

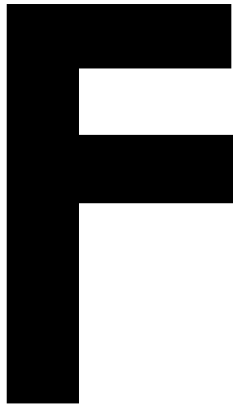
Assignment 2

Seonghyun Ban
1005127738
banseong

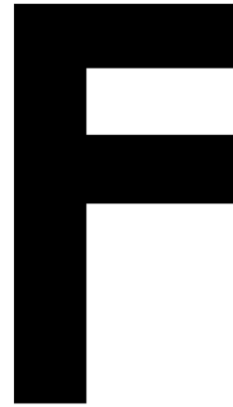
March 10, 2025

1 Comparison of the result

I find the reference output and my output to be fairly similar, although it may not be pixel perfect.

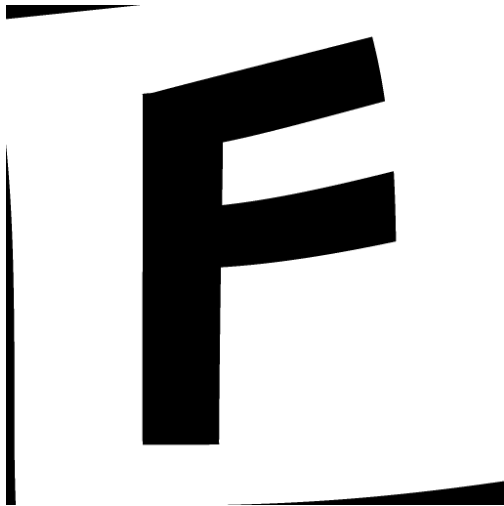


(a) Reference 1



(b) Output 1

Figure 1: Comparison of f 1 arrow



(a) Reference 2



(b) Output 2

Figure 2: Comparison of f 2 arrow

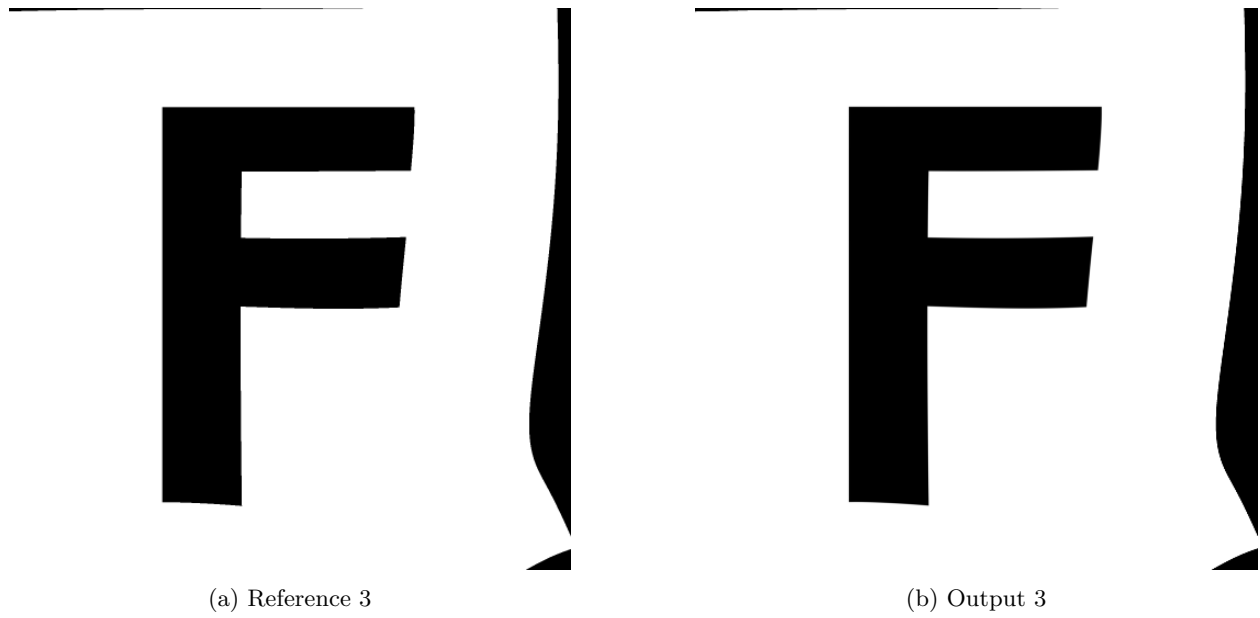


Figure 3: Comparison of f 3 arrow

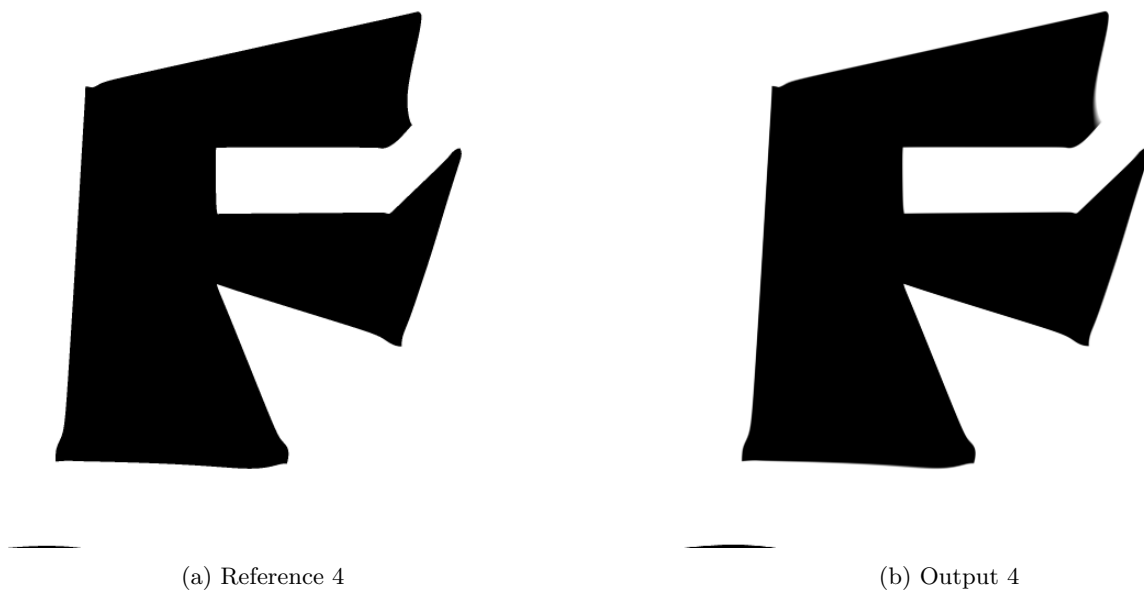


Figure 4: Comparison of f 4 arrow

2 Effect of Using Bilinear Interpolation and Supersampling

We warped an image of smoke to reshape its form while preserving its smooth appearance. The goal was to maintain natural curvature despite the transformation.

The images below are presented in this order:

- The original image.
- Warping with neither bilinear interpolation nor supersampling.
- Warping with bilinear interpolation but no supersampling.
- Warping with both bilinear interpolation and supersampling.

Nearest Neighbor Interpolation → Bilinear

Using nearest neighbor interpolation (2) produces a jagged, blocky effect, which is undesirable for smoke, as it should appear soft and fluid. Bilinear interpolation (3) improves this by blending pixel values based on the four nearest neighbors, creating smoother transitions across the image.

Single Sampling → Supersampling

However, bilinear interpolation alone does not fully eliminate artifacts present in nearest neighbor interpolation. When zooming in, white dots appear in the middle of darker regions in both methods. These artifacts are removed by supersampling (4). The reason is that bilinear interpolation can still misinterpret sharp transitions—if a pixel in a dark region is influenced by surrounding bright pixels, it may appear unnaturally bright. Supersampling solves this by averaging multiple samples across a broader area, better capturing the overall structure.

By combining bilinear interpolation with supersampling, we achieve a smoother and more natural-looking transformation of the smoke.



Figure 5: Original image.



Figure 6: Nearest Neighbor Interpolation without Supersampling.



Figure 7: Bilinear Interpolation without Supersampling.



Figure 8: Bilinear Interpolation with Supersampling.

3 Limitation of the algorithm

Field morphing, including the Beier-Neely algorithm, requires manually placing control points or lines, making it challenging to process complex, detailed images, especially at a large scale. Misplaced control elements can cause unintended distortions, and significant shape changes in the image may lead to unnatural stretching. Additionally, warping can result in interpolation from regions outside the source image, necessitating proper boundary handling techniques.