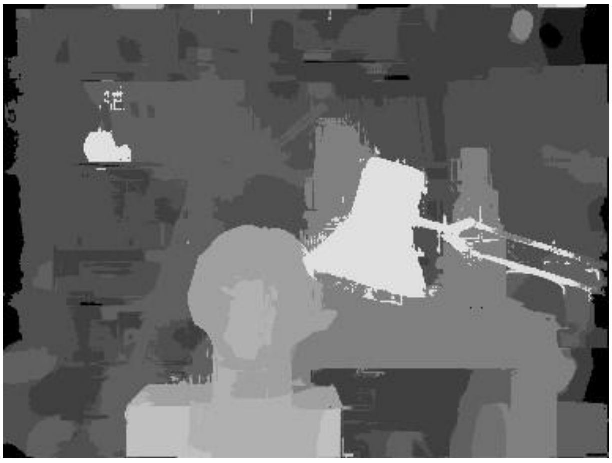
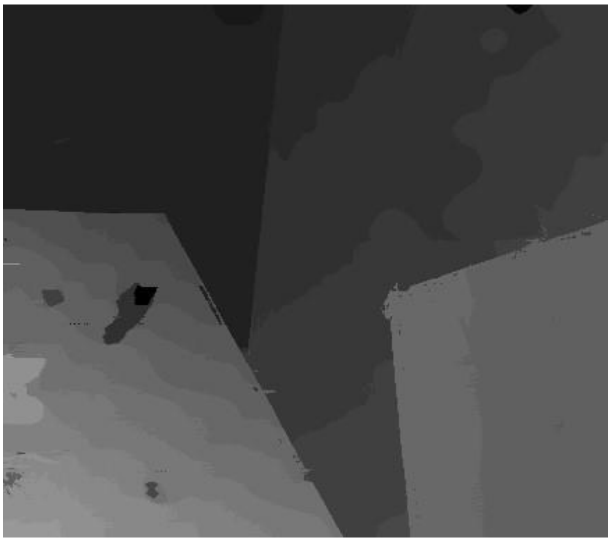
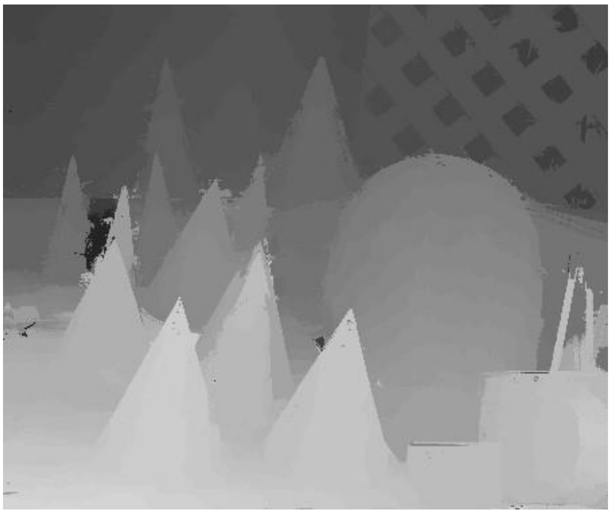
Computer Vision HW4 Report

Student ID: R11521201
Name: 廖沁柔

Visualize the disparity map of 4 testing images.

Tsukuba	Venus
	
Teddy	Cones
	

Report the bad pixel ratio of 2 testing images with given ground truth (Tsukuba/Teddy).

	bad pixel ratio
Tsukuba	4.69%
Teddy	9.84%

Describe your algorithm in terms of 4-step pipeline.

Step 1: Cost Computation

In this step, the code estimates matching costs by calculating the "Census Transform" and "Hamming Distance" between the left and right images. The Census Transform is a non-parametric feature transformation that generates a binary feature by comparing the brightness of a pixel with its neighborhood. For each pixel, its feature is compared with the corresponding pixel's feature in the right image at the given disparity, using an **XOR operation** to calculate the differences (Hamming Distance). These differences are accumulated to form preliminary cost values, which are stored in **the "cost_box" list** for subsequent processing.

Step 2: Cost Aggregation

Cost aggregation is performed using a **Joint Bilateral Filter (JBF)**. This filter considers the spatial proximity and pixel value similarity of the image, **smoothing the costs while preserving edges**. The JBF uses the left image as a guidance image and smooths the costs for each disparity. This helps to reduce noise introduced during computation and improves the quality of disparity estimation.

Three hyperparameters are defined:

JBF_s (Spatial sigma) = 10
JBF_d (Diameter sigma) = -1
JBF_c (Color sigma) = 5

Step 3: Disparity Optimization

Winner-take-all

After cost aggregation, each pixel position will have a series of disparity costs. This step optimizes the disparity estimation by selecting the disparity with the **lowest cost**. This is commonly referred to as the **"winner-take-all"** strategy because, for each pixel, the disparity with the lowest cost is chosen as the result. This forms a preliminary disparity map.

Step 4: Disparity Refinement

After obtaining the preliminary disparity map, the following refinement processes are applied to enhance its accuracy and reliability:

- **Left-right consistency check:**
  Compute the disparity from the right to the left image and compare it with the disparity from the left to the right image. This helps identify and correct potential matching errors.
- **Hole filling:**
  For pixels marked as inconsistent in the consistency check, fill these "holes" by finding the nearest consistent disparity value. Typically, the smaller of the consistent disparity values from either side is chosen.
- **Weighted median filtering:**
  Further smooth the disparity map using weighted median filtering to reduce noise and preserve edge clarity. The filter uses the grayscale image as a weight, which helps maintain the accuracy of disparity near edges.