Text-to-Panorama Assignment Report

Kunind Sahu
kusahu@ucsd.edu

January 28, 2025

1 Introduction

The task is to get a coherent panoramic image based on the user's input text prompt. The panorama should be aligned with the user's text prompt and also, the panorama's seams need to be tiled. That is, it's ends should meet. Deliverables:

- Get a coherent 360 panorama as an output from text-to-image SD model for 5 prompts
- Leverage the prior answer to get a panorama generated using both, the user's text prompt and a given depth map conditioning.

The code can be found here: `https://github.com/kunind27/Panorama-Generation.git`

2 Methodology

I used a LoRA checkpoint to enable a pretrained text-to-image Stable Diffusion-XL [1] on Diffusers [2] to generate panoramas and did some additional processing on the generated panorama candidate to correct the seams (as explained below).

2.1 Generate Initial Panorama

I used a LoRA checkpoint to enable a pretrained SDXL model to generate an initial panorama candidate. The prompt used was "`Tron World`". To enhance the quality of the panorama, I used Textual Inversion [3] using embeddings that were available with the LoRA checkpoint. The panorama can also concurrently be conditioned on a depth map using ControlNet [4]. The generated panorama (Fig.. 1) might have seams that are not tiled. The panorama resolution is 1024×512

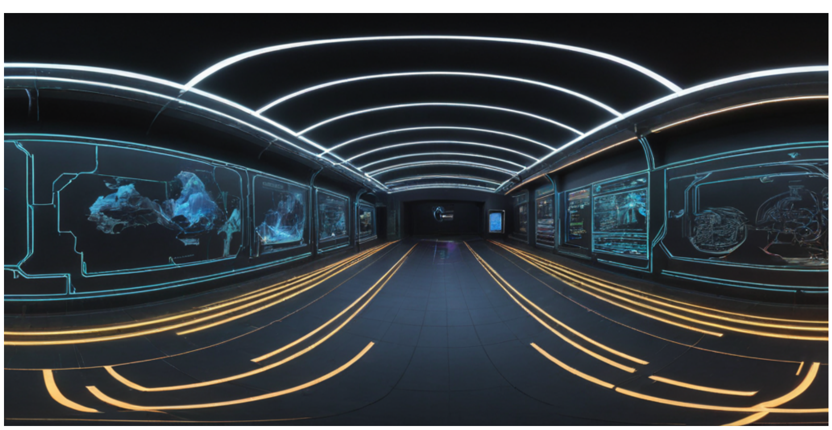


Figure 1: Initial Panorama generated with Prompt: "`Tron World`"

2.2 Upscale the Image (Optional)

You can optionally upscale the image to **double** its resolution to 2048×1024. The reason will be made clear in section 4.2.

2.3 Swap Halves

We swap the left and right halves of the image to check if the seam is tiled. As we can see in Fig. 2, there iss a slight mismatch which makes the panorama incoherent.

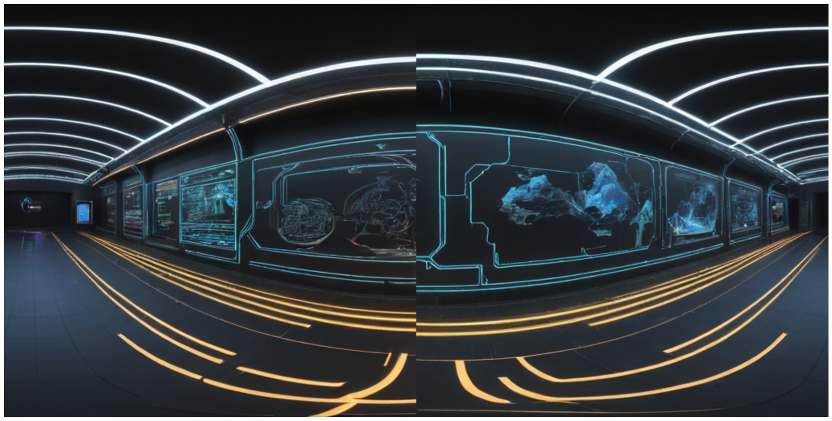


Figure 2: Swapped Left and Right Halves of Panorama shows an untiled seam

2.4 Crop Square from Swapped Image

We take the image with swapped halves and crop a square (Fig. 3)

Figure 3: Crop a square centred around the middle of the image

2.5 Seam Tiling via Image Inpainting

We take the cropped (square) image (described above) and inpaint the central region (the cause of incoherent seams) using an SD-XL inpainting model (Fig. 4). A mask is created with the square size regions denote the region to be inpainted).

