Exercise on Intrinsics, Extrinsics & Perspective-3-Point

March 19, 2019

In this exercise, we will work with a few assignments related to intrinsic and extrinsic camera parameters, camera calibration, and pose estimation using perspective-3-point method.

1 Intrinsics/Extrinsics

Here you should exercise in computing with the pinhole camera model. For you to have something to 'photograph'. a sample object is supplied in the accompanying file **Box3D.mat**. To see this box try the following script:

```
clear
close all
Q=Box3D;
plot3(Q(1,:),Q(2,:),Q(3,:),'.'),
axis equal
axis([-1 1 -1 1 -1 5])
xlabel('x')
ylabel('y')
zlabel('z')
```

Q1: The intrinsics and extrinsics are given as:

- the rotation is given by the function **R** =**Rxyz**(0.2,-0.3,0.1).
- $\mathbf{t} = \begin{bmatrix} 0.88 \\ 0.57 \\ 0.19 \end{bmatrix}$.
- $f = 1000, c_u = 300, c_v = 200$.

Form the projection matrix as $\mathbf{P} = \mathbf{K}\left(\mathbf{R}|\mathbf{t}\right)$ and project the 3D points from the Box3D. Plot the image you get.

Q2: Using the same pinhole camera as in Question 1, extend the model with radial distortion with $k1 = -5e^{-1}$, $k2 = -3e^{-1}$, $k3 = -5e^{-2}$, project the Box3D points and compare to the results from Question 1. Plot again and observe what has changed.

2 Camera Calibration

In this exercise, you are asked to do the camera calibration for given images. There are two calibration toolboxes that you can use:

- **Camera Calibrator** from MATLAB computer vision toolbox.
- **Camera Calibration Toolbox** for MATLAB by Jean-Yves Bouguet[1]: this toolbox is one of the earliest camera calibration toolbox.

[1] http://www.vision.caltech.edu/bouguetj/calib_doc/

Q3:

1. read relevant docs to learn how to use those toolboxes.
2. load images prepared in the sub-folder **calibration** and complete camera calibration.
3. verify the calibrated results and interpret the results you get.

Q4: Use the previous calibrated parameters to rectify a distorted image **distort.bmp**. **Hints:** Recall that the distortion models are nonlinear, so it is not trivial to directly find the undistorted points from distorted points with known distorsion parameters. In stead, we can do it as follows:

- Initialize a blank image.
- For each pixel (u, v), compute the so called normalized coordiante (u_n, v_n) as $u_n = \frac{u-c_u}{f_u}, \ v_n = \frac{v-c_v}{f_v}$.
- Apply the radial and tangent distorsions to (u_n, v_n) and denote the result as (u_d, v_d).
- Find the nearest neighbor of (u_d, v_d) in distorted image and copy the corresponding intensity (or RGB) for (u, v).

Better results can be obtained by applying bilinear interpolation.

3 Perspective 3 Point

References

Figure 1: Example of undistorted image.