CS276 Homework 1: Light Field Rendering

Zhaoyu Qiu 2024192955
School of Biomedical Engineering
ShanghaiTech University

October 28, 2024

1 Introduction

The purpose of this assignment is to perform refocusing based on light field data and to experiment with different aperture effects. The input for this assignment is a 16x16 image matrix, referencing classic projects such as Lytro camera data processing.

2 Implement

2.1 Light Field Loading

The 16x16 light field input used for the text is shown in Figure 1.

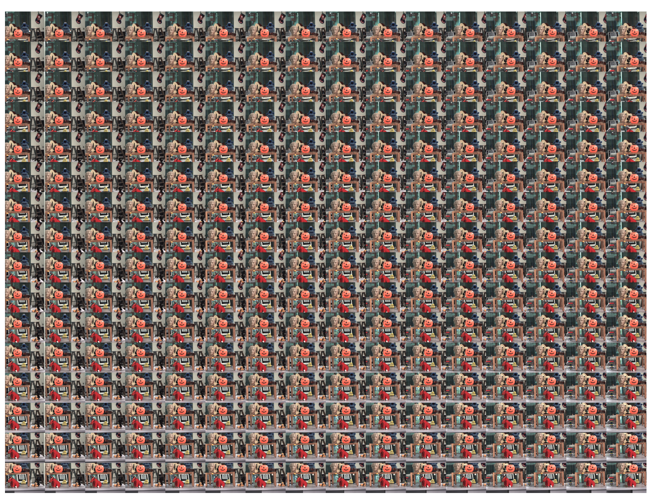


Figure 1: 16x16 Light Field Input

To create a light field, multiple images are captured from various viewpoints and organized into a 5D data structure, represented as:

$$\text{Light Field}(u, v, x, y, c)$$

where (u, v) represent the camera indices, (x, y) are the pixel coordinates, and c denotes the color channels. Each image is resized and stored in the array, ensuring uniform dimensions for accurate interpolation.

2.2 Interpolation Resampling

Interpolation is a crucial step in rendering the light field. Two primary interpolation methods are employed:

2.2.1 Bilinear Interpolation

For bilinear interpolation, the pixel value $I(x, y)$ at an arbitrary position is calculated using:

$$I(x, y) = w_a \cdot I_a + w_b \cdot I_b + w_c \cdot I_c + w_d \cdot I_d$$

where w_a, w_b, w_c, w_d are the weights based on the neighboring pixel values.

2.2.2 Trilinear Interpolation

In a 3D light field, trilinear interpolation is utilized, which extends the bilinear approach into the depth dimension. The formula is given by:

$$\begin{aligned} I(x, y, z) = {} & I_{000}(1-x)(1-y)(1-z) + I_{100}(x)(1-y)(1-z) \\ & + I_{010}(1-x)(y)(1-z) + I_{110}(x)(y)(1-z) \\ & + I_{001}(1-x)(1-y)(z) + I_{101}(x)(1-y)(z) \\ & + I_{011}(1-x)(y)(z) + I_{111}(x)(y)(z) \end{aligned}$$

2.3 Focal Plane Control

Let the camera position be $C = (x_c, y_c)$ and the focus distance be f. The distance from a pixel point $P = (x, y)$ to the camera position is calculated as:

$$d = \sqrt{(x - x_c)^2 + (y - y_c)^2}$$

The weight w for each camera view can be computed using the Gaussian function:

$$w = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{d^2}{2\sigma^2}}$$

where σ is related to the aperture size, influencing the weight distribution.

2.4 Aperture Control

Let A be the aperture size, which affects the weight calculation. A larger aperture allows more camera views to contribute to the final image. The relationship can be described as:

$$w \propto \frac{1}{A}$$

This means that as the aperture increases, the weight distribution becomes flatter, impacting the clarity and detail of the final image.