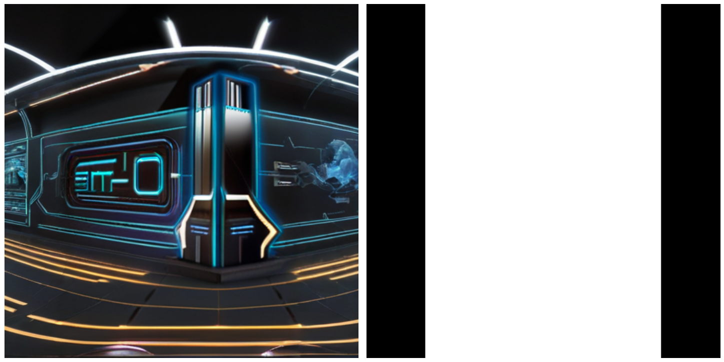


Figure 4: Inpainted Square (left) and Inpaint Mask (right)

2.6 Paste Result back to Image

We take the inpainted image (which is basically the seam-tiling operation) and then paste it back into the panorama with swapped halves (Fig. 5. It can be seen that the seems are tiled, and thus the panorama will be coherent.

Figure 5: Pasting the Inpainted region back to the swapped image

2.7 Generate Final Panorama

Generate the final panorama (Fig. 6) by swapping the halves back to their original place.

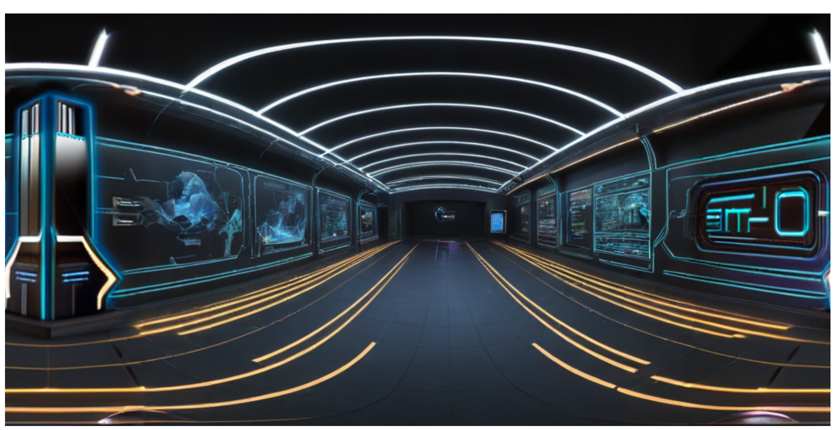


Figure 6: Swapping halves back to get the final result

3 Results

3.1 Text-to-Panorama

We generated results for five prompts. The generated results and the prompts are mentioned below

- "Tron World": Fig. 9
- "A Futuristic Room": Fig. 10
- "Modern Art Museum": Fig. 11
- "Science Lab": Fig. 12
- "Sci-fi Cryo Pods": Fig. 13

3.2 Text-to-Panorama with Depth Control

We generated results for the same prompts as before, but with conditional depth control. The Depth Map is shown in Fig. 7. The generated results and the prompts are mentioned below:

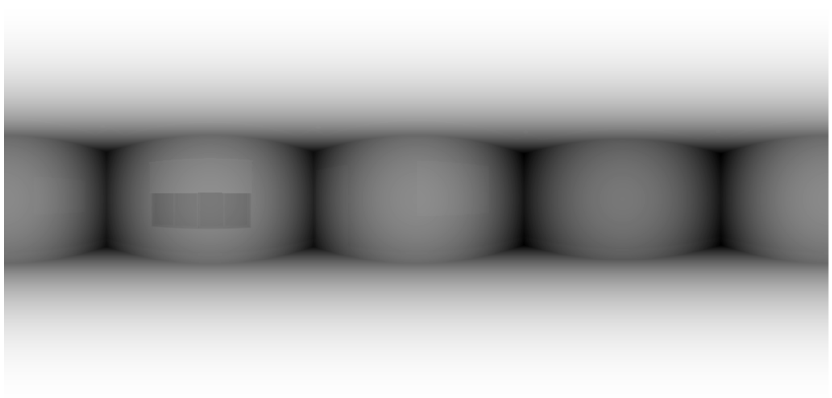


Figure 7: The given depth map for conditional panorama generation

- "Tron World": Fig. 14
- "A Futuristic Room": Fig. 15
- "Modern Art Museum": Fig. 16
- "Science Lab": Fig. 17
- "Sci-fi Cryo Pods": Fig. 18

4 Motivations for the Workflow

4.1 Image Inpainting for Seam Tiling

As seen in sections 2.3 and 2.4, the initial panorama has an untiled seam.

