CV Master project: Pinhole camera exercise

Task Description

In this exercise we will write a program that can take a picture. You are provided with a real world object and an image plane. Your task is to project this object onto the image plane with a virtual camera.

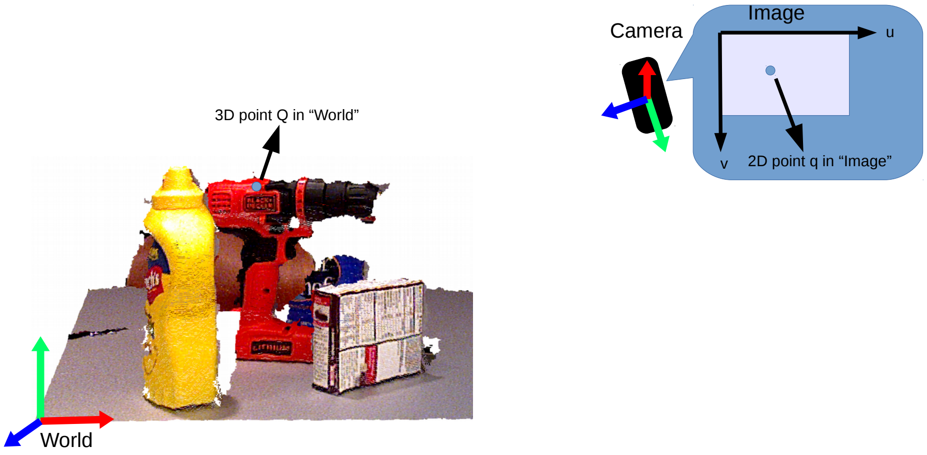

Figure 1: Illustration of the problem statement.

Method

To project a 3D point to a 2D image with a virtual camera, we need to know the camera model and the transformation matrix.

- Pinhole camera projects a 3D point $\vec{Q} = \begin{bmatrix} X & Y & Z \end{bmatrix}^T$ to a point $\vec{q} = s \cdot \begin{bmatrix} u & v & 1 \end{bmatrix}^T$ on image plane, where $s \in \mathbb{R}$

- $\vec{q} \sim C\vec{Q}$, where $\sim$ means “is equal up to scalar multiplication”, with camera projection matrix

$$C = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

- The transformation T is defined by a 3×3 rotation and a 3D translation

$$T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \tag{2}$$

- Figure 1 is an illustration of the problem statement. There are three coordinate, ”World”, ”Camera” and ”Image”. we have $\vec{Q}$ defined in the ”World” coordinate,

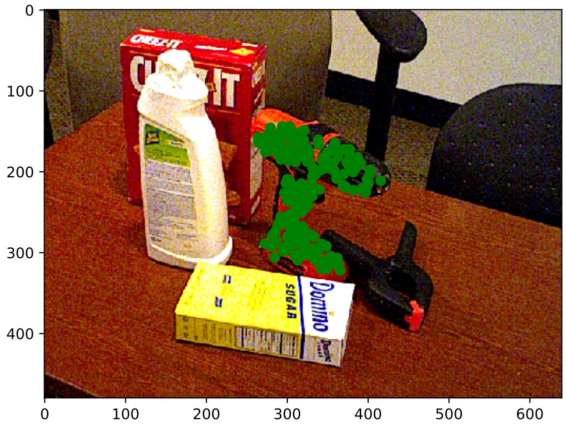

Figure 2: The image shown with the 3D points projected to the image plane (green dots).

T helps to transform $\vec{Q}$ from "World" to "Camera" coordinate, and finally C helps to transform to the "Image" coordinate. Overall, $\vec{q} = CT\vec{Q}$

$$s\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & t_x\\r_{21} & r_{22} & r_{23} & t_y\\r_{31} & r_{32} & r_{33} & t_z\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix} \tag{3}$$

Implementation details

- The object is represented with a bunch of 3D points, that lies on the object surface. You can find it in file `point_3D.txt`. It can be loaded with

```
point_3D = np.loadtxt("point_3D.txt")
```

- The camera model C and transformation matrix T are provided in file `take_a_picture.py`.
- Complete `take_a_picture.py` with the help of Equation 3 for the 3D-2D projection.
- To verify your projection, an image containing the object is provided (`000001-color.png`). You can draw the projected points onto the image for visual inspection. The result should look like Figure 2.