Programming Assignment 1:

Light fields are simple but useful image-based rendering primitives. In this assignment, you are required to implement two light field renderers, one using traditional light field interpolation and the other using a neural network. You also need to implement some rendering effects, such as refocusing, expanding aperture field, and translational motion of the virtual viewing camera. You need to complete the assignment independently. Please do not plagiarize, we will check.

Get started:

Download the dataset: http://pan.shanghaitech.edu.cn/cloudservice/outerLink/decode?c3Vnb24xNjYzMTQ1NDQ4NjUxc3Vnb24=

Part 1: Traditional Light Field Rendering

Deadline: 23:59, 2022.09.29

Task

1. Interpolation. Implement both the bilinear and quadra-linear interpolation schemes to interpolate views along x and y directions. You need to set up your translation step small enough so as to observe the interpolated views between the data cameras. What do you observe if you use bilinear interpolation on undersampled light field?
2. Change focal plane. Map the depth of the focal plane to the disparity. What happens when you move your focal plane from far to near? Which focal plane gives you the optimal results (least aliased reconstruction)?
3. Change aperture size. Implement a wide aperture filter to show the effect of changing aperture size. You can use Gaussian weight as described in [1]. What happens when you increase the size of the aperture?
4. Expand field of view. Implement the z-directional motion of the camera.

Part 2: Light Field Network

Deadline: 23:59, 2022.10.13

Task

1. Neural Light field renderer: Design a neural light field renderer which takes (u, v, s, t) as input, and ouputs (R, G, B) color. You can design your own network architecture, we recommend using the combination of MLP and positional encoding.
2. Evaluation. You need to do some evaluations of your design. For example, you can try different positional encoding methods and compare their results in epipolar plane image.

3. Camera translational motion. Implement the translational motion of the virtual viewing camera along x, y, and z directions.
4. Refocusing and change aperture size. You need to sample a sufficient amount of rays to achieve a large aperture effect. Compare your results with the tranditional light field rendering.

Hint

1. You may need to normalize the value of (u, v, s, t) before feeding it into the network.
2. Using the SIREN activation function may work better.

Submission

For each part, you need to compress the following into a zip file and submit it to gradescope.

- **Source code** (with model checkpoint in part 2) and a `README.md` describing how to run your code.
- **A short report** in English containing overall design and what you have done. No more than 4 pages.
- **Video demos** showing the rendering results.

Reference

[1] Isaksen, A., McMillan, L. and Gortler, S.J. Dynamically reparameterized light fields. SIGGRAPH, 2000.

Acknowledgments

This project is the first coding assignment of ShanghaiTech University Computational Photography (CS 276).

If you have any questions, welcome to ask on Piazza or contact Ziyu Wang via email: wangzy6@shanghaitech.edu.cn