Depth-Controlled-Spherical-Panorama-Generation

Niraj Mahajan

December 21, 2023

1 Introduction

This project demonstrates the generation of text-controlled 360-degree panoramas using Stable Diffusion. I have implemented seam-tiling to achieve coherent panoramas and also experimented with depth conditioning using a ControlNet. In this study, I compare the results of panoramas with and without depth conditioning and seam-tiling.

(a) A non coherent panorama with inconsistent seams

(b) A coherent panorama: The right and left ends match seamlessly.

Figure 1: Comparison of panoramas with and without seam-tiling

2 Requirements

To set up this project, run the following commands:

```
$ pip install diffusers transformers accelerate safetensors huggingface_hub
$ git clone https://github.com/replicate/cog-sdxl cog_sdxl
```

2.1 Approach

In this section, I describe the approach that I used to solve both tasks. I describe the method to obtain a 360 panorama, followed by performing seam correction and depth conditioning.

2.2 Generating a Prompt Conditioned Panorama

For generating 360-degree panoramas, I utilized the LoRA weights from this model. This is an XL Stable diffusion model that is tuned to generate 360 panorama images when referred to the T0k token.

2.3 Seam Correction

Initially, the seams of these panoramas were not coherent. To address this, I rolled the panorama by half its width, placing the incoherent transition in the centre. Then, by masking and inpainting, the center using XL Stable Diffusion, a coherent panorama was achieved.

(a) Rolled Image (b) Inpainted Image

Figure 2: Fig(a): An image generated by rolling over the pixels by half the width. The centre has an inconsistent transition.
Fig(b): The coherent panorama generated by masking and inpainting the central region.

2.4 Generating a Prompt and Depth Conditioned Panorama

For depth-conditioned panoramas, I used the ControlNet pipeline. Similar to the previous method, I corrected the non-coherent seams through rolling and inpainting.

3 Results

I have tested this methodology with the following prompts:

1. 'A dining room in Persia'
2. 'Inside Bag End'
3. 'Inside the cantina in Star Wars'
4. 'Inside a bedroom in France'
5. 'Inside a medieval prison'

Below are the results for each prompt:

