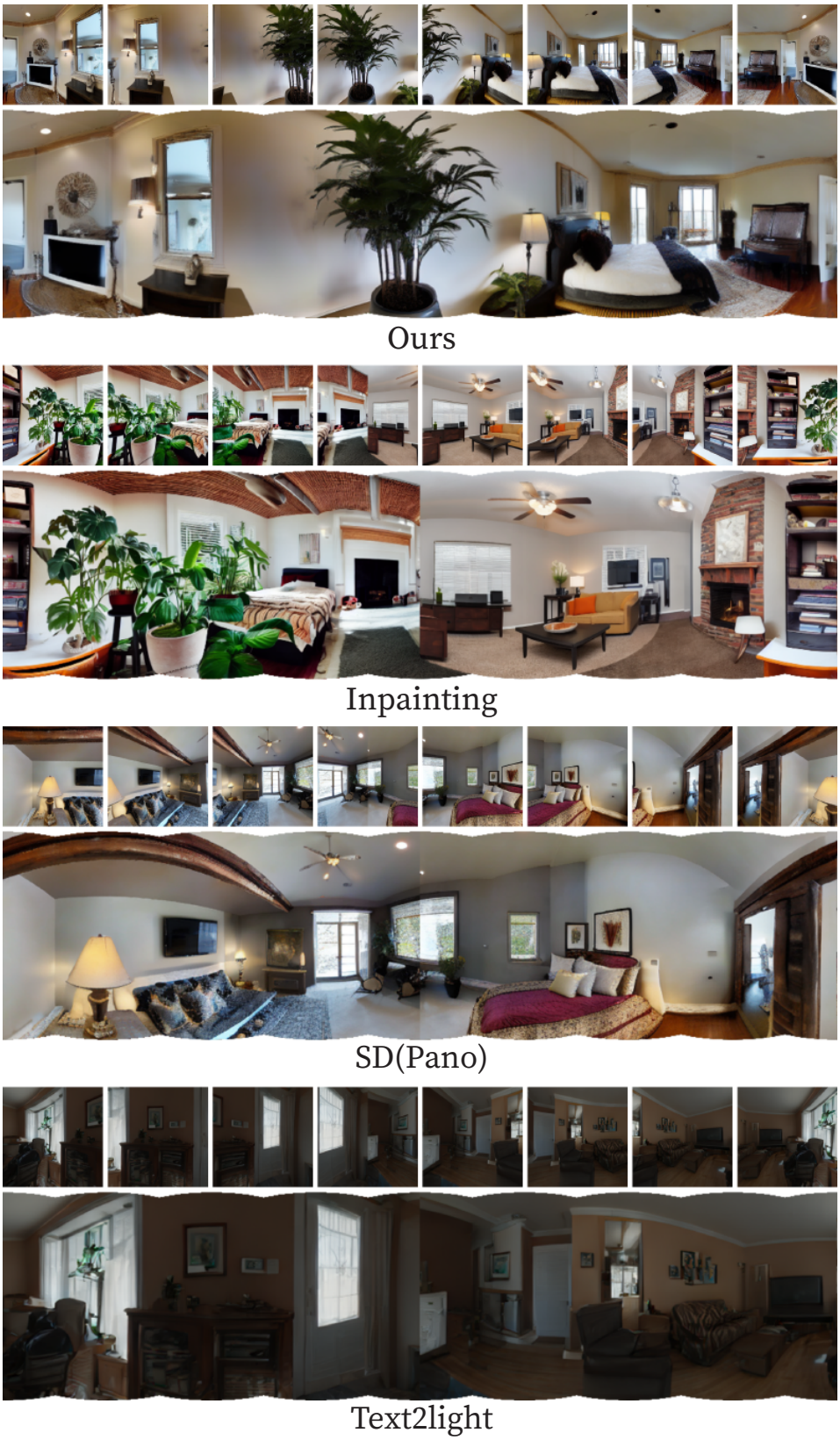

Figure 14: Addition results for panorama generation

