

Exercises in 3D Computer Vision II

In this exercise, we will consider a system consisting of a Time-of-Flight (ToF) camera rigidly connected with a regular, high-resolution RGB camera in a stereo setup. We will see how to calibrate the cameras, how to reconstruct 3D point clouds from the ToF depth data and how to match the depth information with the color information from the RGB camera.

Exercise 1 **ToF and RGB Camera Calibration**

- Explain the necessary process for determining the intrinsic camera parameters \mathbf{K}_{rgb} and \mathbf{K}_{tof} and the lens distortion parameters \mathbf{d}_{rgb} and \mathbf{d}_{tof} of the RGB and the ToF camera, respectively.
- Explain the calibration steps that are necessary as a prerequisite to relate the pixels in the ToF depth image to their corresponding pixels in the RGB image.

Exercise 2 **Reconstruction of Metric 3D Data**

- Outline how to reconstruct a metric 3D point cloud from the ToF depth data. Assume that you are given the depth d at each pixel location $\mathbf{x} = (x, y)^\top$ in the ToF depth image. We would like to obtain a set of 3D points with coordinates $\mathbf{X} = (X, Y, Z)^\top$.
- Explain why you need to normalize the depth when computing the 3D point coordinates.

Exercise 3 **Matching between ToF and RGB Data**

- Show how to map each 3D point to the coordinate frame of the RGB camera.
- Explain how to find the corresponding color value for each 3D point.

Exercise 4 **(H) Colored 3D Point Cloud in Matlab**

The homework consists of implementing a 3D reconstruction method that uses the depth data from the ToF camera and colors the 3D points according to the information from the RGB camera. Assume that the ToF and RGB cameras are rigidly connected in a stereo setup.

- You are provided with depth data, the corresponding RGB images and all calibration parameters (\mathbf{R} , \mathbf{t} , \mathbf{K}_{tof} , \mathbf{K}_{rgb} , \mathbf{d}_{tof} , \mathbf{d}_{rgb}). We also provide a Matlab function `readToFImageBinary.m` to access the binary ToF data files. Download the material from the lecture website.
- Implement a Matlab function that takes a corresponding pair of depth and RGB images and displays a metric 3D point cloud where every point is colored according to the information from the RGB image. Pay attention to correctly match the ToF and RGB camera views. *Hint:* To draw colored 3D points, you can use the Matlab function `scatter3`, see its documentation. Note that you only have to draw the points for which you have depth data.