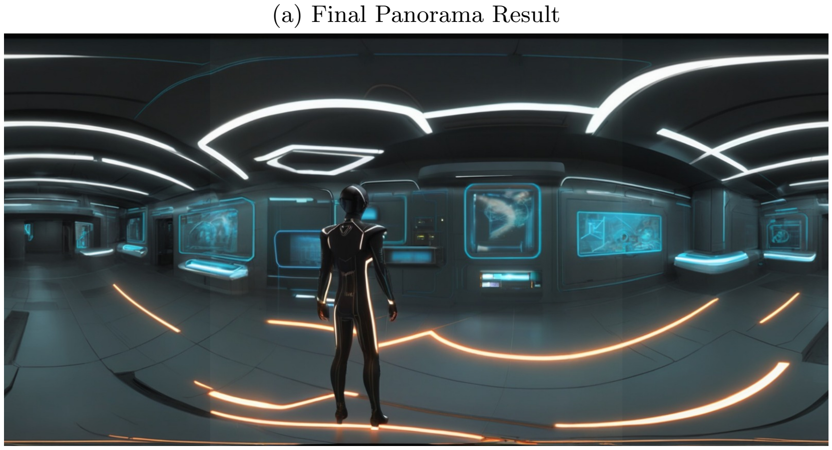
- I initially employed a Multi-Band Blending algorithm to blend the two seams, but it did not work too well.
- The reason for this is because there is no feature matching between the seams. That is, even though they are aligned correctly, there is no feature registration possible between the two parts of the seam.
- This leads to me turning to image in-painting, which will ensure that the generated in-painted feature is contiguous and coherent.

4.2 Upsampling the Panorama

- The workflow works fine without image upsampling, but I noticed that it required some tuning to the image in-painting model hyperparameters (such as guidance-scale, strength etc), and text prompts to get an in-paint that seemed consistent with the overall scene.
- Upsampling the initial panorama to twice the inital resolution and then performing in-painting improved the coherence of the in-painting results with the remaining scene.
- But, there was a drawback, upscaling the image and then in-painting it led to a filter-like artefact on the in-painted region which seemed to demarcate the in-painted region from the original image (Fig. 8)

So, in-conclusion, there was a trade-off between the variance and consistency of the in-painting result and a filter-like artefact on the in-painted region that demarcates it within the panorama.

(a) Final Panorama Result



(b) Swapped Halves to the in-painted seam

Figure 8: Filter-like artefact on the in-painted region when upscaling

5 Methods of Improvement

5.1 Image Inpainting

- A wider, and more systematic hyperparameter search to get hyperparameters that work well for the specifc use-cases for this workflow.
- Prompt Engineering to find out the ideal prompts that improve coherence with the surroundings. Currently I include the original user prompt to generate the panorama. We could instead use the original prompt and the square seam image to generate a newer prompt that will be more effective to inpaint the regions to maintain the seams using an LLM

- Using different checkpoints for different situations. In my experiments, I empirically realized that the checkpoint '`stabilityai/stable-diffusion-2-inpainting`' works better than '`diffusers/stable-diffusion-xl-1.0-inpainting-0.1`' for panorama generation without depth conditioning because the latter checkpoint is prone to generating images that are more animated and non-realistic, this leads to it being unable to in-paint seams that are coherent with the rest of the image. This has also been documented in this reddit thread.

- I cut out a square region to in-paint because it was shown to work well (empirically) for the specific Stable-Diffusion model I was using. Instead of a square cutout, we could try using the entire image to give the in-painting model more context about the entire scene, thus improving the inpainting results

- When we in-paint seams for a depth-controlled panorama, the in-paint does not have information about the depth-map constraints, which leads to the result not being fully aligned with the depth map. An approach could try to also condition the in-painting process on the depth map for better coherence.

5.2 Removing Filter Artefact when Upscaling

- When upscaling and in-painting, a filter-like artefact is introduced in the in-painted region which enables users to ascertain the in-painted region from the image.

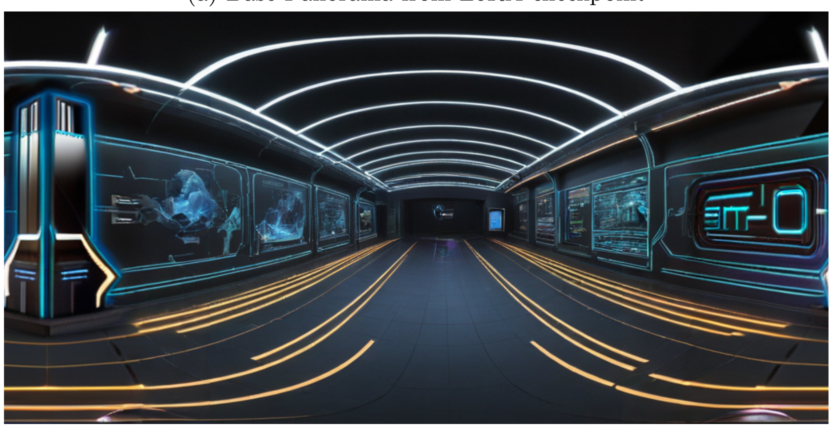
- To rectify this, we can implement a Poisson Image Editing [5] based post-processing to correct the filter-like artefact on the in-painted region.

