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

Shitao Tang[1*] **Fuyang Zhang**[1*] **Jiacheng Chen**[1] **Peng Wang**[2] **Yasutaka Furukawa**[1]
[1]Simon Fraser University [2]Bytedance

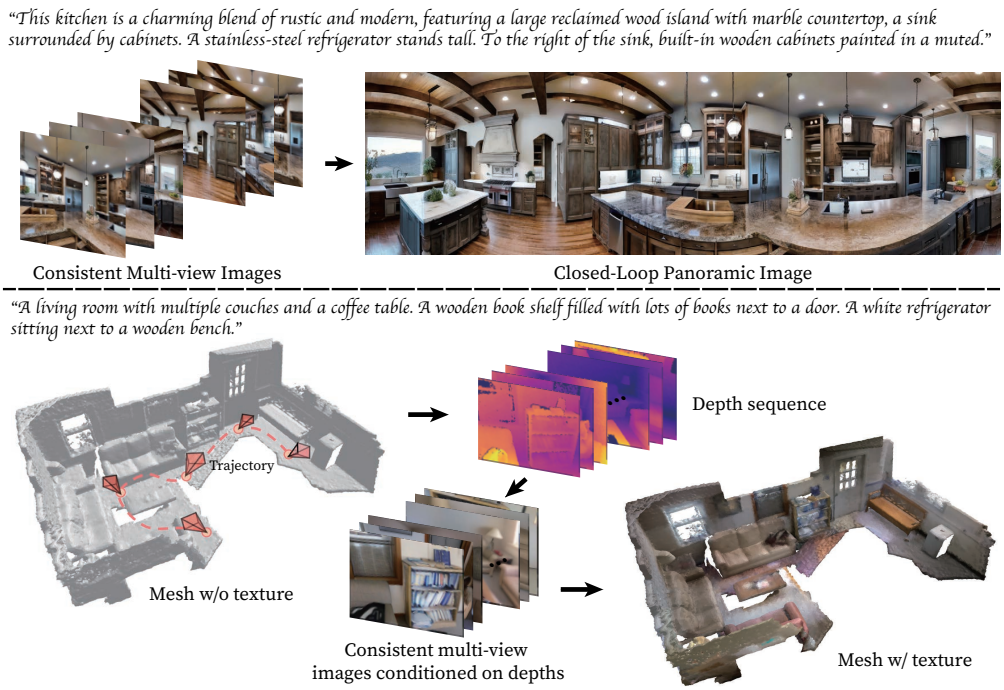

Figure 1: MVDiffusion synthesizes consistent multi-view images. *Top:* generating perspective crops which can be stitched into panorama; *Bottom:* generating coherent multi-view images from depths.

Abstract

This paper introduces *MVDiffusion*, a simple yet effective multi-view image generation method for scenarios where pixel-to-pixel correspondences are available, such as perspective crops from panorama or multi-view images given geometry (depth maps and poses). Unlike prior models that rely on iterative image warping and inpainting, MVDiffusion concurrently generates all images with a global awareness, encompassing high resolution and rich content, effectively addressing the error accumulation prevalent in preceding models. MVDiffusion specifically incorporates a correspondence-aware attention mechanism, enabling effective cross-view interaction. This mechanism underpins three pivotal modules: 1) a generation module that produces low-resolution images while maintaining global correspondence, 2) an interpolation module that densifies spatial coverage between images, and 3) a super-resolution module that upscales into high-resolution outputs. In terms of panoramic imagery, MVDiffusion can generate high-resolution photorealistic images up to 1024×1024 pixels. For geometry-conditioned multi-view image generation, MVDiffusion demonstrates the first method capable of generating a textured map of a scene mesh. The project page is at `https://mvdiffusion.github.io/`.

*Equal contribution. Contact the authors at shitaot@sfu.ca.

Preprint. Under review.

1 Introduction

Photorealistic image synthesis aims to generate highly realistic images, enabling broad applications in virtual reality, augmented reality, video games, and filmmaking. The field has seen significant advancements in recent years, driven by the rapid development of deep learning techniques such as diffusion-based generative models [2, 18, 23, 41, 45, 46, 47].

One particularly successful domain is text-to-image generation. Effective approaches include generative adversarial networks [4, 14, 21], autoregressive transformers [12, 37, 52], and more recently, diffusion models [17, 19, 36, 39]. DALL-E 2 [36], Imagen [39] and others generate photorealistic images with large-scale diffusion models. Latent diffusion models [38] apply the diffusion process in the latent space, allowing more efficient computations and faster image synthesis.

Despite the impressive progress, multi-view text-to-image synthesis is still a difficult problem, with computational efficiency and multi-view consistency challenges. A common approach involves an autoregressive generation process [6, 13, 20], where the generation of the n-th image is conditioned on the $(n-1)$-th image through image warping and inpainting techniques. However, this autoregressive approach results in accumulated errors and does not handle loop closure [13]. Moreover, the reliance on the previous image may pose challenges for complex scenarios or large viewpoint variations.

This paper generates images simultaneously using a latent diffusion model pretrained on perspective images. We retain the original stable diffusion model while incorporating a correspondence-aware attention (CA) mechanism between U-Net blocks associated with different views. This attention mechanism serves three key modules: 1) a generation module generating low-resolution images, where the attention enforces consistency among generated images; 2) an interpolation module that takes the low-resolution images produced by the generation module and creates dense interpolations between them, utilizing the same network architecture and weights as the generation module 3) a super-resolution module upscaling to high-resolution, where the correspondence-aware attention enforces consistency among the high-res images and the conditioning on the low-res images.The multi-view consistency or conditioning has pixel-to-pixel correspondences where correspondence-aware attention is applicable.

Additionally, we propose a metric to quantify multi-view consistency, which computes the Peak Signal-to-Noise Ratio (PSNR) for overlapping regions in both generated and natural image pairs. The consistency is then assessed by contrasting these PSNR values across the entire test set.

In summary, the paper makes three contributions: 1) A multi-view image generation architecture, based on a latent diffusion model pretrained on perspective images, that simultaneously generates consistent multiple images; 2) State-of-the-art performance on generating panoramas and multi-view images conditioned on geometry. To the best of our knowledge, MVDiffusion is the first method capable of generating a textured map of a scene mesh; and 3) A novel metric for evaluating multi-view consistency of generated images.

2 Related Work

Diffusion models.

Diffusion models [18, 23, 41, 44, 45, 46, 47] (DM) or score-based generative models are the essential theoretical framework of the exploding generative AI. Early works achieve superior sample quality and density estimation [10, 47] but require a long sampling trajectory. Advanced sampling techniques [22, 28, 42] accelerate the process while maintaining generation quality. Latent diffusion models [38] (LDMs) apply DM in a compressed latent space to reduce the computational cost and memory footprint, making the synthesis of high-resolution images feasible on consumer devices. We enable holistic multi-view image generation by the latent diffusion model.

Image generation. Diffusion Models (DM) dominate content generation. Foundational work such as DALL-E 2 [36], GLIDE [32], LDMs [38], and Imagen [39] have showcased significant capabilities in text-conditioned image generation. They train on extensive datasets and leverage the power of pre-trained language models. These large text-to-image Diffusion Models also establish strong foundations for fine-tuning towards domain-specific content generation. For instance, MultiDiffusion [1] and DiffCollage [56] failitates 360-degree image generation. However, the resulting images are not true panoramas since they do not incorporate camera projection models. Text2Light [7] synthesizes

HDR panorama images from text using a multi-stage auto-regressive generative model. However, the leftmost and rightmost contents are not connected (i.e., loop closing).

3D content generation. Content generation technology has profound implications in VR/AR and entertainment industries, driving research to extend cutting-edge generation techniques from a single image to multiple images. Dreamfusion [33] and Magic3D [26] distill pre-trained Diffusion Models into a NeRF model [30] to generate 3D objects guided by text prompts. However, these works focus on objects rather than scenes. In the quest for scene generation, another approach [20] generates prompt-conditioned images of indoor spaces by iteratively querying a pre-trained text-to-image Diffusion Model. SceneScape [13] generates novel views on zoom-out trajectories by employing image warping and inpainting techniques using diffusion models. Text2Room [20] adopts similar methods to generate a complete textured 3D room geometry. However, the generation of the n-th image is solely conditioned on the local context, resulting in accumulation errors and less favorable results. Our research takes a holistic approach and generates consistent multi-view images given camera poses and text prompts while fine-tuning pre-trained perspective-image Diffusion Models.

3 Preliminary

Latent Diffusion Models(LDM) [38] is the foundation of our method. LDM consists of three components: a variational autoencoder (VAE) [24] with encoder $\mathcal{E}$ and decoder $\mathcal{D}$, a denoising network ϵ_θ, and a condition encoder τ_θ. High-resolution images $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$ are mapped to a low-dimensional latent space by $\mathbf{Z} = \mathcal{E}(\mathbf{x})$, where $\mathbf{Z} \in \mathbb{R}^{h\times w\times c}$. The down-sampling factor $f = H/h = W/w$ is set to 8 in the popular Stable Diffusion. The latents are converted back to the image space by $\tilde{\mathbf{x}} = \mathcal{D}(\mathbf{Z})$. The LDM training objective is given as:

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(\mathbf{x}),\mathbf{y},\boldsymbol{\epsilon}\sim\mathcal{N}(0,1),t}\left[\|\boldsymbol{\epsilon} - \epsilon_\theta(\mathbf{Z}_t, t, \tau_\theta(\mathbf{y}))\|_2^2\right], \tag{1}$$

t is uniformly sampled from 1 to T. The denoising network ϵ_θ is a time-conditional U-Net [10], augmented with cross-attention mechanisms to incorporate the optional condition encoding $\tau_\theta(\mathbf{y})$. $\mathbf{y}$ can be a text prompt, an image, *etc*. At sampling time, the denoising (reverse) process generates samples in the latent space, and the decoder produces high-resolution images with a single forward pass. Advanced samplers [22, 28, 42] are employed to accelerate the sampling process of latents.

4 MVDiffusion: Holistic Multi-view Image Generation

MVDiffusion addresses the challenge of multi-image generation in scenarios where pixel-to-pixel correspondences are available. These scenarios include panoramas and multi-view images accompanied by camera poses and depths. Specifically, 1) In the case of a panorama, the image is generated from N perspective images that cover the panoramic field of view. The consistency of these images is achieved through pixel-to-pixel correspondences, realized by a 3×3 homography matrix for each image pair. 2) For multi-view depth-to-image, the image set is generated with the given camera poses. Here, pixel-to-pixel correspondences are realized through an unprojection and projection operation using the depth information. Our method operates in the latent feature space whose resolution is 1/8 of the images, where the VAE decoder of the Stable Diffusion [54] turns final feature maps to the output images. In the following sections, we elaborate on the correspondence-aware attention mechanism and the three network modules that constitute our multi-view latent diffusion model.

