2022 – 2023 第一学期 实验报告

课程编号： 2801000060， 课程名称：机器人学导论， 主讲教师： 邱国平

学号：2022280485	姓名：贾苏健	专业年级：电子信息工程 22

Coursework 1 （20%）

教师评语/成绩

1. An explanation of the principle of binocular stereo vision using **your own language**.

Stereoscopic vision, or binocular vision, is a key perceptual principle that enables biological and technological systems to perceive and understand the three-dimensional world. This principle relies on our two eyes, which capture different perspectives, generating small disparities. By comparing these disparities, the brain can calculate the depth positions of objects, creating a sense of depth.
Similarly, for computers, obtaining depth information from an image requires the use of two cameras to compute the depth of the photograph.

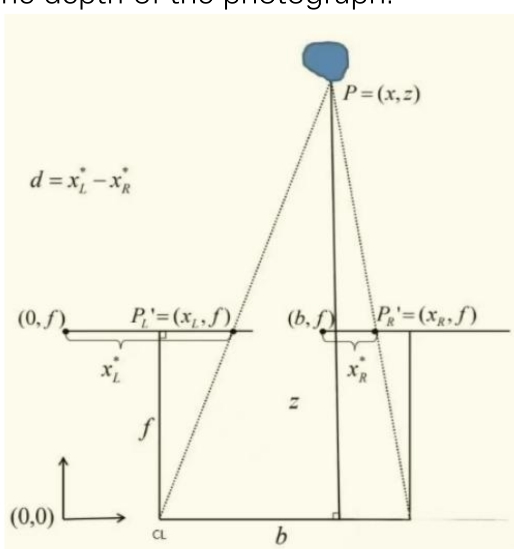

Figure 1

$$z = \frac{bf_x}{d}$$

As shown in Figure 1, the disparity "d" can be calculated from the images captured by the left and right cameras. In the formula, "b" represents the distance between the cameras, "f_x" represents the camera's focal length, and "z" represents the depth.

2. A detailed explanation of how you implemented your solution.

Import the necessary libraries, including NumPy and OpenCV, for image processing and computation.
Load the left and right images and convert them into grayscale images to prepare for the disparity calculation.
Use median filtering to denoise the images, enhancing matching accuracy.
Define the window size and maximum disparity range, which will be used in the disparity calculation.
Initialize a disparity map of the same size as the left image.
Iterate through each pixel in the left image, extract a window, and search for a match in the right image.
For each window, compute a similarity metric to find the best-matching disparity value.
Store the best-matching disparity value in the disparity map to represent depth information.
Finally, save the generated disparity map as an image file and display it in a window, waiting for a key event to view the resulting depth map.

Please refer to the code implementation for specifics.

3. A discussion on how different window sizes affect the results, and why.

Smaller windows provide finer disparity estimation but may be more sensitive to noise. Larger windows can smooth the image and reduce noise but might result in coarser depth estimation.
Using a smaller window for disparity calculation requires more computations because more windows need to be matched. Larger windows can reduce computation time but may lose some details.
The window size also affects the depth range of the disparity map. Smaller windows typically limit the maximum estimated disparity because small windows may not have enough pixels to cover large disparity cases. Larger windows can handle a broader range of disparities.

Reasons:
Smaller windows provide more detailed information but can lead to erroneous matches in the presence of noise or texture variations. Larger windows help smooth out noise but may blend regions with different depths.
The choice of window size is also influenced by computational resources. In real-time applications, a balance between computation time and accuracy must be struck, so the appropriate window size is chosen.
Smaller windows generally restrict the disparity range because the pixels within a small window might not be sufficient to represent large depth differences. Larger windows can accommodate a wider range of disparities but may result in a blurrier outcome.