- I tried using Poisson Image Editing which alleviated the problem to a certain extent.

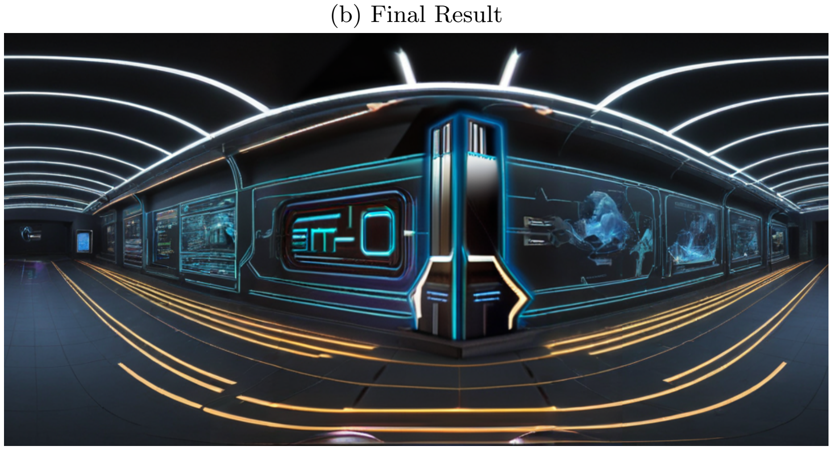
6 Figure of Generated Panoramas

6.1 Text to Panorama

(a) Base Panorama from LoRA checkpoint



(b) Final Result

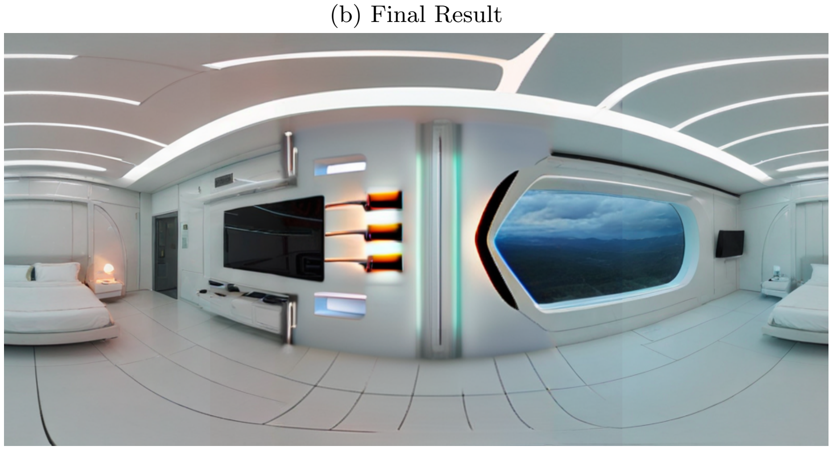


(c) Swapped Halves to see there is NO Seam

Figure 9: Results for "Tron World"