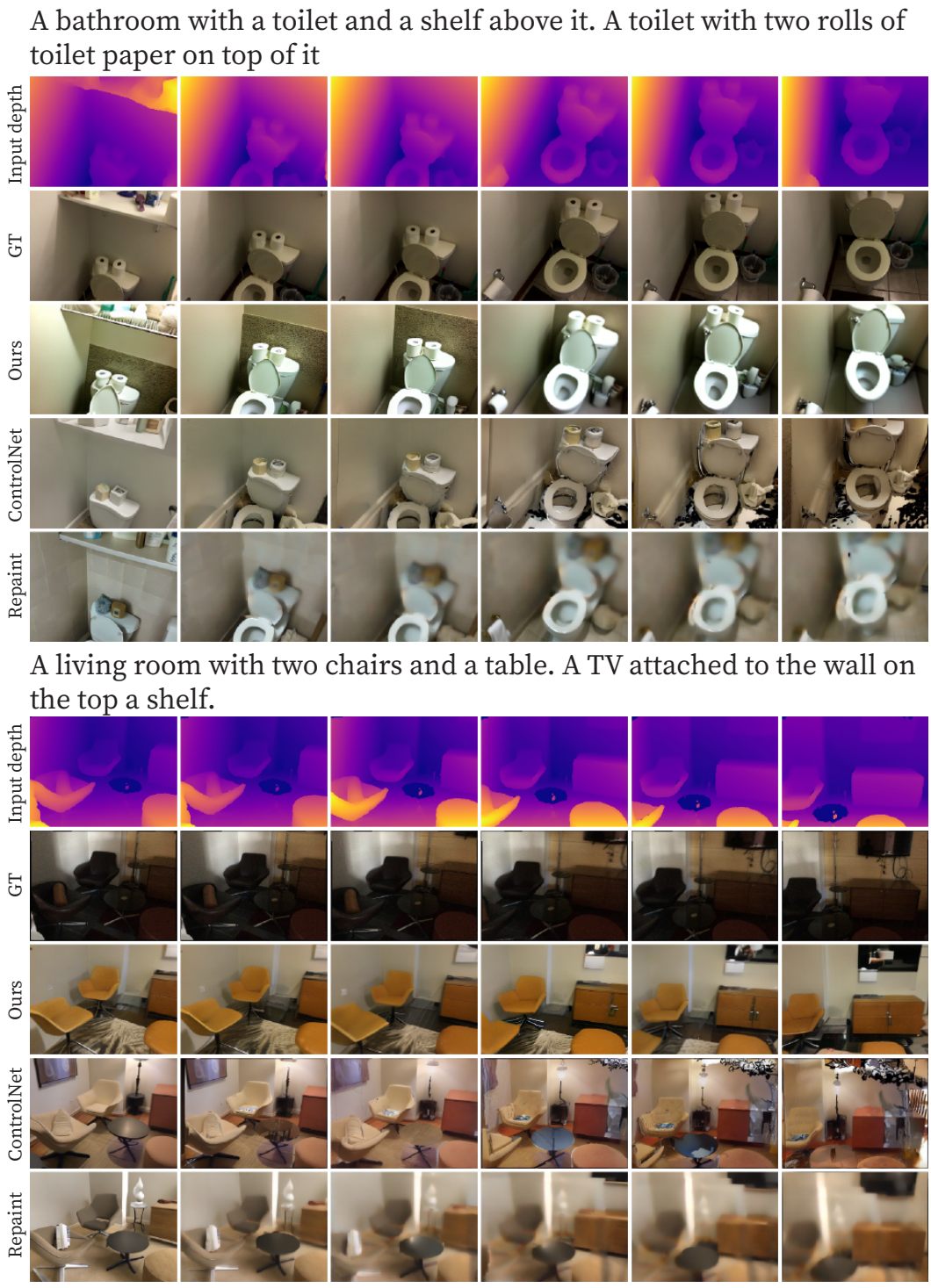

Figure 15: Addition results for depth-to-image generation.

