Generative Panning with Outpainting

Kayla Akyüz
kaylaakyuz@gmail.com
kaylaakyuz@hacettepe.edu.tr
Hacettepe University

Figure 1: **An Image of a School Building**

ABSTRACT

As a term project for BBM444 course at Hacettepe University, I will be exploring and researching the concept of generative panning with the usage of outpainting. In this document you will find my progress report which includes tests conducted on Stable Diffusion models with various parameters to achieve 90 degree rotation of panning.

KEYWORDS

outpainting, neural networks, stable diffusion, camera panning

1 PROBLEM DESCRIPTION

In the recent era, generative AI has taken a special place among us slowly filling our life with its various use cases. Some of these rather remarkable use cases invites us to wonder a future where AI might surprise us on the unforeseen use cases that it accomplishes.

One particularly intriguing use case involves the generation of visual worlds by morphing images and constructing entire environments one frame at a time. In essence, an AI can comprehensively generate a graphical world, understanding the current context and seamlessly extending the frame of an image within reasonable definition.

This implies instead of developing computer graphics side for a game, we can define certain context, then let the AI generate whatever images per second it does, and if it is feasible for our satisfaction we can play it. This differs from simply stitching continuous images together to form a video because it also accounts for user interaction, particularly the movement and looking-around actions performed by the user. This is just one of the many interactions users can provide, and if AI can accommodate the user's ability to move freely, we could potentially generate a wide range of game genres and 3D software solely by generating images.

One of the fundamental questions we need to address is: what happens when the user looks around or moves around, in the simplest terms? We know the user's camera pans or rotates, or when they move in a certain direction, the view either extends to certain direction or for the moving forwards or backwards cases it kind of creates a cascading effect of zooming in and out! Understanding this basic concept, which resembles fractal imagery, where zooming in and out uncovers more detail, we can simulate the morphing of all possible camera movements onto an image. The resulting image will have missing parts, which can then be generated by AI. If the AI understands the context, it can create a world with which we can interact and explore. Furthermore, if the AI is sophisticated enough, it could even simulate aspects of physics such as gravity or time on these images that are morphed on a frame-per-second basis.

This truly remarkable concept of being able to generate reality solely through 2D images prompts us to question what it means to perceive and whether there is a distinction between a well-generated 2D world and a 3D one. To delve into this, we can examine 360-degree scenes or virtual reality (VR) and augmented reality (AR) technologies, observing how our minds effortlessly create the illusion of a 3D world through a 2D lens.

Indeed, while the question may evolve into a philosophical one, as computer engineers, our primary concern is to determine if such a feat is technologically feasible. Firstly, can AI generate contextually appropriate images that simulate camera movements? To approach this question in a more researchable manner, I have simplified it to: **"What are the effects of outpainting on perspective-warped images, and can this method achieve camera panning?"** This problem definition will serve as my research topic, and I will endeavor to provide an answer using the current technology available to me.

2 RELATED WORK

There are numerous related works addressing the broader question of this idea, including specialized works focusing on camera movements and panning. Additionally, related research provides

valuable tools to understand and implement these technologies effectively.

Firstly, we can examine the Corridor Crawler Outpainting [1], which is essentially testing out a very specific context of camera movement: zooming out in corridors. The results appear satisfactory, and when played in reverse, they create the illusion of a corridor, reminiscent of early DOS-era games.

Figure 2: **Corridor Crawler Outpainting**

Another related work comes from the renowned Midjourney, where they delve into the concept of panning. [2]

Figure 3: **Panning banner generated with Midjourney.**

Their approach to panning involves extending the canvas in the given direction without any morphing to the original image. While this differs from my envisioned camera panning, it still seems to capture the movement of the camera and enables users to immerse themselves in a further envisioned world where the borders are expanded in either direction.

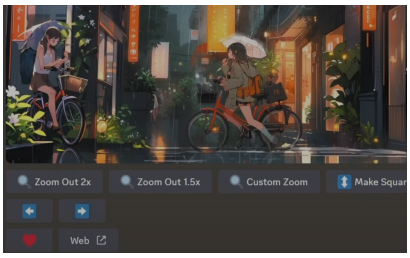

Figure 4: **Panning interface of Midjourney. [3]**

Some users even appear to be able to look around if the given image is already perspective-warped; however, further expansion generates extremely warped edges. The question arises: if implemented differently, by regenerating the entire image to include a panning effect, can we achieve true rotation or even a complete 360-degree rotation?