4.1 Correspondence-aware Attention

We have devised a "correspondence-aware attention" mechanism that enforces correspondence constraints across multiple feature maps (See Figure 3). This mechanism operates by taking into account the source feature maps $\mathbf{F}$ and N target feature maps $\mathbf{F}^l$ ($l \in [0, N-1]$). For a token located at position $(\mathbf{s})$ in the source feature map, we compute a message based on the corresponding pixels $\{\mathbf{t}^l\}$ in the target feature maps $\{\mathbf{F}^l\}$ (not necessarily at integer coordinates) with local neighborhoods. Concretely, for each target pixel $\mathbf{t}^l$, we consider a $K \times K$ neighborhood $\mathcal{N}(\mathbf{t}^l)$ by adding integer displacements (d_x/d_y) to the (x/y) coordinate, where $|d_x| < K/2$ and $|d_y| < K/2$. In practice, we use $K = 3$ with a neighborhood of 9 points for panorama for quality and $K = 1$ for geometry-

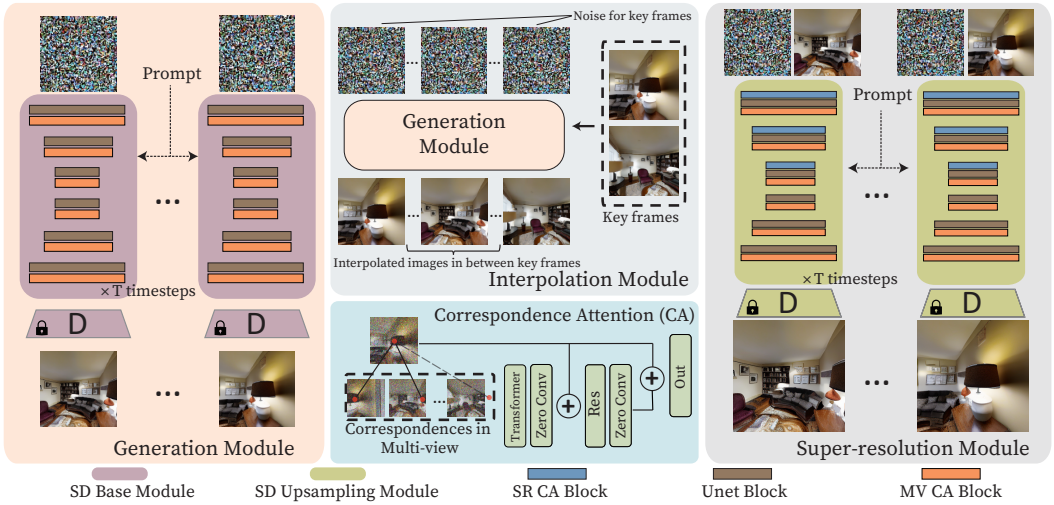

Figure 2: **MVDiffusion** consists of three modules: 1) the generation module, which generates low-resolution images while preserving global correspondences, 2) the interpolation module, designed to densify the spatial coverage between two images (key frames), and 3) the super-resolution module, tasked with upscaling low-resolution images to high-resolution outputs. Both the interpolation module and the generation module share the same U-Net architecture and weights. The Multi-View Correspondence-Aware Attention blocks (MV CA block) are served to connect different views. For the super-resolution module, additional SR CA blocks are incorporated to aggregate information from low-resolution images.

conditioned multi-view image generation for efficiency. The message $\mathbf{m}$ is calculated as

$$\mathbf{m} = \sum_{l} \sum_{t_*^l \in \mathcal{N}(\mathbf{t}^l)} \text{SoftMax}\left(\left[\mathbf{Q}\bar{\mathbf{F}}(\mathbf{s})\right] \cdot \left[\mathbf{K}\bar{\mathbf{F}}^l(t_*^l)\right]\right) \mathbf{V}\bar{\mathbf{F}}^l(t_*^l), \tag{2}$$

$$\bar{\mathbf{F}}(\mathbf{s}) = \mathbf{F}(\mathbf{s}) + \boldsymbol{\gamma}(0), \quad \bar{\mathbf{F}}^l(t_*^l) = \mathbf{F}^l(t_*^l) + \boldsymbol{\gamma}(\mathbf{s}_*^l - \mathbf{s}). \tag{3}$$

The message calculation follows the standard attention mechanism that aggregates information from the target feature pixels $\{t_*^l\}$ to the source (s). $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the query, key and value matrices. The key difference is the added position encoding $\gamma(\cdot)$ to the target feature $\mathbf{F}^l(t_*^l)$ based on the 2D displacement (panorama) or 1D depth error (geometry) between its corresponding location $\mathbf{s}_*^l$ in the source image and $\mathbf{s}$. In panorama generation, the displacement provides the relative location in the local neighborhood. In the depth-to-image generation case, the displacement provides cues about depth discontinuities or occlusions, crucial for high-fidelity image generation. Note that a displacement is a 2D vector, and we apply a standard frequency encoding [53] to the displacement in both x and y coordinates, then concatenate. A target feature $\mathbf{F}^l(t_*^l)$ is not at an integer location and is obtained by bilinear interpolation.

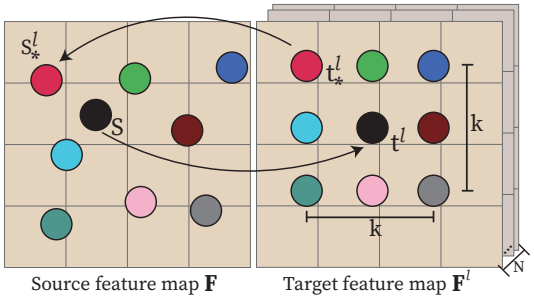

Figure 3: Correspondence-aware attention

4.2 Multi-view Latent Diffusion Model

We design three modules with the correspondence-aware attention (CA) mechanism to facilitate multi-view image generation, given a textual prompt and the view information of the output images.

• The *generation module* initializes the latent of output images with Gaussian noise and generates corresponding images via an SD pipeline under textual conditions. To ensure multi-view consistency, correspondence-aware attention layers are introduced and fine-tuned at every U-Net block.

• The *interpolation module* is inspired by the recent VideoLDM [2], taking a pair of generated images, "key-frames", as the conditions, and generates the in-between images, where the CA layers are added between every pair of pre-generated key-frame feature maps and/or in-between feature maps to enforce multi-view consistency.

• The *super-resolution module* takes a set of generated images as the condition, initializes high-resolution latent by the Gaussian noise, and generates high-resolution images, where the CA layers are added between every pair of feature maps and/or high-resolution feature maps within the U-Net block to enforce multi-view consistency.

In the panorama generation task, a panorama image is formed by eight perspective views, where the horizontal field of view is 90 degrees with an overlap of 45 degrees. The generation module generates eight 256×256 images, and the super-resolution module upscales to eight 1024×1024 images. Regarding the depth-to-image task, it involves the integrated use of the generation and interpolation modules to produce images. The data required for these tasks, specifically the input camera poses and depth details, are extracted from ScanNet scenes, as described in Dai et al. (2017) [9]. The process initiates with the generation module creating key images, which are then densified for more detailed representation via the interpolation module.

Generation module. The module generates a set of low-resolution images through simultaneous denoising - with dimensions 256×256 for the panorama case and 192×256 for the multiview images with depth case. Throughout each denoising step, the noisy images are fed into a multi-branch U-Net architecture to predict noises. Within each U-Net block, we have integrated a transformer block equipped with correspondence attention, followed by an additional residual block. To retain the inherent capabilities of the stable diffusion model, we initialize the final linear layer of the transformer block and the final convolutional layer of the residual block to be zero, as suggested in ControlNet [55]. This initialization strategy ensures that our modifications do not disrupt the original functionality of the stable diffusion model. Note that our system allows one to specify different text prompts to different perspective images, enhancing controls over the generated content.

Interpolation module. The interpolation module of MVDiffusion, inspired by VideoLDM [2], creates n images between a pair of 'key frames', which have been previously generated by the generation module. This model utilizes the same unet structure and correspondence attention weights as the generation model, with extra convolutional layers, and it reinitializes the latent of both the in-between images and key images using Gaussian noise. A distinct feature of this module is that the Unet branch of key images is conditioned on images already generated, as depicted in Figure 2. Specifically, this condition is incorporated into every Unet block. In the Unet branch of key images, the generated images are concatenated with a mask of ones (4 channels) , and then a zero convolution operation is used to downsample the image to the corresponding feature map size. These downsampled conditions are subsequently added to the input of the Unet blocks. For the branch of in-between images, we take a different approach. We append a black image, with pixel values of zero, to a mask of zeros, and apply the same zero convolution operation to downsample the image to match the corresponding feature map size. These downsampled conditions are also added to the input of the Unet blocks. This procedure essentially trains the module such that when the mask is one, the branch regenerates the conditioned images, and when the mask is zero, the branch generates the in-between images.

Super-resolution module. The module takes generated images and upscales to a higher resolution. We first use the VAE encoder to convert the input (low-resolution) images to image latent, followed by two convolutional layers with 3×3 kernels to increase the channel dimension from 4 to 320 and obtain low-resolution image features. We initialize high-resolution image latents by Gaussians and employ a lighter Unet, compared to the one of the generation module, to generate high-resolution images, where the CA layers connect low-resolution image features and the high-resolution feature maps. Concretely, within each UNet encoder block, we apply one convolutional layer to these extracted features to match the Unet channel dimension, and we introduce a one-directional correspondence transformer block from low-resolution to high-resolution images, SP CA block in Figure 2. This additional correspondence attention layer intensifies the detail and fidelity in the high-resolution output, effectively capitalizing on the coarse global structure acquired in the initial stage.

Last but not least, training a super-resolution model directly on high-resolution images is not feasible due to the computational demand. During training, we only use a 512×512 image patch at the center of each 1024×1024 high-resolution image. During inference, we apply the trained model to

the full extent of the images. This approach allows us to efficiently handle high-resolution images without compromising the performance and effectiveness of our super-resolution model.

Training. We first train the generation module, and then the interpolation, while they share the same Unet and correspondence-aware weights. The super-resolution module is trained separately. For each module, the training process comprises two phases. In the first phase, we fine-tune the stable diffusion U-Net model while freezing the VAE components that were trained on single-view perspective images. This approach allows the model to acquire knowledge from a broader spectrum of content. In the second phase, we integrate the CA transformer block with zero convolution into the network to enforce multi-view consistency or conditioning. Only the additional CA transformer blocks are trained, while the pre-trained weights are fixed.

During each training step, we uniformly sample a shared noise level t from 1 to T for multi-view images. We employ the following loss function for both the generation and the super-resolution modules, where ϵ_θ^i is the estimated noise and $\mathbf{Z}_t^i$ is the noisy latent for the i-th image:

$$L_{\text{MVDiffusion}} := \mathbb{E}_{\left\{\mathbf{Z}_t^i=\mathcal{E}(\mathbf{x}^i)\right\}_{i=1}^N,\{\boldsymbol{\epsilon}^i\sim\mathcal{N}(0,1)\}_{i=1}^N,\mathbf{y},t}\Big[\sum_{i=1}^{N}\|\boldsymbol{\epsilon}^i-\epsilon_\theta^i(\left\{\mathbf{Z}_t^i\right\},t,\tau_\theta(\mathbf{y}))\|_2^2\Big]. \tag{4}$$

5 Experiments

We evaluate MVDiffusion on two tasks: panorama image generation and multi-view image generation with camera poses and depths. We first describe implementation details and the evaluation metrics.