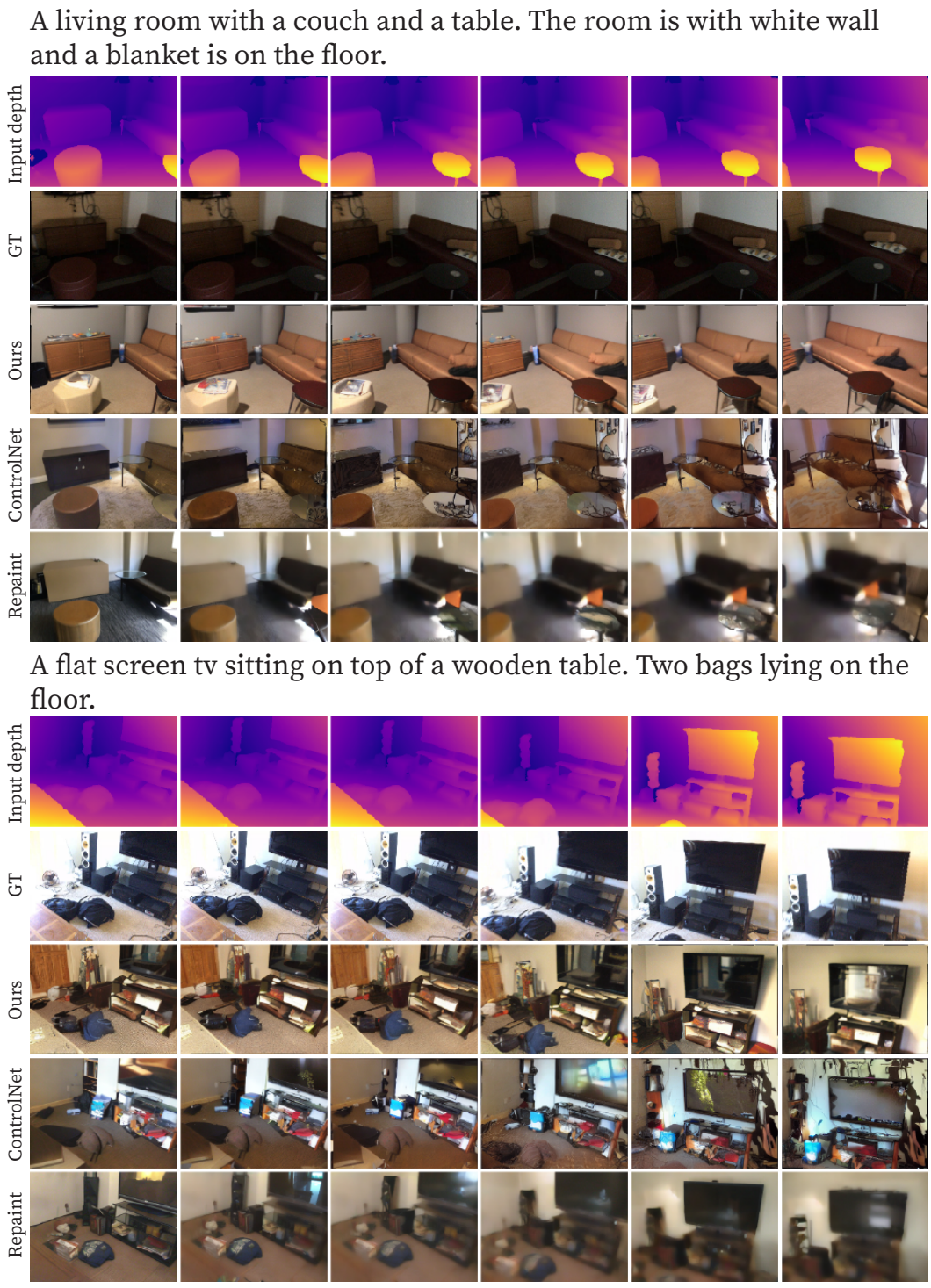

Figure 16: Addition results for depth-to-image generation.

