

CS 6384 Computer Vision Homework 1

Professor Yu Xiang

January 30, 2023

Download the [homework1_programming.zip](#) file from eLearning, Assignments, Homework 1. Finish the following programming problems and submit your scripts to eLearning. You can zip all the data and files for submission. TA will run your scripts to verify them.

Install the Python packages needed by

- `pip install -r requirement.txt`

Here are some useful resources:

- Python basics <https://pythonbasics.org/>
- Numpy <https://numpy.org/doc/stable/user/basics.html>
- OpenCV https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html

Problem 1

(4 points) 2D Transformations.

Implement the `transform()` function in [image_transformations.py](#). The function takes an image and a 2D transformation T , a 3×3 matrix, as input, and outputs a transformed image according to the transformation T .

After your implementation, run the [image_transformations.py](#) in Python to verify it. Figure 1 shows an example of running the script.

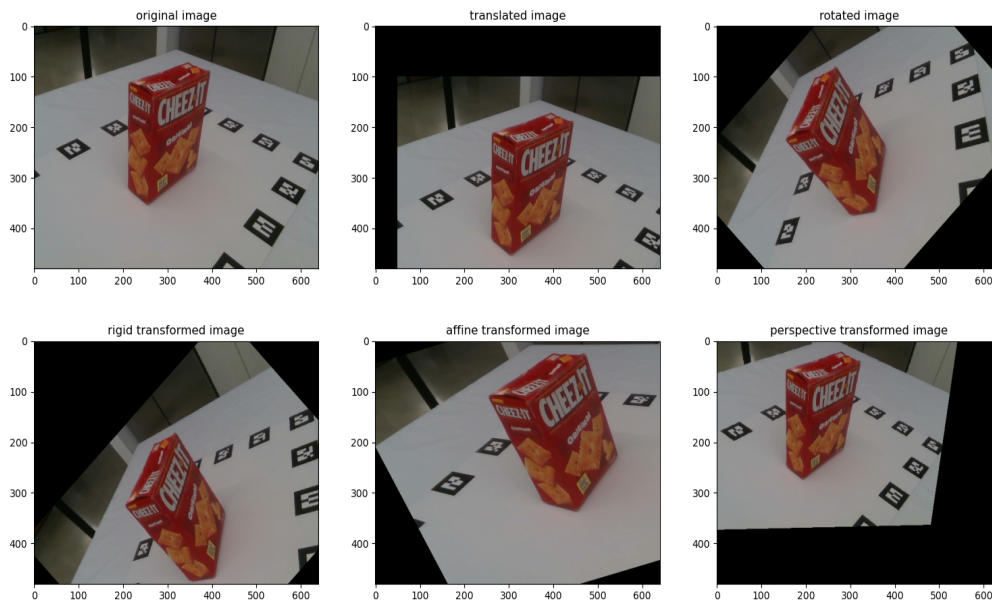


Figure 1: Image transformations.

Problem 2

(3 points) Backprojection.

Implement the `backproject()` function in `backproject.py`. The function takes a depth image with height H and width W , and a camera intrinsics matrix as input, and outputs a point cloud with shape $H \times W \times 3$ generated from the depth image.

After your implementation, run the `backproject.py` in Python to verify it. Figure 2 shows an example of running the script. It shows the 3D points on the cracker box.

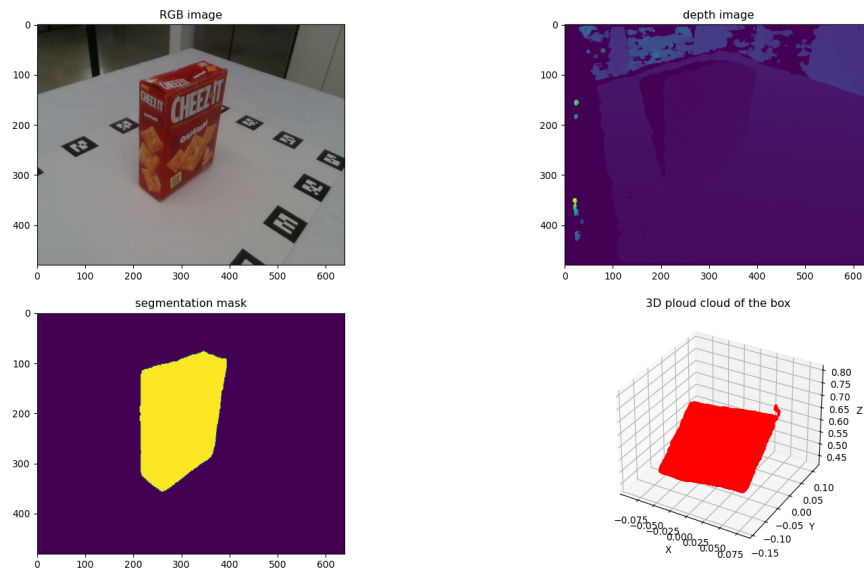


Figure 2: Backprojection.

Problem 3

(3 points) Correspondences.

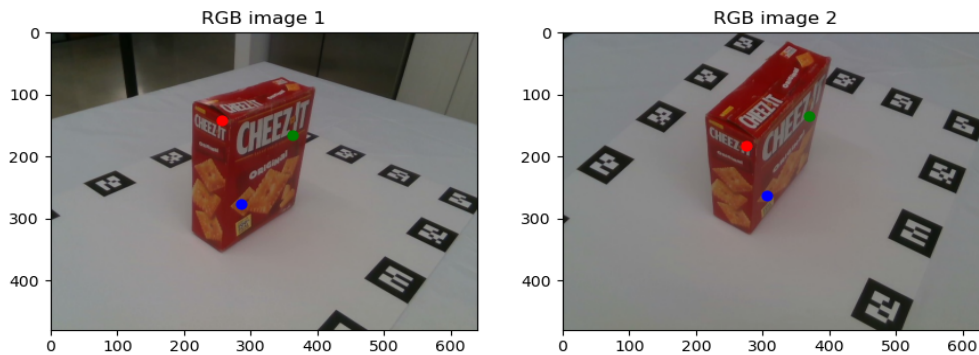


Figure 3: Correspondences

Implement the missing code in `correspondences.py`. This script finds the correspondences of pixels between two images using their camera parameters. You need to use the `backproject()` function in Problem 2.

After your implementation, run the `correspondences.py` in Python to verify it. Figure 3 shows an example of running the script. The 2D points with the same color correspond to each other.