Implementation details. We have implemented the system with PyTorch while using publicly available Stable Diffusion code from Diffusers [54]. The model consists of a denoising U-Net to execute the denoising process within a compressed latent space and a VAE to connect the image and latent spaces. The pre-trained VAE of the Stable Diffusion is maintained with official weights and is used to encode images during the training phase and decode the latent codes into images during the inference phase. We have used a machine with 4 NVIDIA RTX A6000 GPUs for both training and inference. We strategically select the modules to validate our system due to the time-consuming nature of inferring with all three modules. For panorama image generation, we employ the generation module followed by the super-resolution module. For multi-view depth-to-image generation, we combine the generation module, followed by either the interpolation module. Specific details and results of these varying configurations are provided in the corresponding sections.

Evaluation matrics. The evaluation metrics cover two aspects, image quality of generated images and their consistency.

• *Image quality* is measured by Fréchet Inception Distance (FID) [16], Inception Score (IS) [40], and CLIP Score (CS) [34]. FID measures the distribution similarity between the features of the generated and real images. The Inception Score is based on the diversity and predictability of generated images. CLIP Score measures the text-image similarity using pretrained CLIP models [35].

• *Multi-view consistency* is measured by a novel metric based on pixel-level similarity. The area of multi-view image generation is still in an early stage, and there is no common metric for multi-view consistency. We propose a new metric based on Peak Signal-to-Noise Ratio (PSNR). Concretely, given multi-view images, we compute the PSNR between all the overlapping regions and then compare this "overlapping PSNR" for ground truth images and generated images. The final score is defined as the ratio of the "overlapping PSNR" of generated images to that of ground truth images. Higher values indicate better consistency.

The rest of the section explains the experimental settings and results more, while the full details are referred to the supplementary.

5.1 Panorama image generation

This task generates perspective crops covering the panoramic field of view, where the challenge is to ensure consistency in the overlapping regions.

Matterport3D [5] is a comprehensive indoor scene dataset that consists of 90 buildings with 10,912 panorama images. We allocate 9820 and 1092 panoramas for training and evaluation, respectively.

Figure 4: Qualitative evaluation for panorama generation. The red box indicates the area stitched leftmost and rightmost content. More results are available in the supplementary material.

Baselines. We have selected three related state-of-the-art methods for thorough comparisons. The details of the baselines are briefly summarized as follows (full implementation details can be found in the appendix):

- *Text2Light*[7] creates HDR panorama images from text using a multi-stage auto-regressive generative model. To obtain homographic images, we project the generated panoramas into perspective images using the same camera settings (FoV=90°, rotation=45°).
- *Stable Diffusion (SD)*[38] is a text-to-image model capable of generating high-quality perspective images from text. For comparison, we fine-tuned Stable Diffusion using panorama images and then extracted perspective images in a similar manner.
- *Inpainting methods* [13, 20] operate through an iterative process, warping generated images to the current image and using an inpainting model to fill in the unknown area. Specifically, we employed the inpainting model from Stable Diffusion v2 [38] for this purpose.

Results. Table 1 and Figure 4 present the quantitative and qualitative evaluations, respectively. We then discuss the comparison between MVDiffusion and each baseline:

• *Compared with Text2Light[7]*: Text2Light is based on auto-regressive transformers and shows low FID, primarily because diffusion models perform generally better. Another drawback is the inconsistency between the left and the right panorama borders, as illustrated in Figure 4.

• *Compared with Stable Diffusion (panorama)*[38]: We achieve higher IS and CS but slightly lower FID. This discrepancy could be attributed to the inherent randomness in the generation process and the limitations of FID [3]. More importantly, this model suffers from the same issue as Text2light – the left and right borders are inconsistent. In contrast, our method produces flawless panoramas by explicitly enforcing consistency with correspondence-aware attention. In addition, the vanilla stable diffusion cannot conduct multi-view depth-to-image generation.

Method	FID ↓	IS ↑	CS ↑
Impainting [13, 20]	42.13	7.08	29.05
Text2light [7]	48.71	5.41	25.98
SD (Pano) [38]	**23.02**	6.58	28.63
SD (Perspective) [38]	25.59	7.29	30.25
MVDiffusion(Ours)	23.30	**7.10**	**29.81**

Table 1: Quantitative evaluation with Fréchet Inception Distance (FID), Inception Score (IS), and CLIP Score (CS).

• *Compared with inpainting method* [13, 20]: Inpainting methods also exhibit worse performance due to the error accumulations, as evidenced by the gradual style change throughout the generated image sequence in Figure 4.

• *Compared with Stable Diffusion (perspective)*[38]: We also train the original stable diffusion on perspective images of Matterport3D and evaluate it on the same test set. This method cannot generate

multi-view images but is a good reference for performance comparison. The results suggest that our method does not incur performance drops when adapting SD for multi-view image generation.

Figure 5: Qualitative evaluation for depth-to-image generation. More results are available in the supplementary material.

5.2 Multi view depth-to-image generation

This task converts a sequence of depth images into a sequence of RGB images while preserving the underlying geometry and maintaining multi-view consistency. ScanNet is an indoor RGB-D video dataset comprising over 1513 training scenes and 100 testing scenes, all with known camera parameters. We train our model on the training scenes and evaluate it on the testing scenes. In order to construct our training and testing sequences, we initially select key frames, ensuring that each consecutive pair of key frames has an overlap of approximately 65%. Each training sample consists of 12 sequential keyframes. For evaluation purposes, we conduct two sets of experiments. For our quantitative evaluation, we have carefully chosen 590 non-overlapping image sequences from the test set, each composed of 12 individual images. In terms of qualitative assessment, we first employ the generation module to produce all the key frames within a given test sequence. Following this, the interpolation module is utilized to enrich or densify these images. Notably, even though our model has been trained using a frame length of 12, it has the capability to be generalized to accommodate any number of frames. Ultimately, we fuse the RGBD sequence into a cohesive scene mesh.

Baselines. To our knowledge, no direct baselines exist for scene-level depth-to-image generation. Some generate 3D textures for object meshes, but often require complete object geometry to optimize the generated texture from many angles [6, 31]. This is unsuitable for our setting where geometry is provided for the parts visible in the input images. Therefore, we have selected two baselines:

Method	FID ↓	IS ↑	CS ↑
RePaint [29]	70.05	7.15	26.98
ControlNet [55]	43.67	7.23	28.14
Ours	**23.10**	**7.27**	**29.03**

Table 2: Comparison in Fréchet Inception Distance (FID), Inception Score (IS), and CLIP Score (CS) for multiview depth-to-image generation.

• *RePaint*[29]: This method uses an image diffusion model for inpainting holes in an image, where we employ the depth2image model from Stable Diffusion v2[38]. For each frame, we warp the generated images to the current frame and employ the RePaint technique [29] to fill in the holes. This baseline model is also fine-tuned on our training set.

Task →	Panorama		Depth-to-image	
Method	PSNR ↑	Ratio ↑	PSNR ↑	Ratio ↑
G.T.	37.7	1.00	21.41	1.00
SD (Persp.)	10.6	0.28	11.2	0.44
Ours	**31.0**	**0.82**	**17.41**	**0.76**

Table 3: Multi-view consistency for panorama generation and multi-view depth-to-image generation.

Method	CA	PSNR↑	Ratio↑
GT	-	37.7	1.00
GM	✗	10.6	0.28
	✓	25.6	0.68
Ours	✗	28.3	0.75
	✓	**31.0**	**0.82**

Table 4: Ablation study on multiview consistency with Generation Module (GM) only or full system.

• *Depth-conditioned ControlNet*: We train a depth-conditioned ControlNet model combined with the inpainting Stable Diffusion method with the same training set as ours. The implementation is based on a public codebase [8]. This model is capable of image inpainting conditioned on depths. We use the same pipeline as the above method.

Results. Table 2, Figure 5, Figure 6, and Figure 7 present the quantitative and qualitative results of our generation and interpolation modules. Our approach achieves a notably better FID. As depicted in Figure 5, the repaint method generates progressively blurry images, while ControlNet produces nonsensical content when the motion is large. These issues arise since both methods depend on partial results and local context during inpainting, causing error propagation. Our method overcomes these challenges by enforcing global constraints with the correspondence-aware attention and generating images simultaneously. Figure 6 exhibits the keyframes at the left and the right, where the intermediate frames are generated in the middle, which are consistent throughout the sequence. Figure 7 illustrates the textured scene meshes produced by our method. To the best of our knowledge, our approach is the first of its kind that is capable of generating a complete scene mesh.

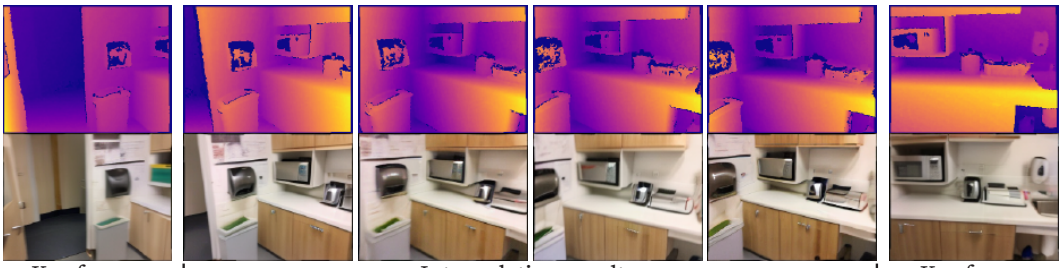

Figure 6: Visualization of interpolated frames. We interpolate 4 frames (second image to fifth image). More visualization results are presented in supplementary materials.

Figure 7: Mesh visualization. We use the generation and interpolation module to generate RGB images given depths/poses and then fuse them into mesh with TSDF fusion.

5.3 Measuring multi-view consistency

The multi-view consistency is evaluated with our novel metric as explained earlier. For panorama image generation, we select image pairs with a rotation angle of 45 degrees and resize them to 1024×1024. For multi-view depth-to-image generation, consecutive image pairs are used and resized to 192×256. PSNR is computed among all pixels within overlapping regions for panorama image generation. For multi-view depth-to-image generation, a depth check discards pixels with depth errors above $0.5m$, the PSNR is then computed on the remaining overlapping pixels.

Results. In Table 3, we first use the real images to set up the upper limit, yielding a PSNR ratio of 1.0. We then evaluate our generation model without the correspondence attention (i.e., an original stable diffusion model), effectively acting as the lower limit. Our method, presented in the last row, achieves a PSNR ratio of 0.82 and 0.76 for the two tasks respectively, confirming an improved multi-view consistency.

Ablation Studies. Table 4 justifies the effectiveness of correspondence-aware attention on panorama image generation. Our generation module achieves much better results when incorporating correspondence-aware attention. For the SR, comparing performances with and without correspondence-aware attention also shows notable improvements.

6 Conclusion

This paper presents MVDiffusion, a method that concurrently generates consistent multi-view images. Our key innovation is a correspondence-aware attention (CA) mechanism that learns to enforce cross-view consistency given pixel-to-pixel correspondences, which leads to three modules: 1) a generation module to produce consistent low-resolution images; 2) an interpolation module to densify spatial coverage; and 3) a super-resolution module to produce high-resolution images. By composing the three modules, our experiments show that MVDiffusion achieves state-of-the-art performance in panorama generation and multi-view depth-to-image generation, effectively mitigating the issue of accumulation error of previous approaches. Furthermore, our high-level idea has the potential to be extended to other generative tasks like video prediction or 3D object generation, opening up new avenues for the content generation of more complex and large-scale scenes.