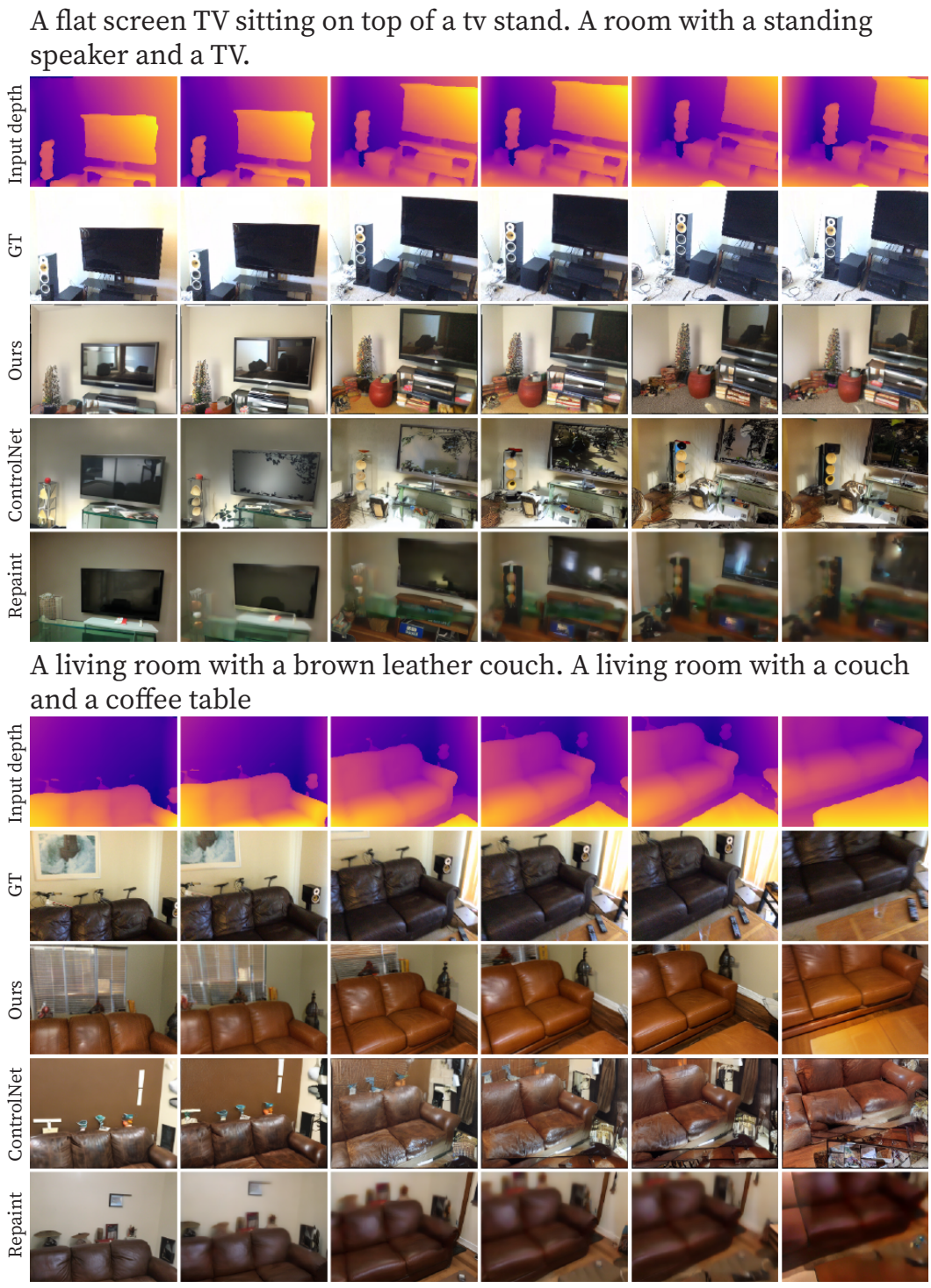

Figure 17: Addition results for depth-to-image generation.