(a) Base Panorama from LoRA checkpoint

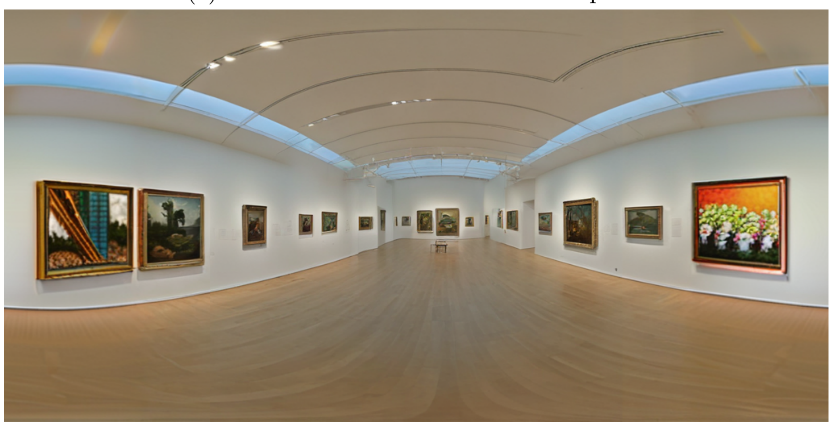
(b) Final Result



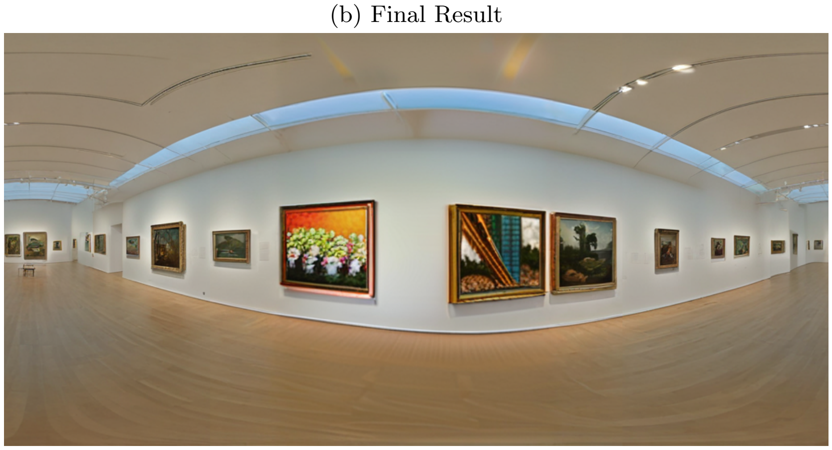
(c) Swapped Halves to see there is NO Seam

Figure 10: Results for “A Futuristic Room”

(a) Base Panorama from LoRA checkpoint



(b) Final Result

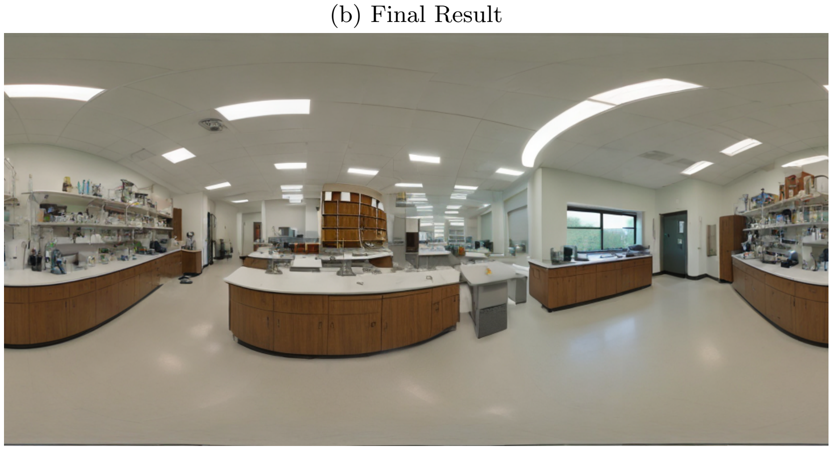


(c) Swapped Halves to see there is NO Seam

Figure 11: Results for "Modern Art Museum"

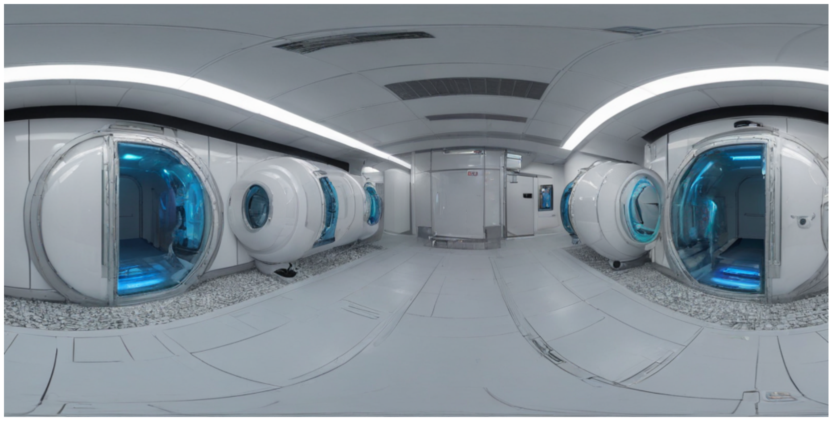
(a) Base Panorama from LoRA checkpoint

(b) Final Result



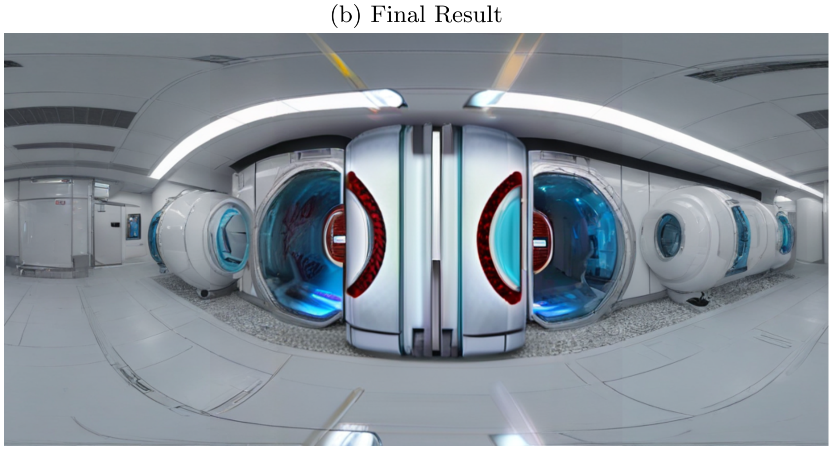
(c) Swapped Halves to see there is NO Seam