Limitations. The primary limitation of MVDiffusion lies in its computational time and resource requirements. Despite using advanced samplers, our modules need at least 50 steps to generate high-quality images, which is a common bottleneck of all DM-based generation approaches. Additionally, the memory-intensive nature of MVDiffusion, resulting from the parallel denoising limits its scalability. This constraint poses challenges for its application in more complex applications that require a large number of images (*e.g.*, long virtual tour).

Broader impact. MVDiffusion enables the generation of detailed environments for video games, virtual reality experiences, and movie scenes directly from written scripts, vastly speeding up production and reducing costs. However, like all techniques for generating high-quality content, our method might be used to produce disinformation.

Acknowledgements. This research is partially supported by NSERC Discovery Grants with Accelerator Supplements and DND/NSERC Discovery Grant Supplement, NSERC Alliance Grants, and John R. Evans Leaders Fund (JELF). We thank the Digital Research Alliance of Canada and BC DRI Group for providing computational resources.

References

[1] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2, 2023.

[2] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023.

[3] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179: 41–65, 2019.

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[5] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

[6] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023.

[7] 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.

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

[9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[11] 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.

[12] 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.

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

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144, 2020.

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

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[17] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022.

[20] 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.

[21] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34: 852–863, 2021.

[22] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

[23] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

[24] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[25] 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.

[26] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *arXiv preprint arXiv:2211.10440*, 2022.

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

[28] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022.

[29] 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.

[30] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[31] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–8, 2022.

[32] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[33] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[36] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[37] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[38] 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.

[39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[40] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[41] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*. PMLR, 2015.

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

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

[44] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems (NeurIPS)*, 2019.

[45] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[46] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[47] Yang Song, Jascha Narain Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *ICLR*, 2021.

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

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

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

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

[52] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[53] 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.

[54] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

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

[56] Qinsheng Zhang, Jiaming Song, Xun Huang, Yongxin Chen, and Ming-Yu Liu. Diffcollage: Parallel generation of large content with diffusion models. *arXiv preprint arXiv:2303.17076*, 2023.

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

The appendix provides 1) the full architecture specification of correspondence attention; 2) the implementation details of the MVDiffusion system; and 3) additional experimental results in the same format as the figures in the main paper.

A Network Architecture of correspondence-aware attention block

The correspondence-aware attention block, depicted in Figure 8, comprises a transformer block and a ResNet block. The architecture of the transformer block is similar to vision transformers [11], with the inclusion of zero convolutions as suggested in ControlNet [55] and GELU [15] activation function. C, H, W are channel numbers, height and width respectively.

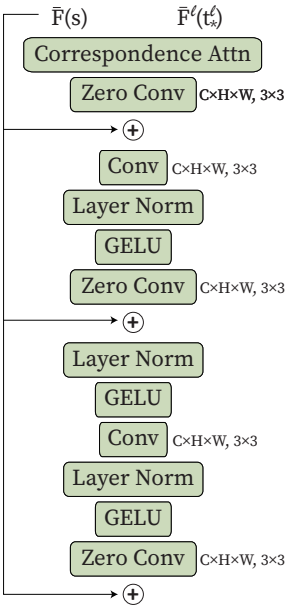

Figure 8: The architecture of the correspondence-aware attention block.

B Implementation details of MVDiffusion

B.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 [49]. In the initial phase, we train the model on perspective images with a resolution of 256×256 for 20 epochs. 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 [25] 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[51] 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 [27] 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 $1e^{-4}$.

B.2 Implementation details of baselines

We introduce implementation details of baseline in the following.

- *Text2Light* [7] We combine the prompts of each perspective image and use the released pretrained model to generate the panorama.

- *Stable Diffusion (panorama)*[38] 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* [13, 20] In our approach, we employ Stable-diffusion-v2 [49] 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 [50] 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.

B.3 Multi-view depth to image generation

Generation model. Our generation model is derived from the stable-diffusion-2-depth framework [48]. In the initial phase, we train the model on a dataset of ScanNet perspective images at a resolution of 192×256 for 50 epochs. This training process employs the AdamW optimizer [27] 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, each comprising 12 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 [43] sampler with a step size of 50 to perform parallel denoising on eight images.

Interpolation model. Our interpolation model adopts the same network architecture and weights as that of the generation model, albeit with the inclusion of extra resizing convolutional layers within each Unet block. We fine-tune these additional convolutional layers as well as the attention layers for optimal performance. The training data consists of two categories: 1) interpolation training data, derived from randomly chosen pairs of consecutive key frames and ten intermediate frames, and The original dataset that was utilized to train the generation module. During each training epoch, we maintain a careful balance by randomly sampling data from both these categories in a 3:7 ratio, where '3' represents the proportion of generation data and '7' represents the proportion of interpolation data. This is a critical step as it guarantees that the network retains its generation capabilities.

B.4 Implementation details of baselines

We introduce the implementation details of baselines in the following.

• *RePaint*[29]: In our method, we utilize depth-conditioned Stable-diffusion-v2 [49] 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 [29] 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 [50] to fill the hole. To incorporate depth information into the inpainting model, we utilize a method from a public codebase [8], which adds the feature from depth-conditioned ControlNet [55] 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.

B.5 Visualization results

Figures 9-21 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 22-27 show additional results with two baseline methods (depth-conditioned ControlNet [55] and Repaint [29]).

Figure 28 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.

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.

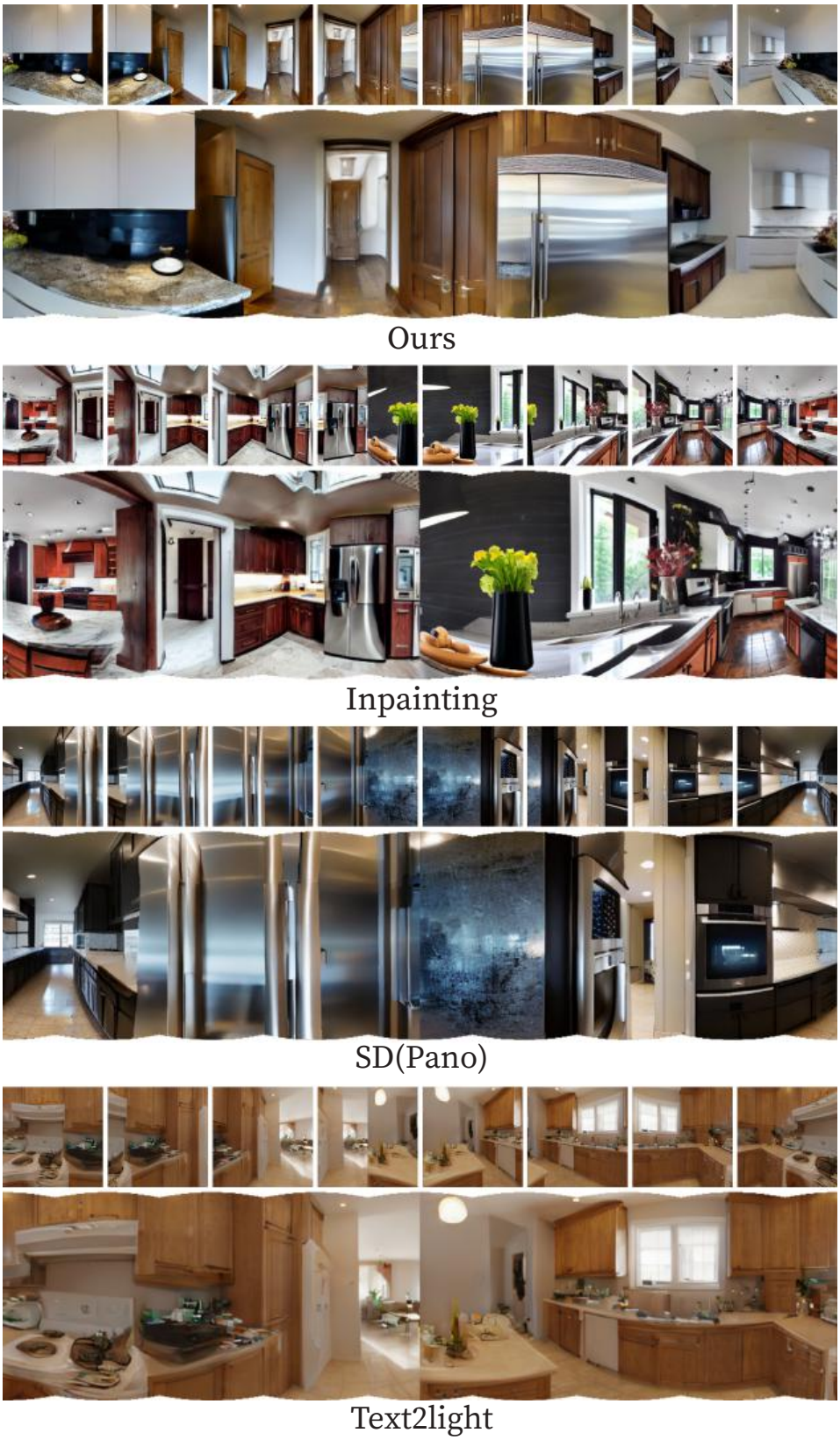

Figure 9: Addition results for panorama generation

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.

Ours

Inpainting

SD(Pano)

Text2light

Figure 10: 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.

Ours

Inpainting

SD(Pano)

Text2light

Figure 11: 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.