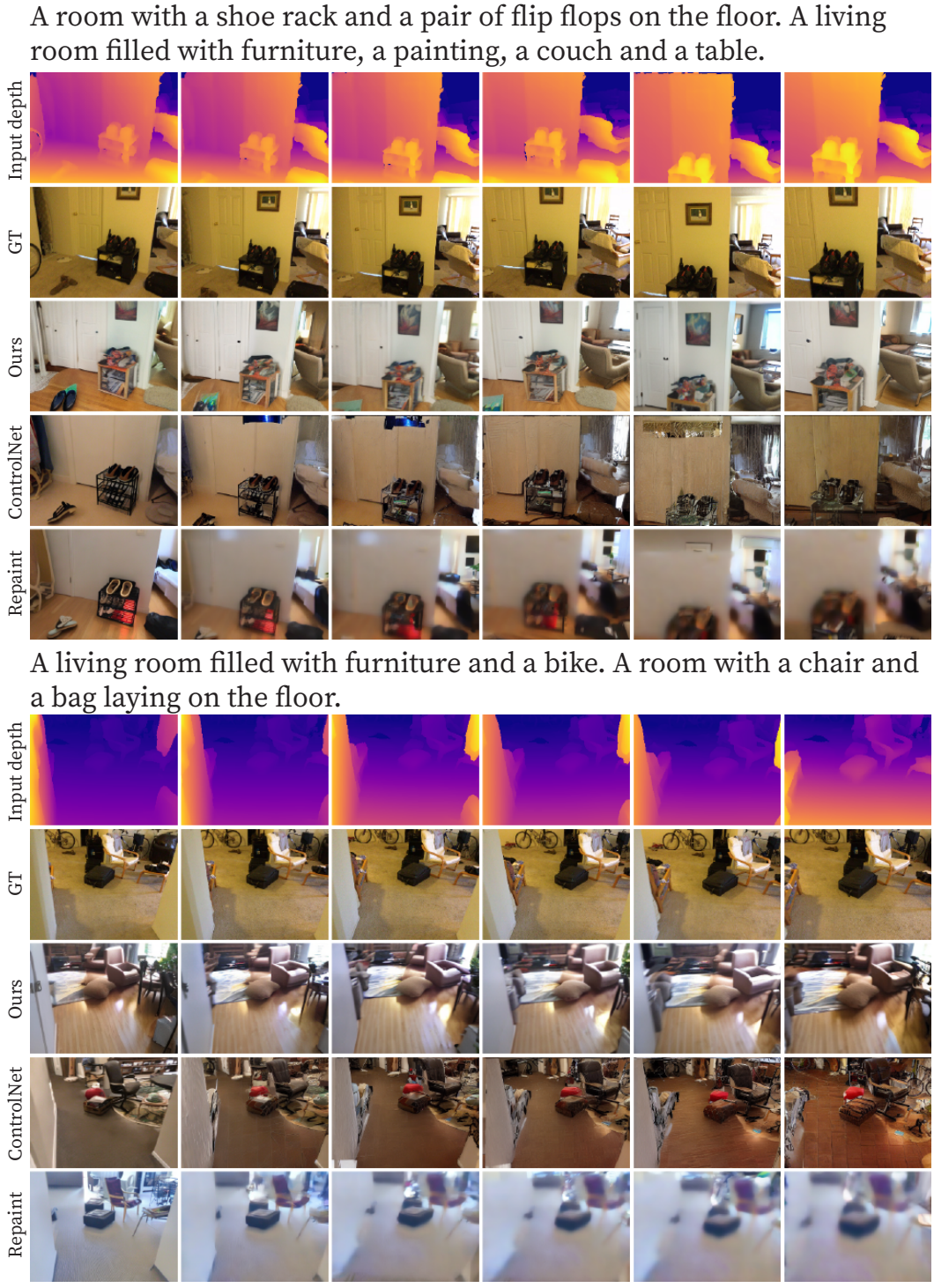

Figure 18: Addition results for depth-to-image generation.