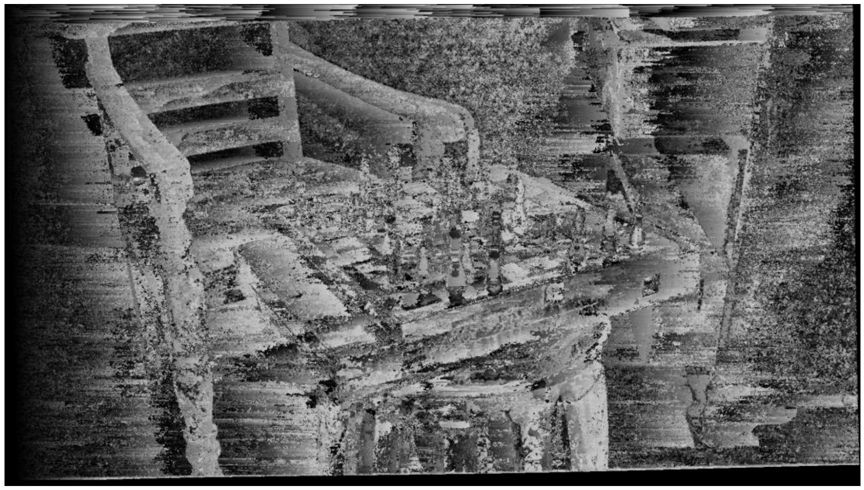

Figure 2 Window Size = 3

Figure 3 Window Size = 5

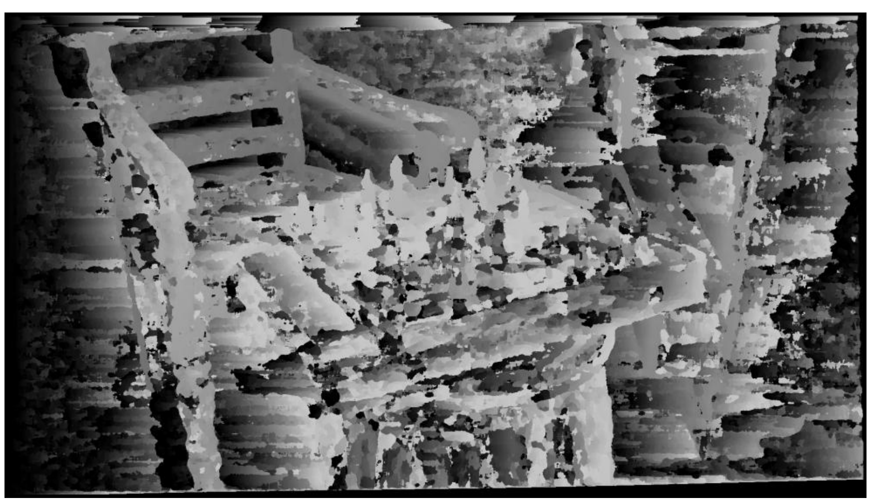

Figure 4 Window Size = 15

4.A discussion on the different similarity metrics you have used, explaining how they affect the results, and why.

$$SAD(k,l) = \sum_{(i,j)\in T} |E_l(i,j) - E_r(i+k, j+l)|$$

SAD (Sum of Absolute Differences):
SAD is a simple and intuitive similarity metric that evaluates matches by computing the sum of absolute differences between pixel values in a template and a target region. This method is relatively sensitive to changes in brightness as it considers only pixel value differences without normalizing for brightness variations. Since SAD does not normalize for brightness, its results can be affected by changes in lighting conditions, leading to instability in the resulting disparity map when lighting conditions vary.

$$SSD(k,l) = \sum_{(i,j)\in T} |E_l(i,j) - E_r(i+k, j+l)|^2$$

SSD (Sum of Squared Differences):
In contrast to SAD, SSD calculates the sum of squared differences between pixel values in a template and a target region. It is more sensitive to changes in brightness because it is influenced by variations in brightness and tends to select matches with smaller differences. However, similar to SAD, SSD does not normalize for brightness, making it less stable in the presence of lighting variations.

$$NCC(k,l) = \sum_{(i,j)\in T} \frac{\sum_{(i,j)\in T} E_l(i,j) E_r(i+k, j+l)}{\sqrt{\sum_{(i,j)\in T} E_l(i,j)^2 \sum_{(i,j)\in T} E_r(i+k, j+l)^2}}$$

NCC (Normalized Cross-Correlation):
NCC takes into account the relationships between brightness, contrast, and structure by evaluating matches through the computation of normalized cross-correlation between a template and a target region. NCC values lie within the range of [-1, 1], with 1 indicating a perfect match. NCC exhibits a degree of robustness and is relatively insensitive to changes in brightness and contrast, thanks to its consideration of normalized pixel values. Therefore, in the presence of lighting and contrast variations, NCC often provides more stable disparity estimates.

The choice of these three similarity metrics should be determined based on the specific application context. If the application involves significant variations in lighting conditions, NCC may be the better choice due to its increased stability under such conditions. However, if lighting variations are minimal, and computational efficiency is of utmost importance, SAD or SSD might be more suitable. Additionally, it is possible to combine multiple similarity metrics to enhance matching robustness, such as using weighted combinations or multiscale matching. The ultimate choice will depend on the specific requirements of the problem and the characteristics of the data. The following is an example chart illustrating the different impacts of these three methods:

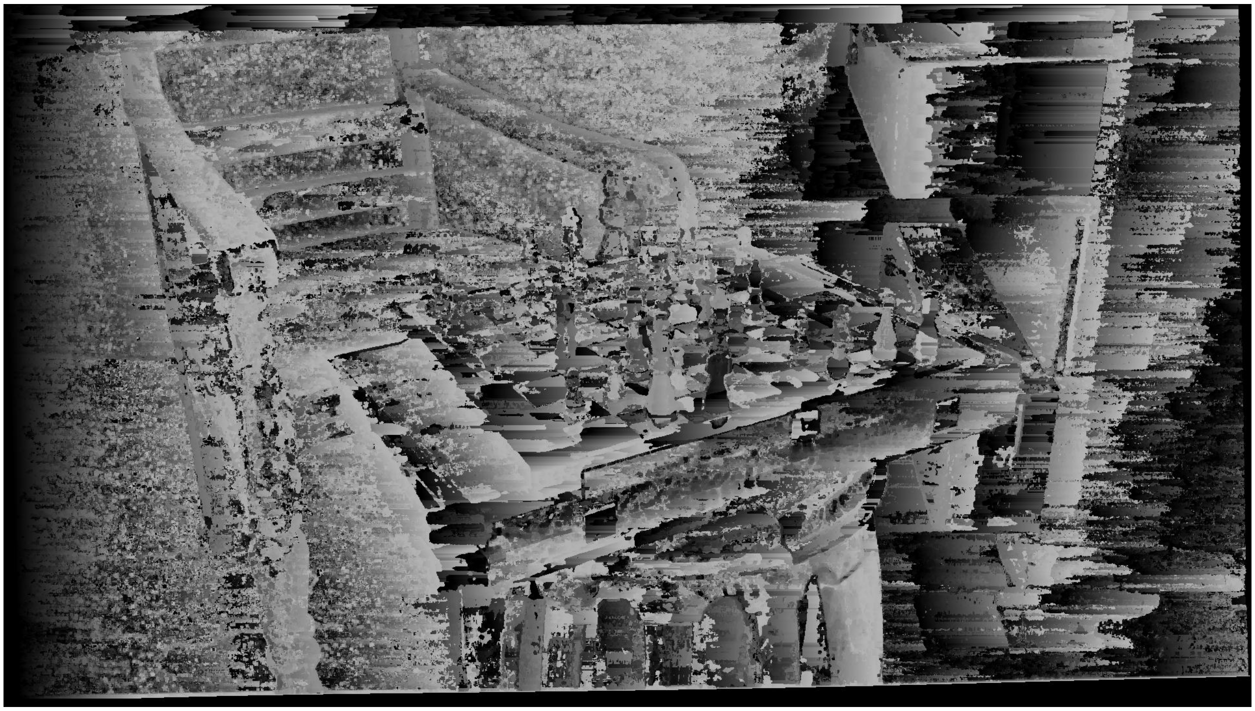

Figure 5 method = SAD

Figure 6 method = SSD

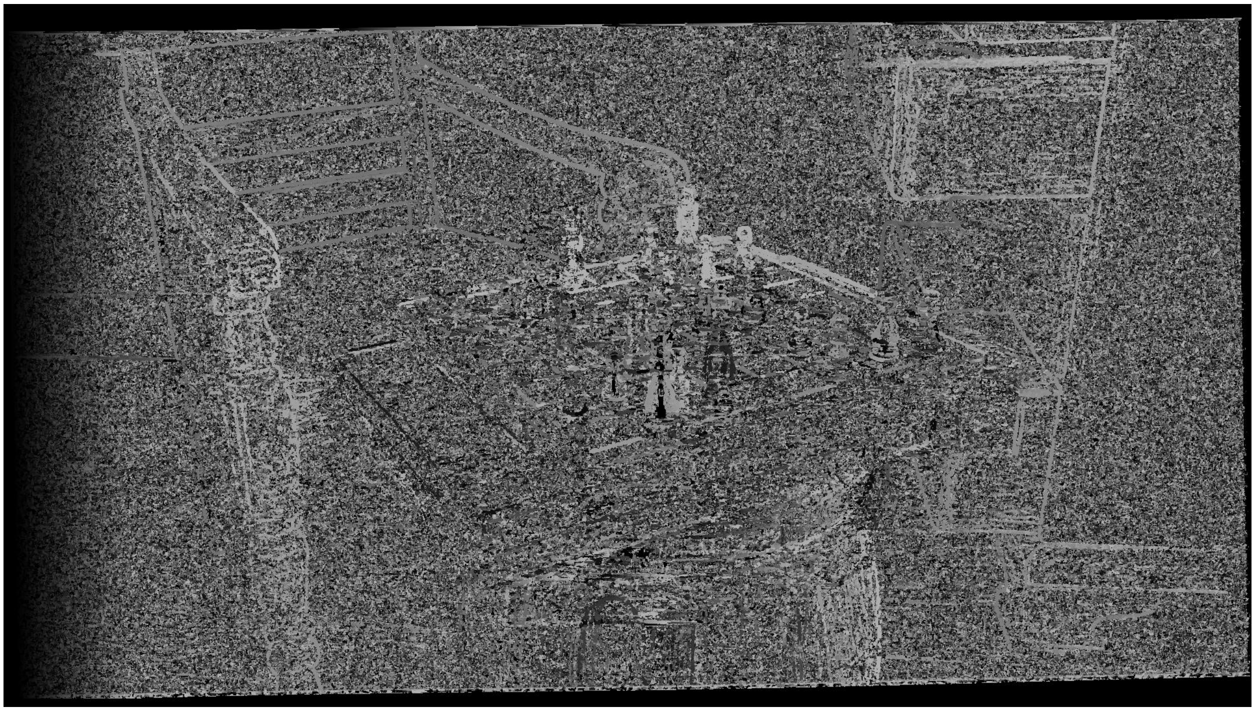

Figure 6 method = NCC

(The three images above use SAD, SSD, and NCC as similarity measures, with a window size of 5 and a disparity range of 0-180.)

From the chart, it is evident that NCC is more stable in the presence of lighting variations, whereas SAD and SSD are more affected by changes in lighting conditions. Nevertheless, in some scenarios, SAD and SSD may perform better in noise reduction due to their consideration of absolute pixel differences.