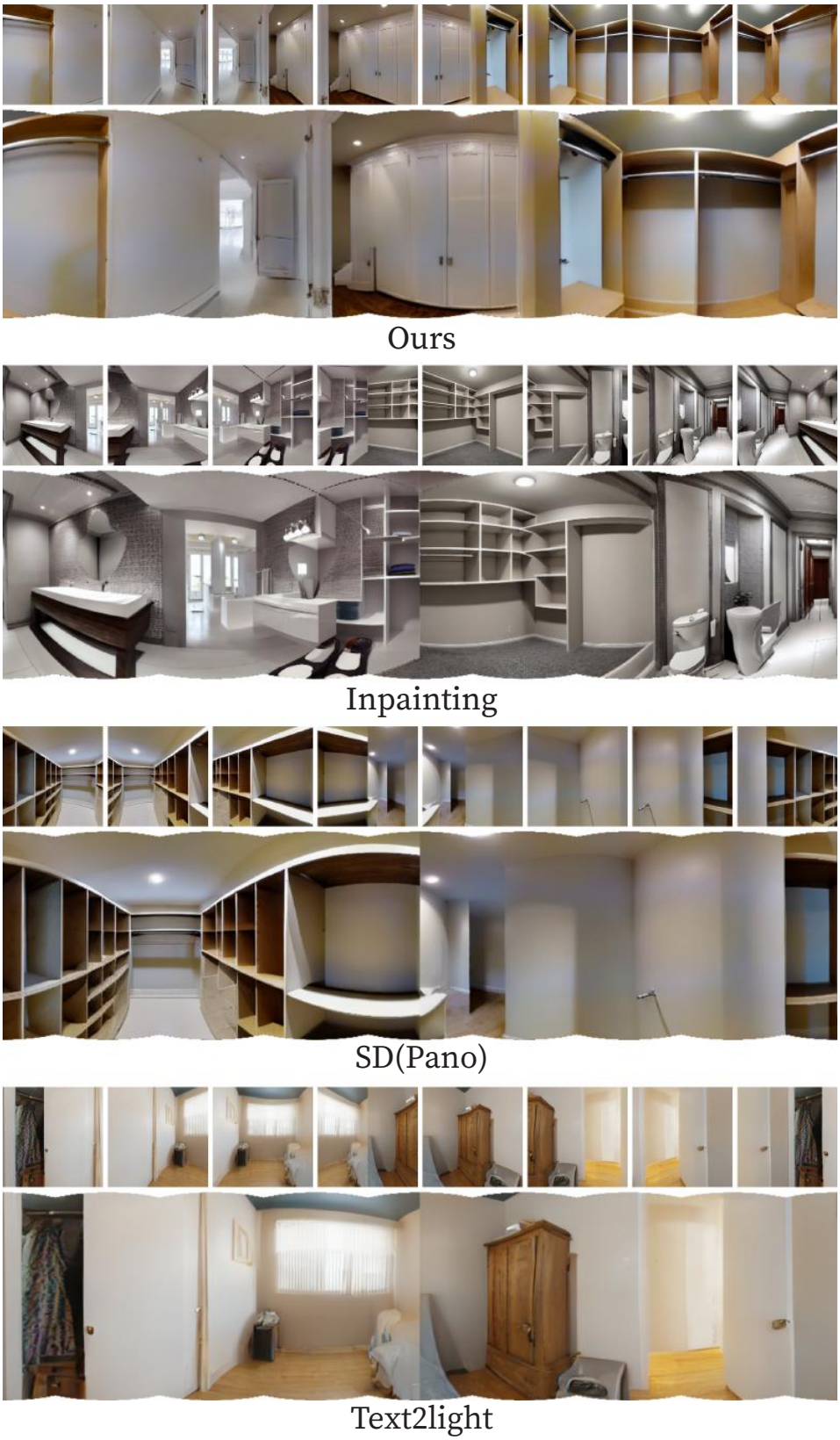

Figure 12: Addition results for panorama generation

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.

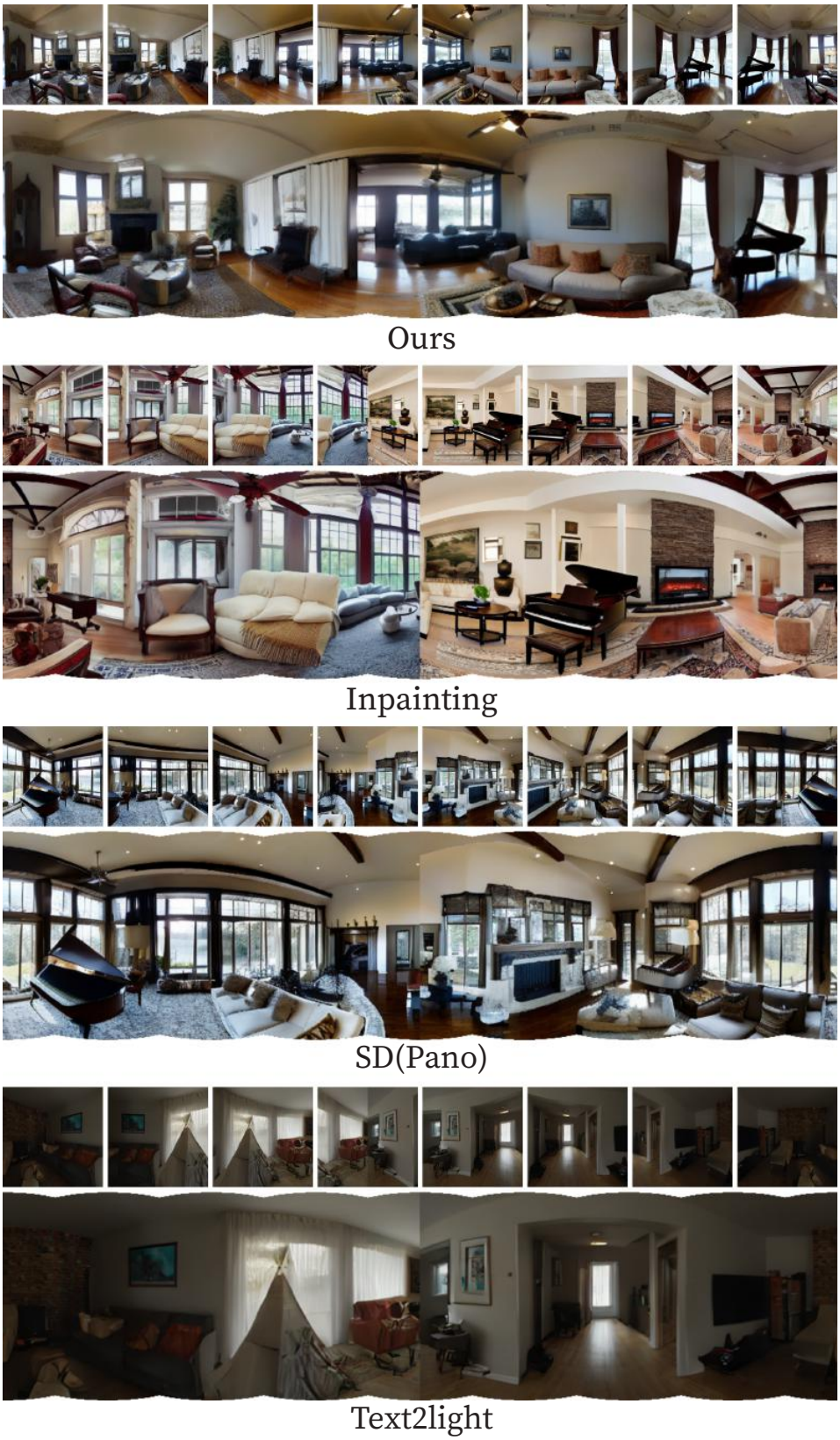

Figure 13: Addition results for panorama generation

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 14: 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.

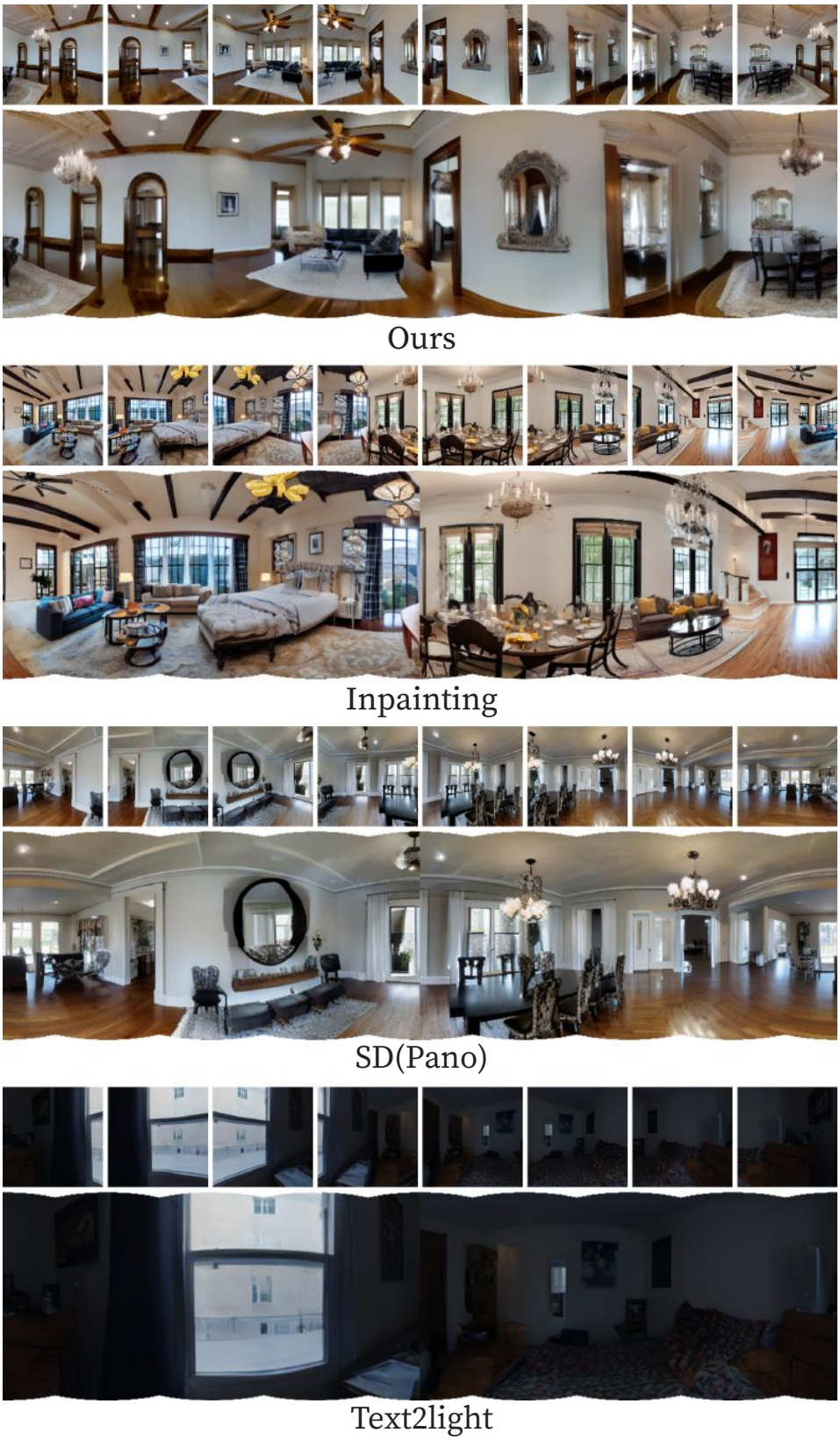

Figure 15: Addition results for panorama generation

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.