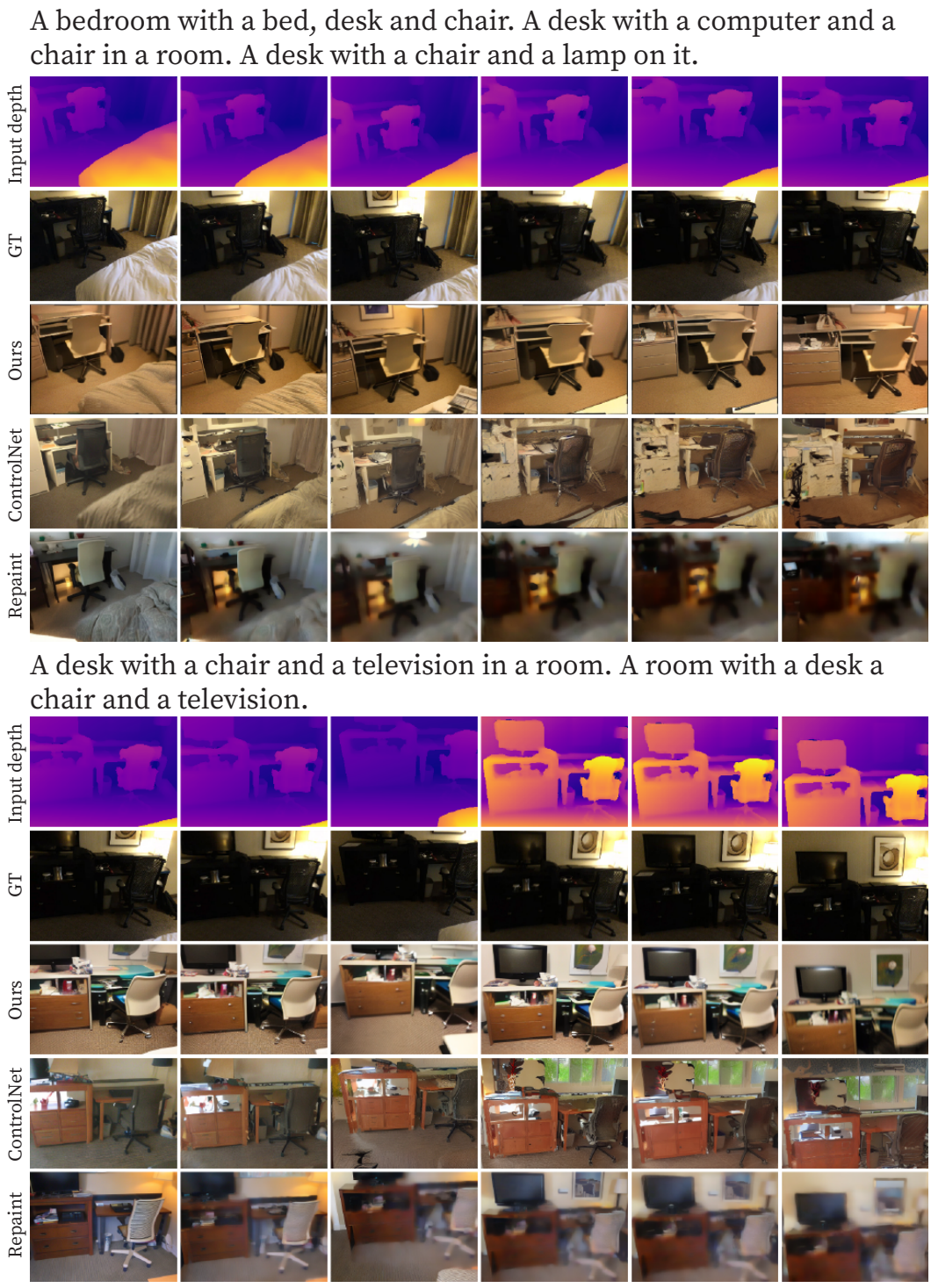

Figure 19: Addition results for depth-to-image generation.