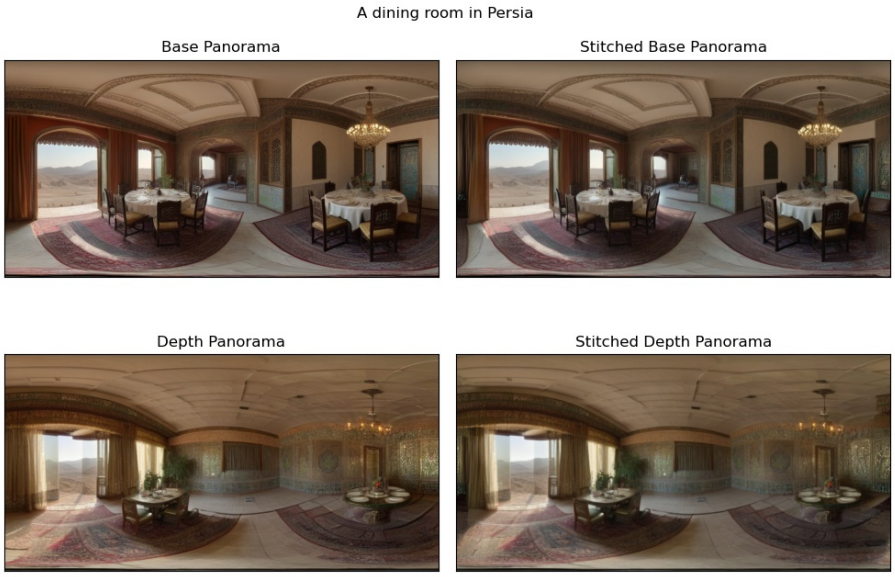

Figure 3: Dining Room in Persia

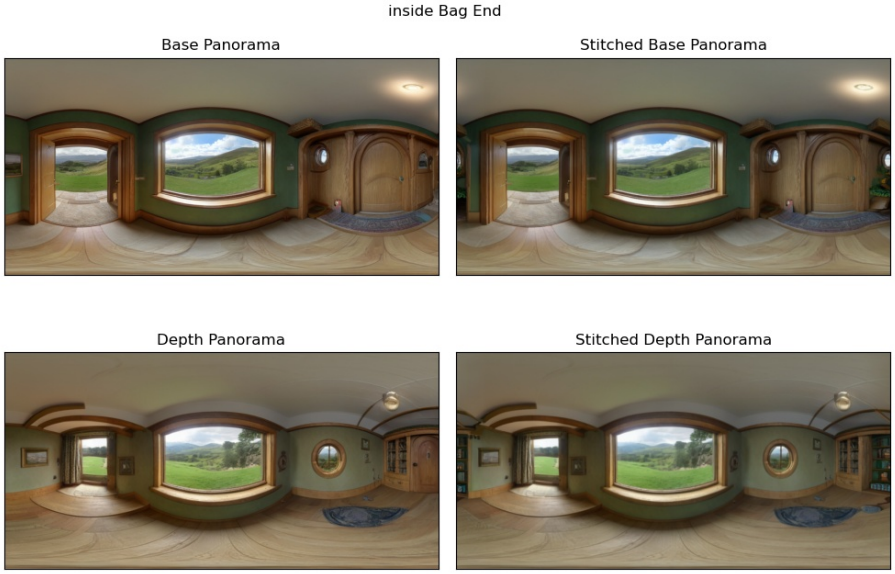

Figure 4: Inside Bag End

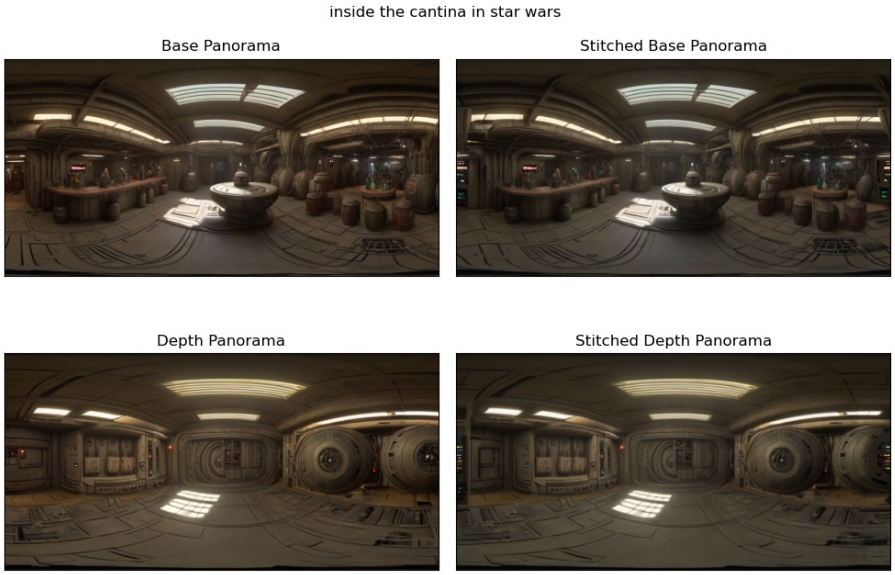

Figure 5: Inside the Cantina in Star Wars

Figure 6: Inside a bedroom in France

Figure 7: Inside a medieval prison

4 Usage of Code

To run the code, execute the following command:

`python3 main.py`

`main.py` contains the prompts, which can be edited as needed.

- `part1.py`: Contains `create_pano` function to generate panoramas using a prompt.
- `part2.py`: Contains `create_pano_depth` function for generating panoramas using prompt and depth information.
- `cleaner.py`: Contains `clean_pano` function for seam-stitching.

5 Rendering Panoramas

The generated images are stored in the `images/` directory. Use images from `images/raw/` to render panoramas at 360 Panorama Web Viewer.

The following are the (**prompt only**) coherent panoramas that can be rendered in the Web viewer:

1. Dining Room in Persia
2. Inside Bag End
3. Inside the Cantina Star Wars
4. Inside a bedroom in France
5. Inside a medieval prison

The following are the **depth-conditioned panoramas**:

1. Dining Room in Persia
2. Inside Bag End
3. Inside the Cantina Star Wars
4. Inside a bedroom in France
5. Inside a medieval prison