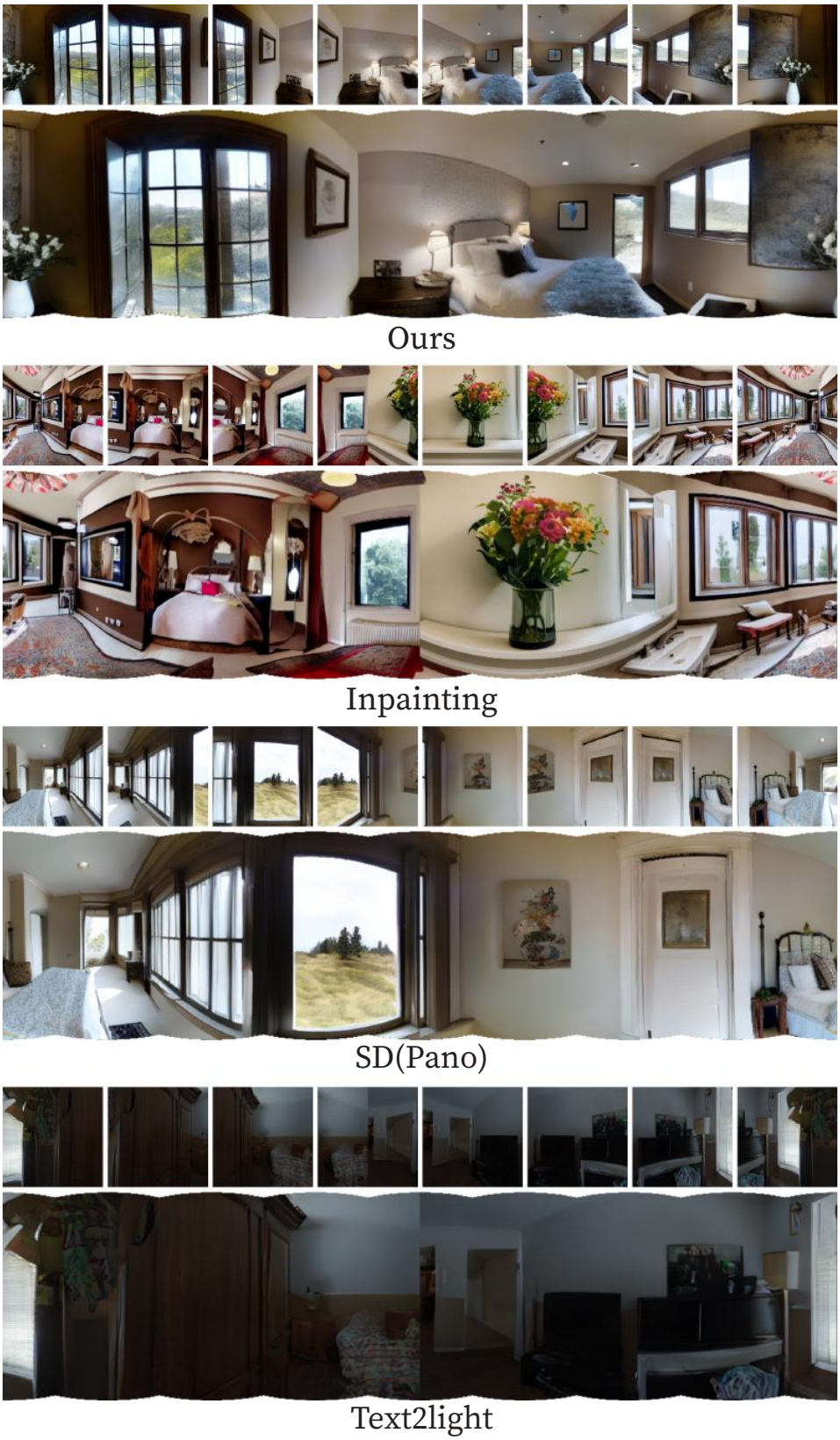

Figure 16: Addition results for panorama generation

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 17: 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.

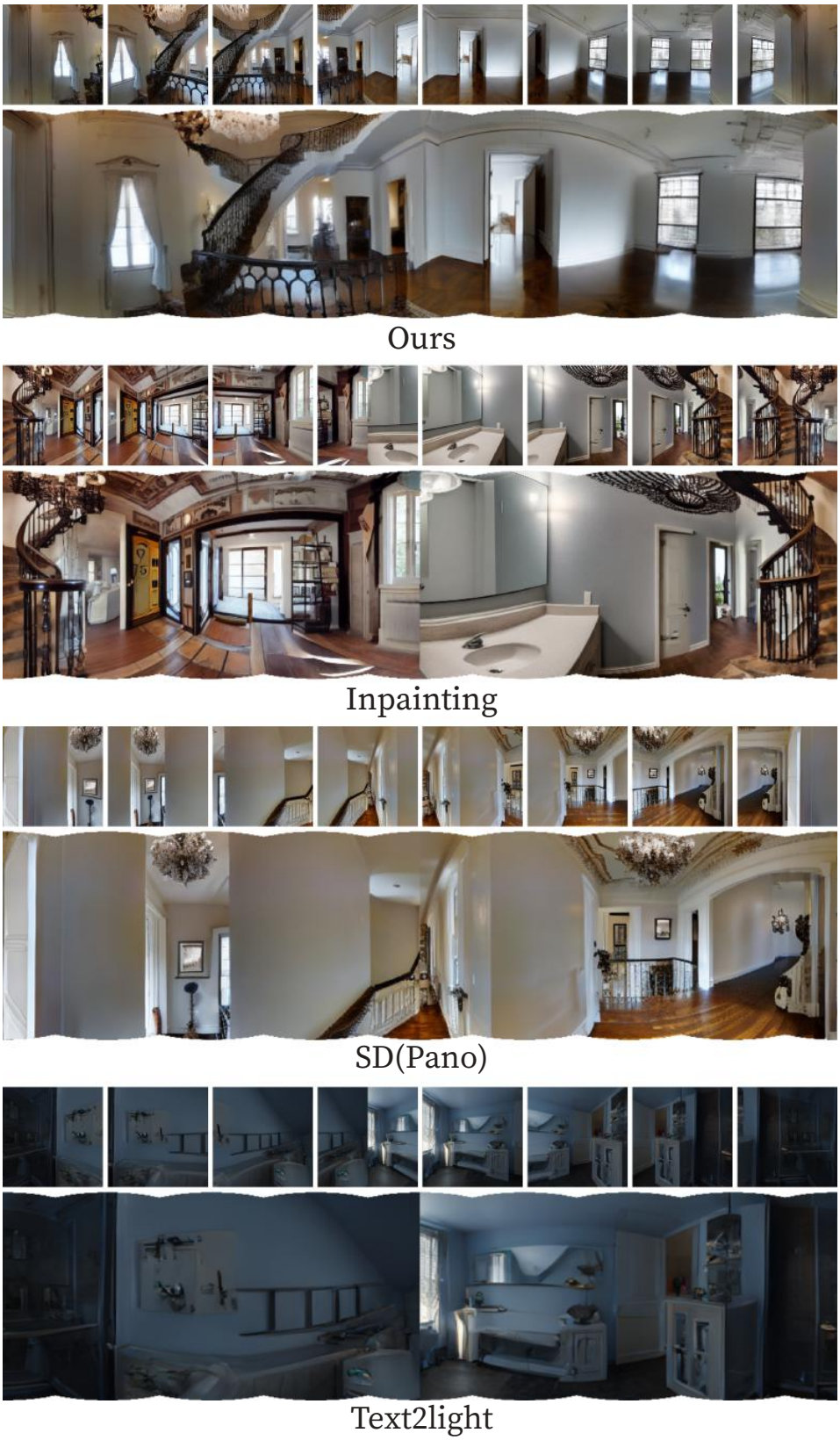

Figure 18: Addition results for panorama generation

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.

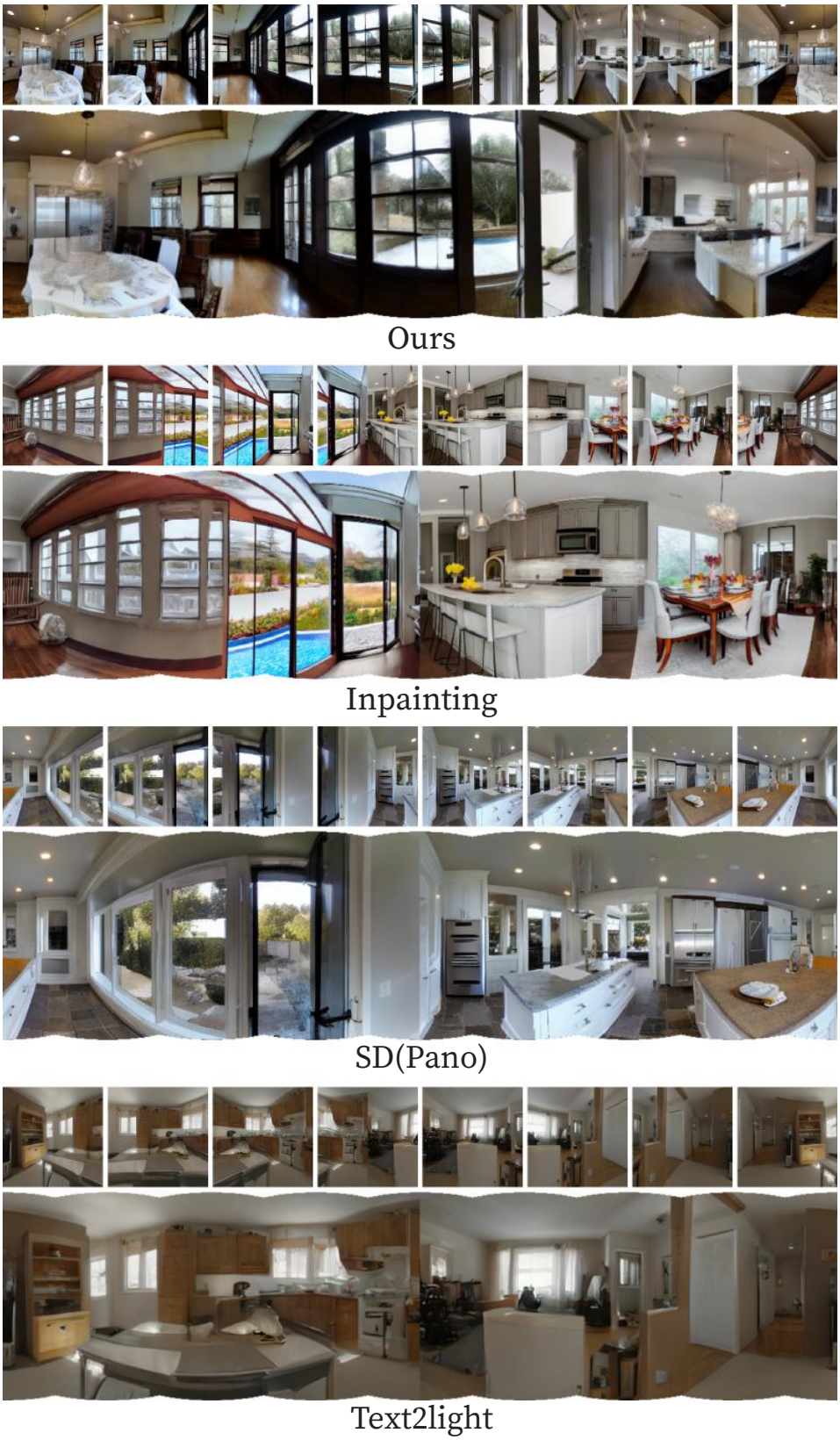

Figure 19: Addition results for panorama generation

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.

Ours

Inpainting

SD(Pano)

Text2light

Figure 20: 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.