Figure 12: Results for “Science Lab”



(a) Base Panorama from LoRA checkpoint

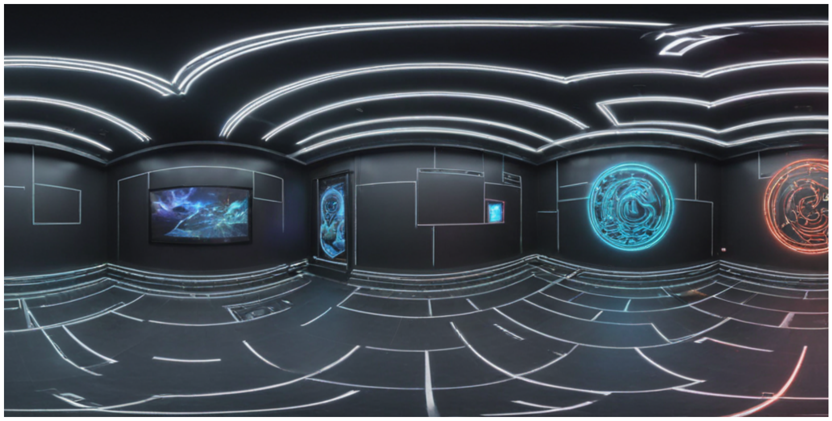
(b) Final Result



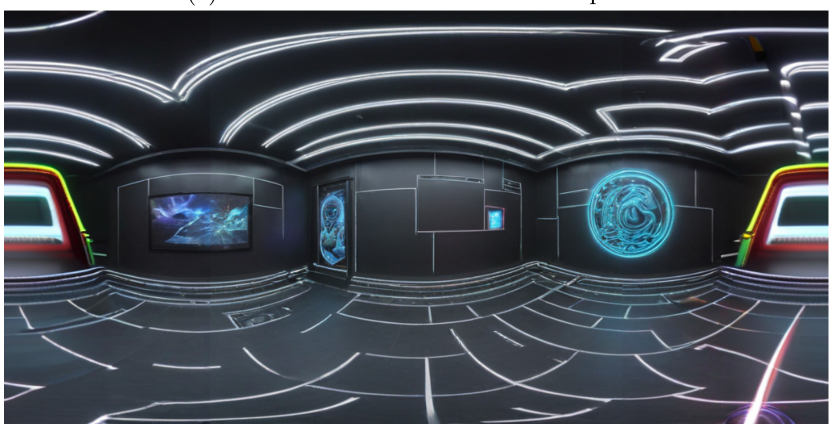
(c) Swapped Halves to see there is NO Seam

Figure 13: Results for "Sci-fi Cryo Pods"