Another work that caught my attention is the 360-drawing.com's 360 painting tool, which seems to offer the ability to paint 360 images with generative painting. Their video on the website provides a detailed showcase of the tools, demonstrating their capability to outpaint in any given direction of the x, y, z plane of a 3D, 360 image.

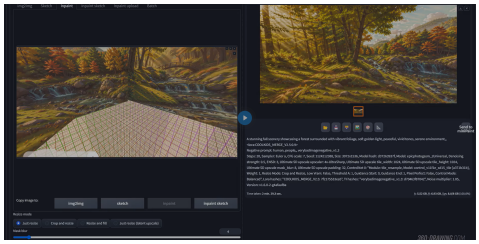

Figure 5: **360-drawing.com's tool that generates images in a 3D plane.**

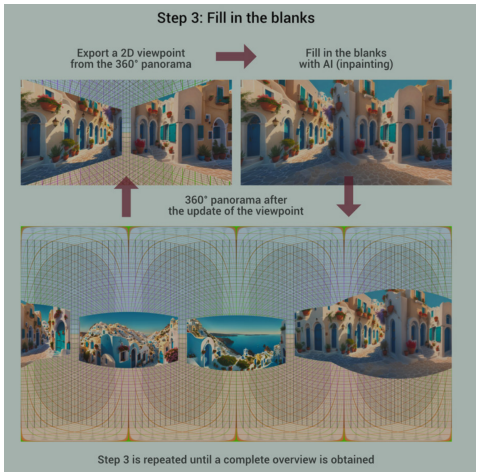

Figure 6: **Kevin Hohler's 360 panorama image generation. [4]**

This idea is explored in a couple of other works, such as the article "How to create 360 Panoramas using Artificial Intelligence" [4]. In this article, Kevin Hohler explains a similar method of visualizing a 3D environment by generating the image in a 360 panorama format.

Another work that expands on the same method is "Beyond The Seen" by Adobe, where they pose the question: "Have you ever wanted to create immersive 360° experiences from flat 2D images?" [5]

Another method involves encoding depth information as a guide to pan the image as desired, thereby addressing occlusion artifacts.

Figure 7: **"Beyond The Seen" by Adobe**

A related work in this area is the T2I-Adapter [6].

Additionally, I must acknowledge a couple of outpainting tools that have paved the way for these developments, such as stablediffusion-infinity [7]. It's also important to mention the Stable Diffusion itself, as well as Fooocus [8] for their user interface, and the Stability Matrix [9] for their package management tool for stable diffusion.

3 METHODOLOGY

As observed in the previous work section, there are two primary methods: converting images to panorama format and expanding the outpainting region. My solution falls into the second category, with a unique twist: perspective warping the main image to iterate panning over time.

This approach emerged organically to me, and currently, I haven't come across any direct work that implements it. I devised my solution by dividing it into two parts: first, warping the image, which involves addressing questions such as determining the correct angle and the extent of warping required for that angle; and second, outpainting, which entails carefully constructing the image based on a certain number of pixels and specific prompts. The overall pipeline and its parameters need to be thoroughly researched to yield optimal results. My objective is to achieve feasible 90-degree rotations while retaining context, with the potential for extension in either direction.

For warping images, I intend to utilize the OpenCV warpPerspective [10] function. I will fine-tune its parameters to achieve a certain degree of rotation, resulting in an image with incomplete edges that are to be outpainted later on.

For outpainting, I initially utilized InvokeAI [11], but later transitioned to Fooocus [8] due to its simpler interface. However, I occasionally alternate between the two to achieve the optimal result. Fooocus refers to the process as inpainting when dealing with defined regions, whereas outpainting is designated for undefined areas. On the other hand, InvokeAI's multi-layer canvas automatically identifies the operation as outpainting. Additionally, I will compare the effects of these two methods.

Upon completing one iteration of a certain degree of rotation, I will continue to iterate the process until I can achieve a complete 90-degree rotation of panning with the generated image.