Ours

Inpainting

SD(Pano)

Text2light

Figure 21: Addition results for panorama generation

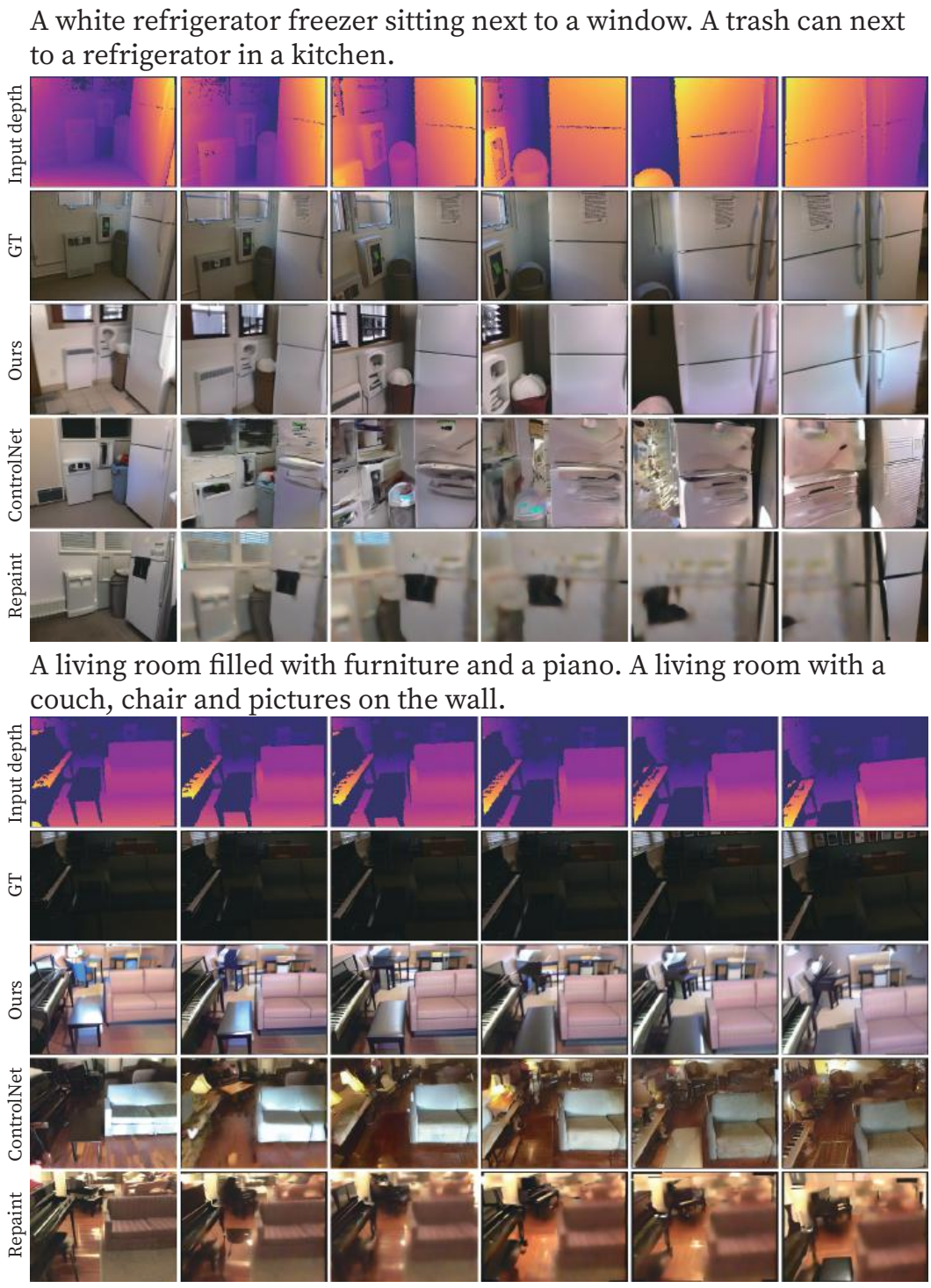

Figure 22: Addition results for depth-to-image generation.

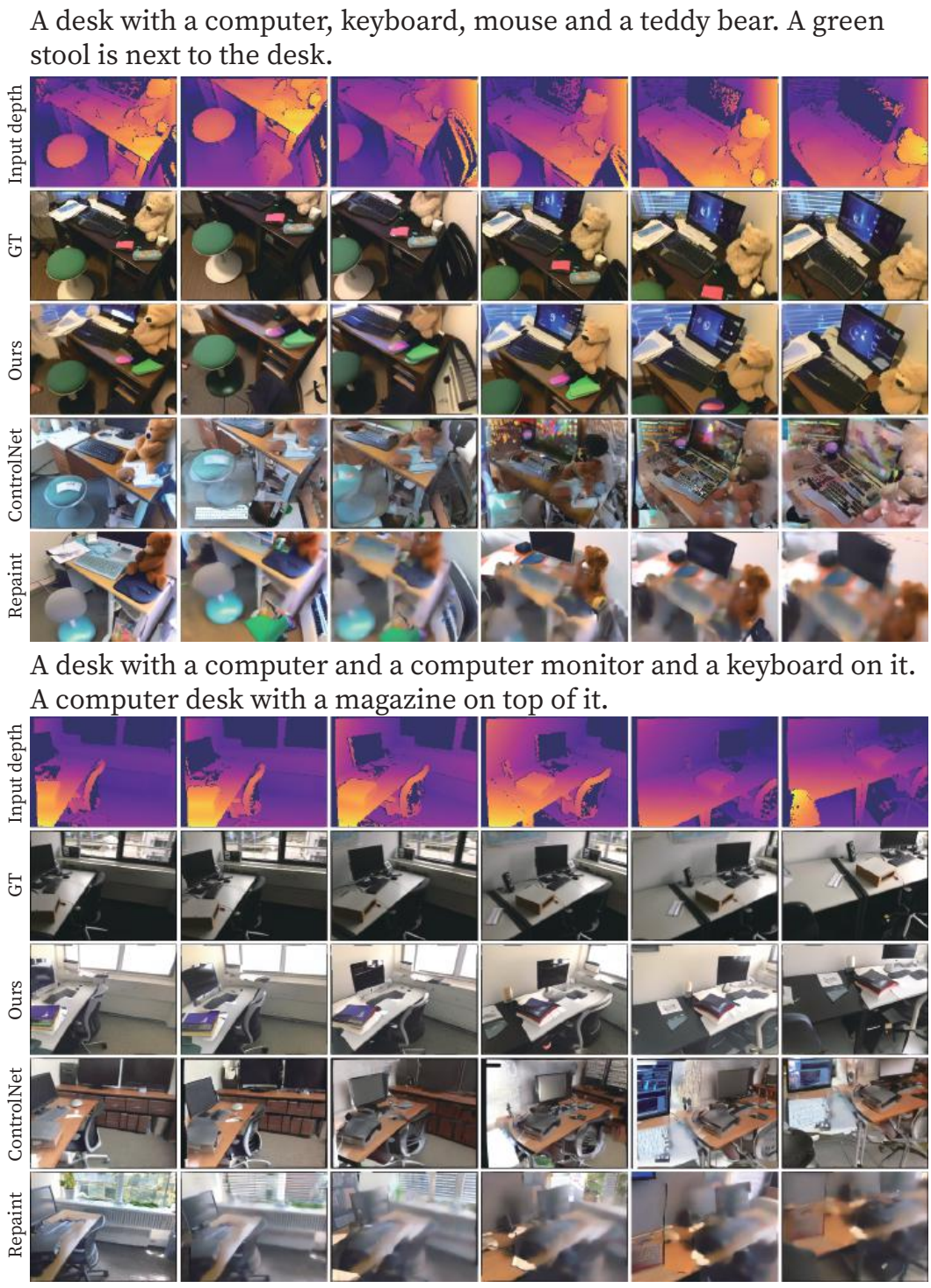

Figure 23: Addition results for depth-to-image generation.

Figure 24: Addition results for depth-to-image generation.

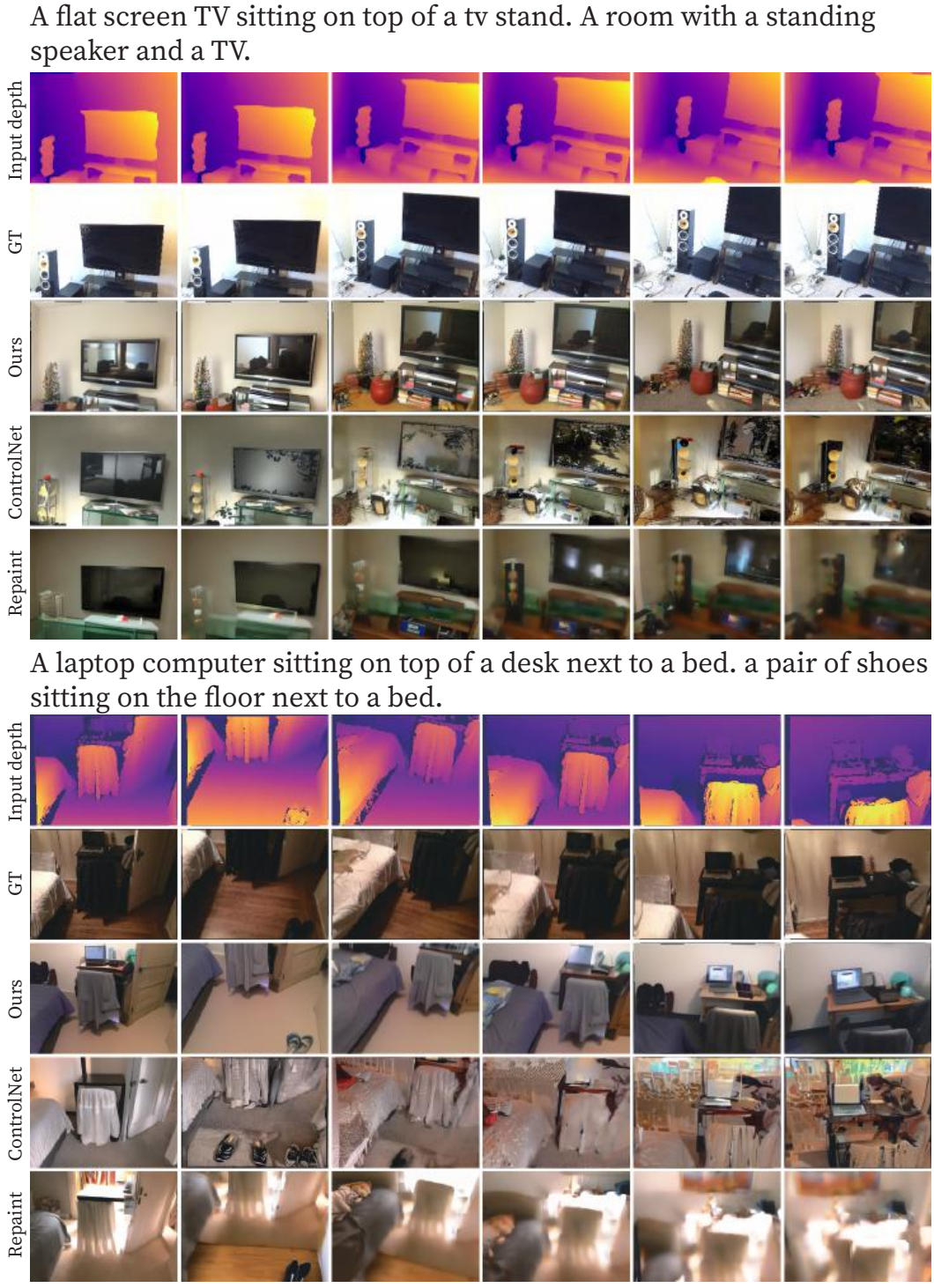

Figure 25: Addition results for depth-to-image generation.