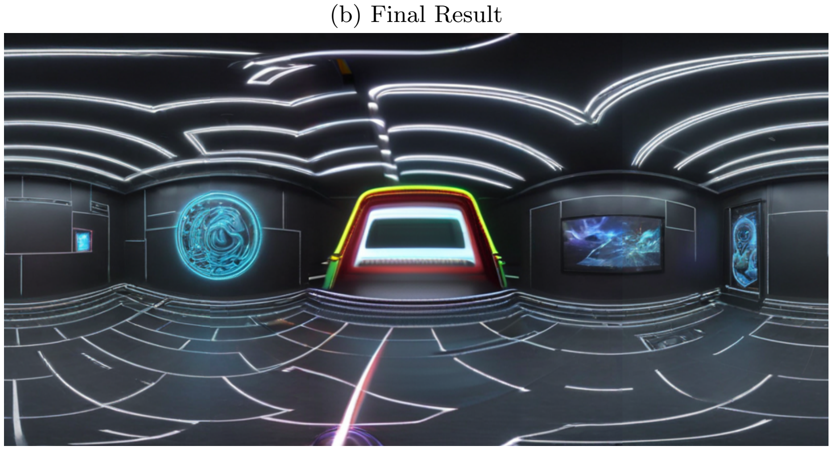
6.2 Depth Controlled Text to Panorama



(a) Base Panorama from LoRA checkpoint

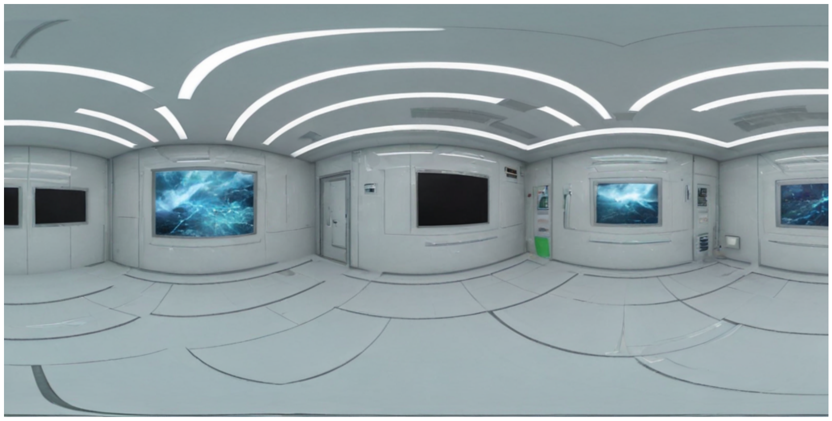


(b) Final Result

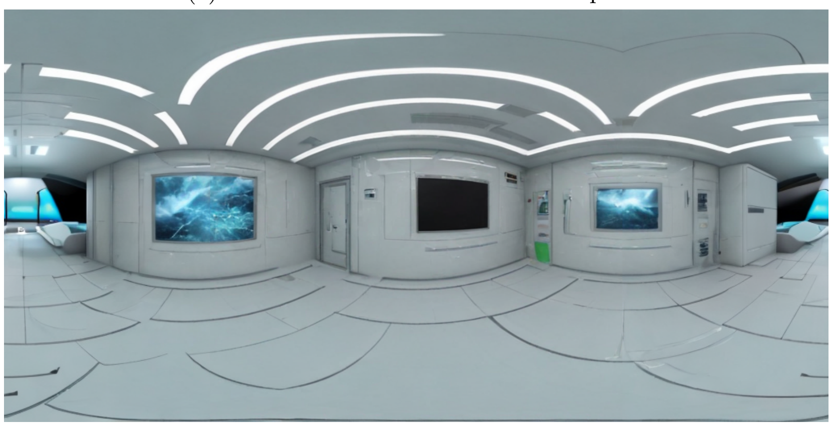


(c) Swapped Halves to see there is NO Seam

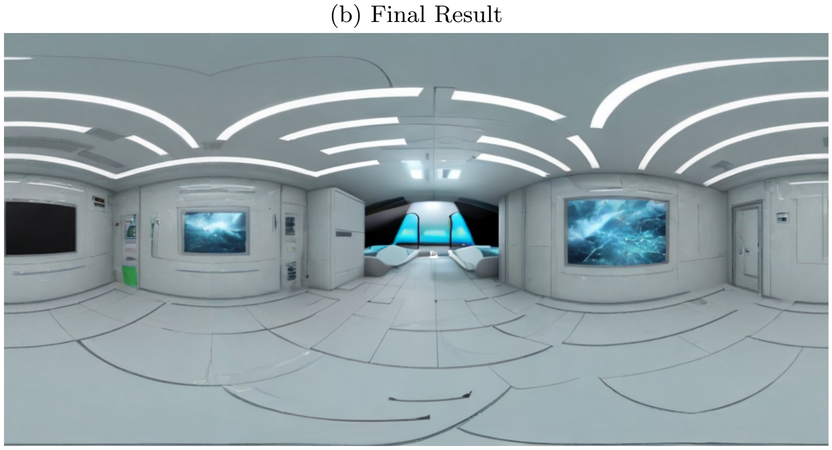
Figure 14: Results for “Tron World”



(a) Base Panorama from LoRA checkpoint



(b) Final Result

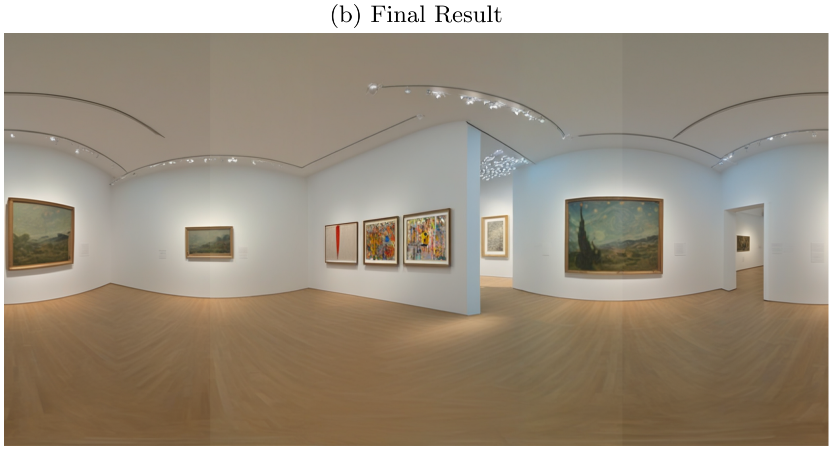


(c) Swapped Halves to see there is NO Seam

Figure 15: Results for “A Futuristic Room”