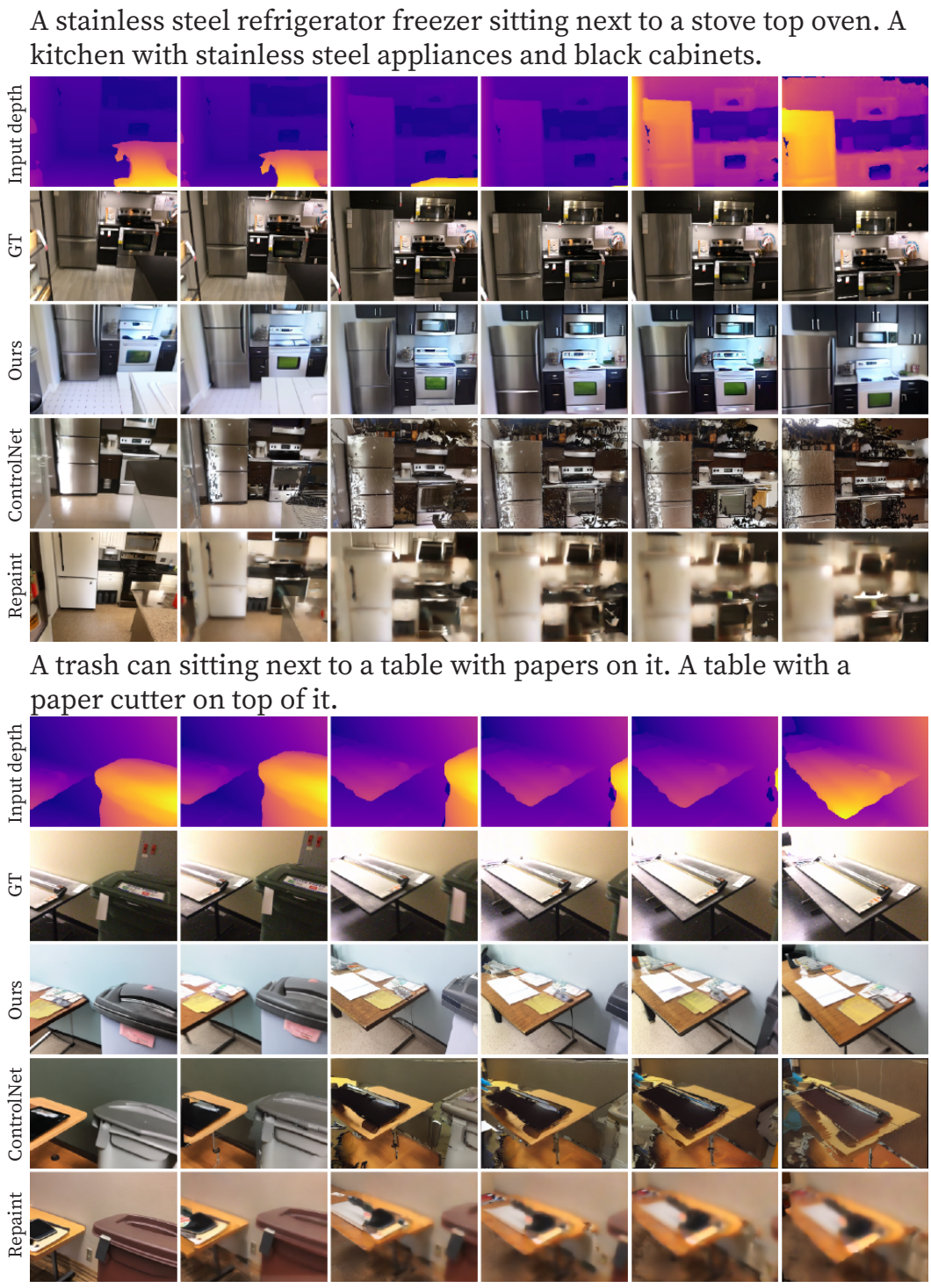

Figure 20: Addition results for depth-to-image generation.