2.5 Z-Axis Control

The focal length factor F simulates depth along the z-axis. The displacement in the x and y directions for each camera view can be calculated as:

$$dx = (col - x_c) \cdot F \cdot \frac{width}{f}$$

$$dy = -(row - y_c) \cdot F \cdot \frac{height}{f}$$

Here, (col, row) represents the position of the camera view, and $(width, height)$ are the dimensions of the final image. The blur effect in the final image is controlled by the kernel size:

$$\text{blur\_kernel\_size} = \max(1, \text{int}(F \cdot 2))$$

3 Experiments

3.1 GUI interface

The user interface of this assignment consists of the components shown in the Figure 2, which includes 6 control sliders, 1 display page, and 1 confirm button.

As shown in Figure 3, the different perspectives are controlled and visualized.

3.2 Interpolation Resampling

The comparison of results between bilinear interpolation and trilinear interpolation is shown in Figure 4. In the relatively undersampled lower-left corner, the effect of multiple interpolations is superior.

3.3 Focal Plane Control

As shown in Figure 5, the effect of increasing focal length is illustrated.

3.4 Aperture Control

As shown in Figure 6, the effect of different aperture sizes is illustrated.

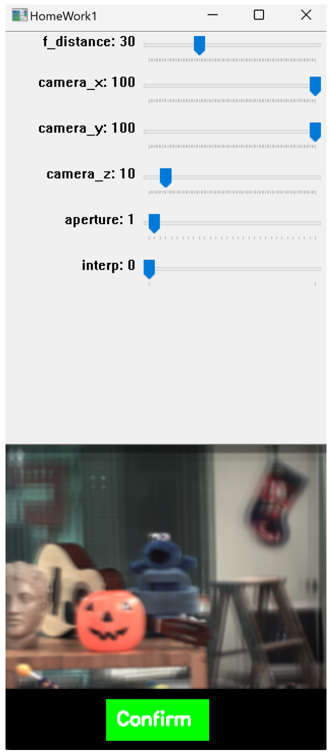
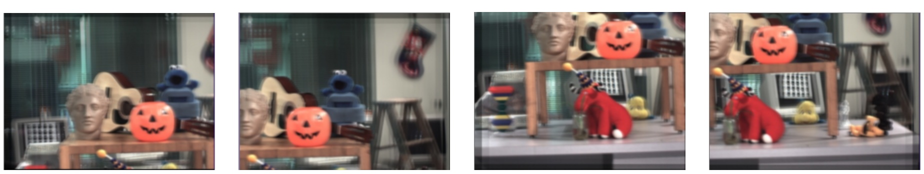


Figure 2: User interface overview



(a) Top-left (b) Top-right (c) Bottom-left (d) Bottom-right

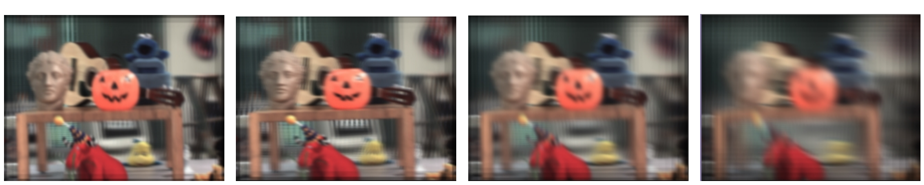
Figure 3: Control of different perspectives: top-left, bottom-left, top-right, and bottom-right views.

(a) Bilinear interpolation (b) Trilinear interpolation

Figure 4: Comparison of interpolation resampling

(a) Focal length = 10 (b) Focal length = 34 (c) Focal length = 60

Figure 5: Effect of increasing focal length from small to large



(a) Aperture 1 (b) Aperture 2 (c) Aperture 3 (d) Aperture 4

Figure 6: Effect of different aperture sizes

3.5 Z-Axis Control

(a) Z = 0 (b) Z = 1 (c) Z = 12 (d) Z = 15

Figure 7: Process of increasing Z-axis from small to large values

As shown in Figure 7, the Z-axis increases progressively in the illustrations.

4 Conclusion

This project implementation was inspired by two open-source projects: Open-source Project 1 and Open-source Project 2.

The final implementation achieved three main tasks: interpolation, view control, and focus control. Since this is my first time handling light field data, I believe the aperture control and Z-axis control are not entirely correct. I look forward to reviewing and learning from other students.