(a) Base Panorama from LoRA checkpoint

(b) Final Result

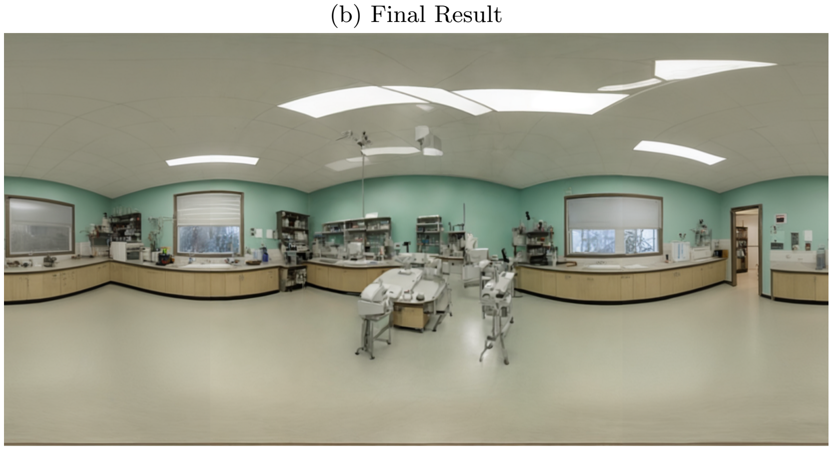


(c) Swapped Halves to see there is NO Seam

Figure 16: Results for “Modern Art Museum”

(a) Base Panorama from LoRA checkpoint

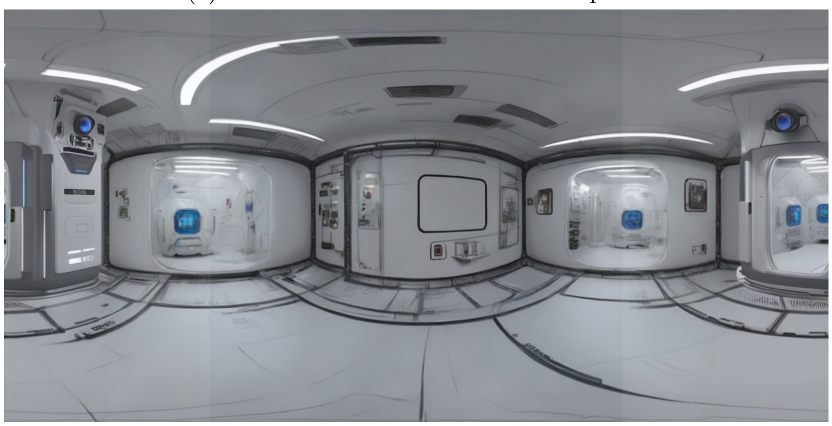
(b) Final Result



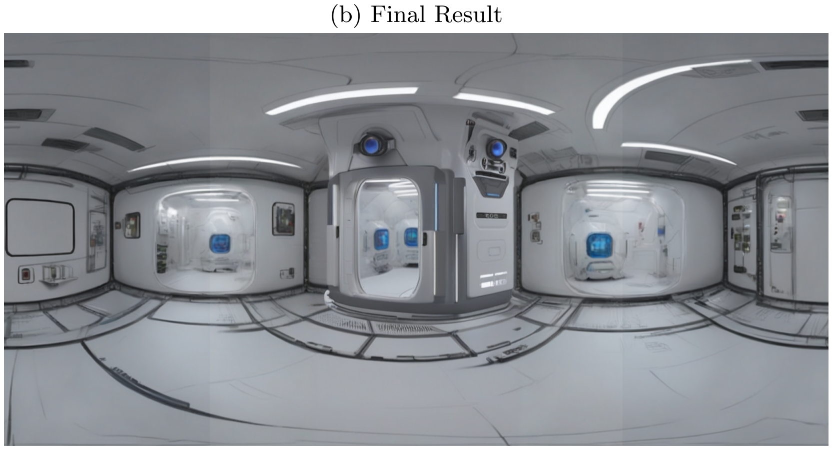
(c) Swapped Halves to see there is NO Seam

Figure 17: Results for “Science Lab”

(a) Base Panorama from LoRA checkpoint



(b) Final Result



(c) Swapped Halves to see there is NO Seam

Figure 18: Results for “Sci-fi Cryo Pods”

References

[1] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, “Sdxl: Improving latent diffusion models for high-resolution image synthesis,” 2023.

[2] P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, D. Nair, S. Paul, W. Berman, Y. Xu, S. Liu, and T. Wolf, “Diffusers: State-of-the-art diffusion models.” https://github.com/huggingface/diffusers, 2022.

[3] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, “An image is worth one word: Personalizing text-to-image generation using textual inversion,” 2022.

[4] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models."

[5] J. Martino, G. Facciolo, and E. Llopis, "Poisson image editing," *Image Processing On Line*, vol. 5, pp. 300–325, 11 2016.