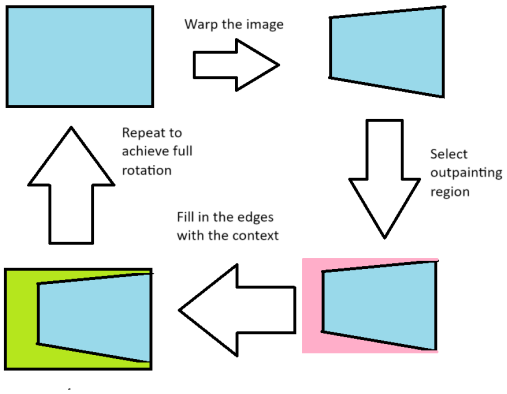

Figure 8: **Graphical explanation of my method.**

4 EXPERIMENTAL EVALUATION

Recognizing the concept behind the method, I will now present the initial results and discuss the challenges encountered along with their corresponding solutions as part of the research output. However, it's important to note that these results are merely preliminary, and as the project advances, they are bound to improve steadily towards the ultimate aim of achieving a seamless 90-degree rotation.

If you've had the chance to review the teaser banner, you may have noticed one of the initial, promising results. In this image, a school building viewed from the front is cleverly outpainted to simulate camera panning, gradually revealing additional elements such as a tree (intelligently inserted by the AI; upon closer inspection, branches can be observed on the top left of the initial image), a person, and other buildings adjacent to the school. As the panning progresses, we begin to glimpse the street beyond the building, showcasing flowerpots and additional buildings.

Figure 9: **AI effectively captures the context of the tree from the upper left corner.**

While this result leaves room for improvement, it stands as the best outcome I have been able to generate thus far. It has provided valuable insights into how and where to enhance my pipeline, insights that I intend to apply to various other images. I will seek

feedback from peers regarding the feasibility of the generated image and strive to achieve at least a 90-degree rotated image that is contextually rational and aesthetically pleasing.

Figure 10: **A current result**

Examining the end result, we can identify a couple of key areas for improvement. Firstly, the initial school building image appears extremely warped, likely due to remaining in the context for too long. Rather than smoothly panning the image, it appears to have been stretched around, resulting in an undesirable outcome. To address this issue, I propose a new method: cropping the image after generation. This approach would gradually phase out the warped portion from view, aligning with the fixed field of view (FOV) characteristic of panning.

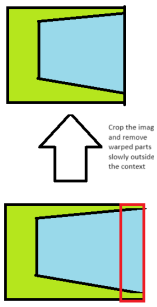

Figure 11: **A fix via cropping.**

Another issue that arises is the quality of the image. This is somewhat beyond my control as I am working with stable diffusion models available in open source, and only a few of them have outpainting capabilities. Among the models I've tested, one seemed to perform decently. However, the upcoming Stable Diffusion 3 [12] might make a significant difference if it is released in time before the term ends and is compatible with my computing power. Nonetheless, I am committed to striving for a better-looking result in terms of realism and quality. My primary metric for success is feedback from peers, which comes in the form of human-initiated critique.

To mention a couple of other test results, you can observe below the AI's creativity flourishing, albeit to a degree that hinders our main objective of panning. The general term for such objects is obstructions, and simply adding them as negative prompts did reduce their occurrence.

Figure 12: **AI generates the perspective of looking out of a car.**

Figure 13: **AI generates arches**

Figure 14: **AI generates frames.**

After removing these with negative prompting obstructions, which also eliminated the picture framing, out-of-context rubble,

and various types of arches, I moved on to the next step where encountered another issue: out-of-context placement. To address this, I opted to include more of the image in the outpainting process, rather than limiting it to regions closer to the edges.

Figure 15: **Considering the whole image will eliminate out-of-context objects.**

Another amusing result I encountered was when, while using the outpainting canvas tool with a brush, I inadvertently painted over a layer intended to provide context. In response, the AI took the initiative to exercise its creativity by replacing those areas with... water.

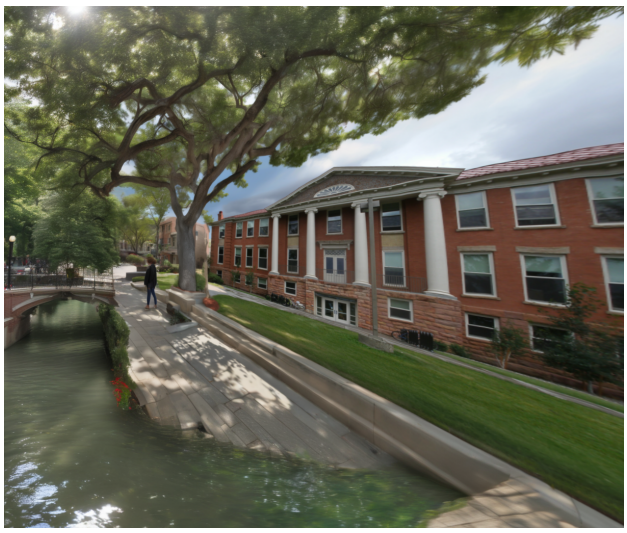

Figure 16: **The outpainting region must be chosen precisely.**

The ability to precisely select the outpainting region effectively eliminates these artifacts. To achieve this, we can utilize a predefined mask and apply it as the designated region.