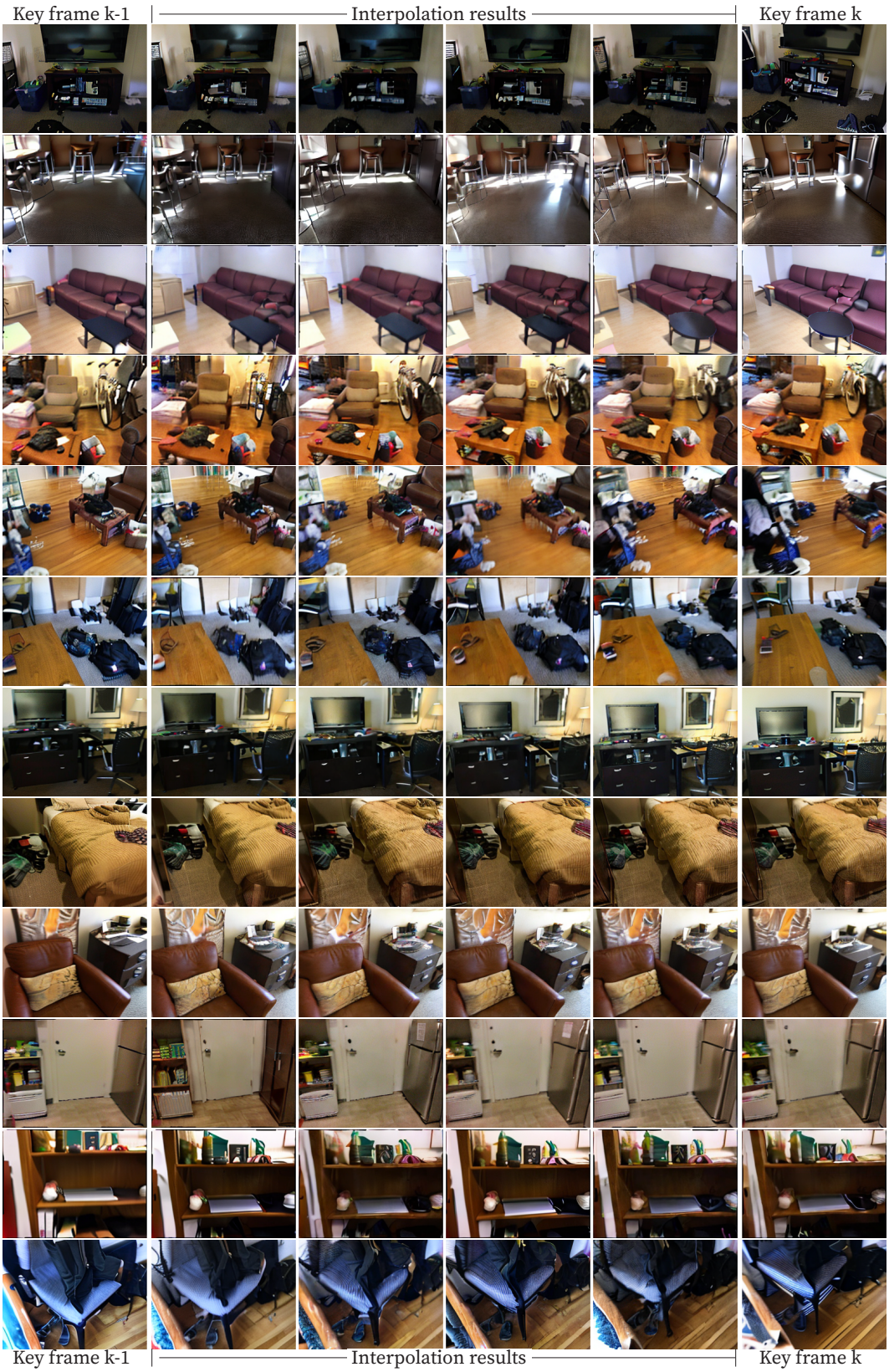

Figure 21: Addition results for interpolated frames.