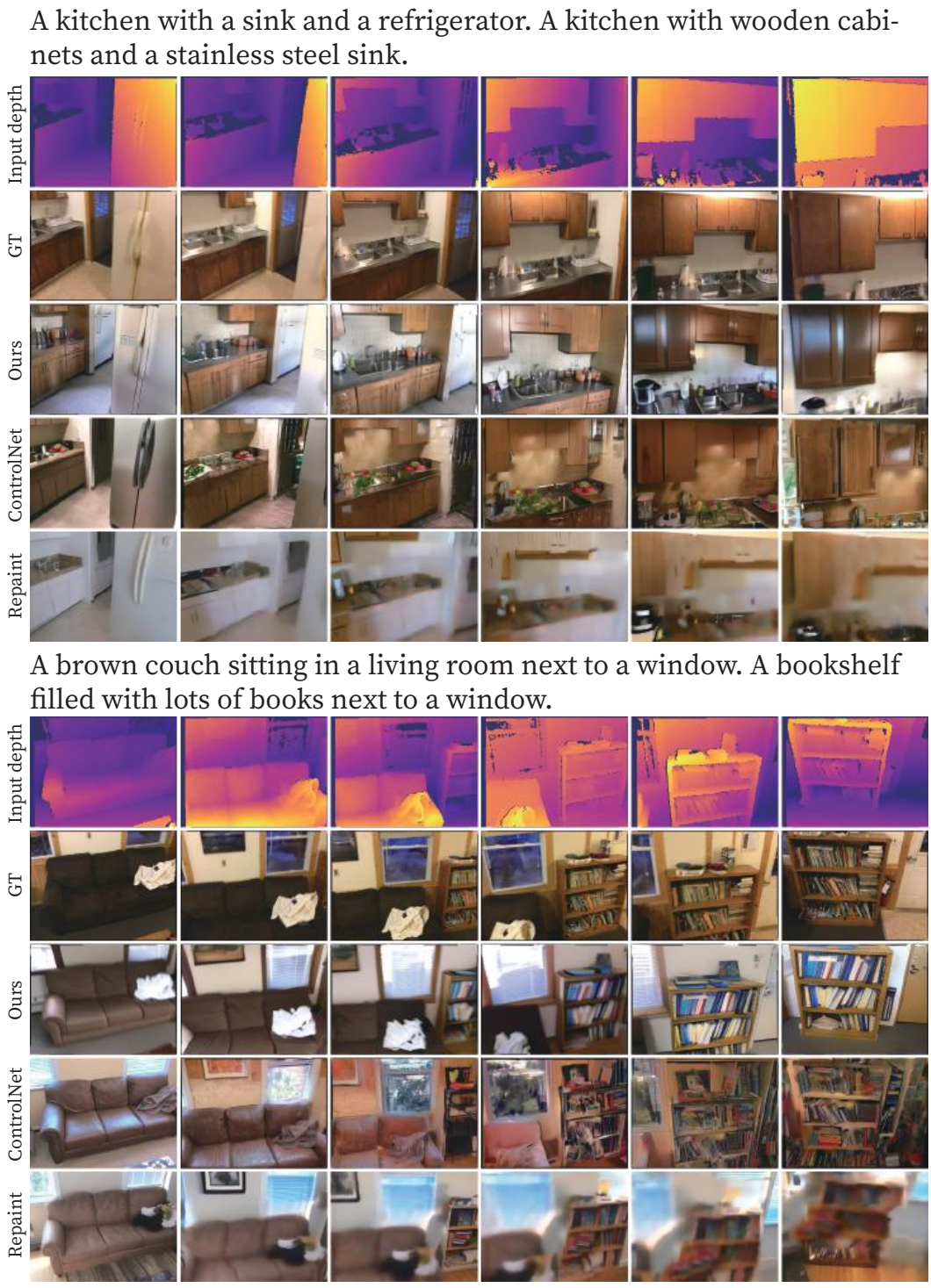

Figure 26: Addition results for depth-to-image generation.

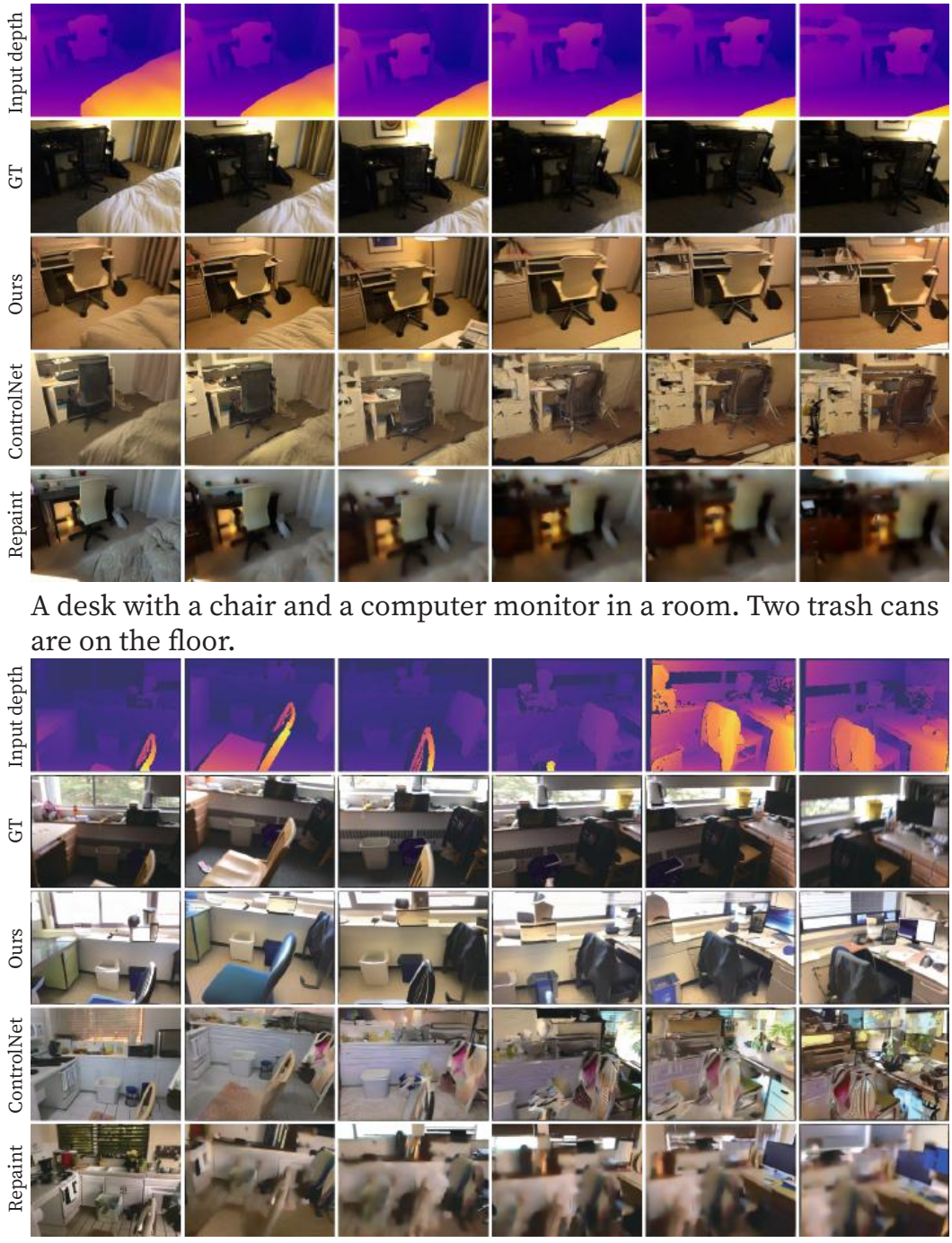

Figure 27: Addition results for depth-to-image generation.

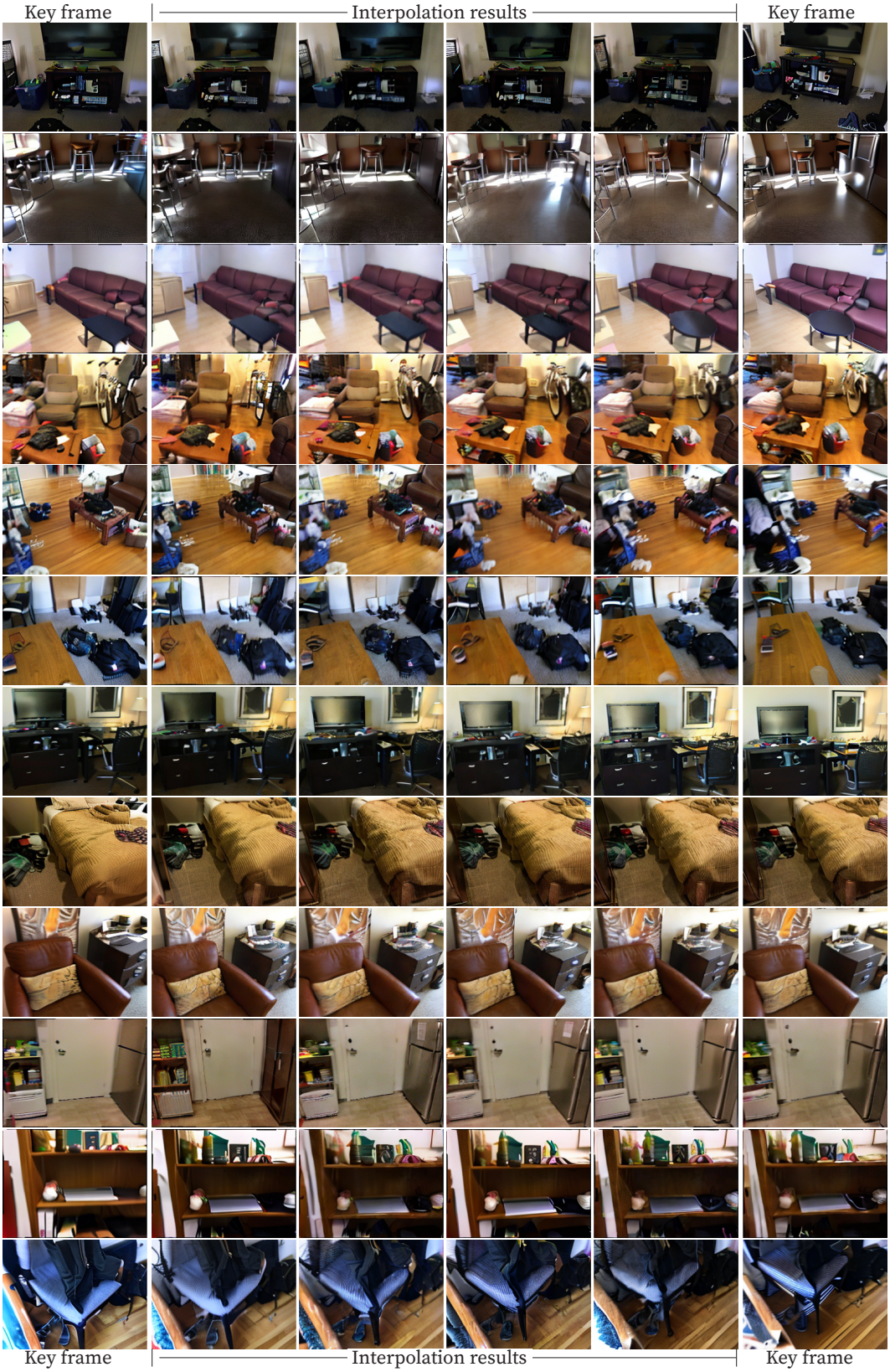

Figure 28: Addition results for interpolated frames.