Another issue arises if the processing is rushed: stretched artifacts appear in the images. These artifacts were also observed in Midjourney's results, as they did not involve a complete rewrite of the image. Consequently, this outcome reinforces my hypothesis and lays the groundwork for the future concept that low frame rate image generation might lead to stretching. However, a PC capable of generating 100+ images per second, thus achieving a 100 image-per-second FPS, should easily produce significantly higher quality results

Regarding rotations, I experimented with a 15-degree rotation. In the following result, any rotation exceeding 30 degrees at once immediately distorts the image beyond tolerance. Therefore, it seems that smaller degrees yield better outcomes. Is there a minimum degree that would result in the best outcome? To determine this, we need at least a somewhat 30 image-per-second generation rate and must actively experiment with this setup to observe whether very slight degrees result in deteriorated quality.

Figure 17: **High degrees in one iteration result in stretched artifacts.**

After the adjustments were implemented, the process proceeded smoothly, as evidenced by the teaser banner. However, the aforementioned enhancements, such as cropping and utilizing a superior

model, are expected to greatly aid in the forthcoming tests. Furthermore, I will be focusing on various images, particularly those extracted from games, considering that the research topic revolves around generating entire games on a per-image-per-second basis.

And, of course, when considering the photography aspect of this course and concept, we must view all these artifacts through an artistic lens. Upon observing the image below, we notice that flowers obstruct the view, potentially hindering further panning. However, as a standalone image, the concept offers a degree of rotation, allowing us to glimpse another building from an angle previously inaccessible. Additionally, the artistic bokeh created by the flowers contributes to a visually appealing photograph.

Figure 18: **The flowers with bokeh contribute to the creation of an artistic image.**

The latest outcome was specifically obtained from another model, with the expectation that it would yield a visually improved result with an appealing artistic touch, showing promise for achieving higher-quality outcomes in future tests. Stay tuned as I delve into further research on additional models, parameters, and artifact solutions, aiming to achieve seamless panning with a 90-degree rotation.

ACKNOWLEDGMENTS

I would like to thank Professor Erkut Erdem for his valuable assistance and insightful discussions during the preparation of this paper and conducting this research.

REFERENCES

[1] Brick2Face. (2023). Corridor Crawler Outpainting. Retrieved from https://github.com/brick2face/corridor-crawler-outpainting
[2] Midjourney. (2023). Documentation. Retrieved from https://docs.midjourney.com/docs/pan
[3] Tao Prompts. (2023). Create Panoramic Images With Midjourney's New Pan Feature! Retrieved from https://www.youtube.com/watch?v=P17BFXLHO0s
[4] Kevin Hohler. (2023). How to create 360 Panoramas using Artificial Intelligence (AI). Retrieved from https://blog.kuula.co/ai-panoramas
[5] Adobe. (2022). Project Beyond the Seen. Retrieved from https://labs.adobe.com/projects/beyond-the-seen/
[6] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, Xiaohu Qie. (2023). T2I-Adapter: Learning Adapters to Dig out More Controllable Ability for Text-to-Image Diffusion Models. Retrieved from https://arxiv.org/abs/2302.08453
[7] lkwq007. (2021). Stable Diffusion Infinity. Retrieved from https://github.com/lkwq007/stablediffusion-infinity
[8] lllyasviel. (2023). Fooocus. Retrieved from https://github.com/lllyasviel/Fooocus
[9] LykosAI. (2023). StabilityMatrix. Retrieved from https://github.com/LykosAI/StabilityMatrix
[10] OpenCV. Geometric Transformations of Images. Retrieved from https://docs.opencv.org/3.4/da/d6e/tutorial_py_geometric_transformations.html
[11] InvokeAI. (2023). InvokeAI. Retrieved from https://github.com/invoke-ai/InvokeAI
[12] Stability AI. (2024). Stable Diffusion 3. Retrieved from https://stability.ai/news/stable-diffusion-3