Exercise 1 - Theory

Abdelaziz, Ibrahim   Somkiadcharoen, Robroo   Berg, Oliver

December 1, 2017

1 Properties of Rotation Matrices

1.1 Showing that $U^T = U^{-1}$ holds for ***general*** rotation matrix $U \in \mathbb{R}^3$

Generally note that $UU^{-1} = I$ subject to $U^{-1} = U^T$, as provided by U being rotation matrix and as such orthogonal, yields $UU^T = I$.

For given column-vectors c_1, c_2, c_3 of U we observe

$$< c_i, c_j >= \delta_{ij}$$

which in combination with

$$U^T = (c_1, c_2, c_3)^T = (c_1^T, c_2^T, c_2^T)$$

results in

$$UU^T = (< c_i, c_j >)_{1 \leq i,j \leq 3} = I$$

Motivated by [1][2][3].

1.2 Geometric interpretation of determinant of 3x3 matrix

The determinant of a matrix $A \in \mathbb{R}^3$ has the geometric intuition of a scaling factor for a given (unit) volume within the cartesian (standard) coordinate system and how its volume changes when applying the matrix transform. [4]

As such, with $det(A) = 1$ the matrix does not change the size of any given volume within its space, which intuitively makes sense for a rotation matrices as its only task is to rotate - not scale - within any given coordinate system.

2 Transformation Chain

Normally, the transformation mapping from the world coordinate system to pixel coordinate system is.

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = K[I_3|O_3] \begin{bmatrix} R & T \\ 0_3^T & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

Where x_{pix} and y_{pix} are u'/w' and v'/w' and $\begin{bmatrix} X_s & Y_s & Z_s & 1 \end{bmatrix}^T$ are the coordinates of a 3D point in the world coordinate space

Homogeneous Coordinates is good. It enables us to use Matrix Multiplication on all transformations. To represent 3d coordinate points you need to have 4 elements in the vector.

$$[x, y, z, w] = [x/w, y/w, z/w] \tag{1}$$

However, when w in (1) is 0. There is no such 3D vector on that. We call points in this form vanishing point.[5]

2.1 Chain Step

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0_3^T & 1 \end{bmatrix} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

multiplying $\begin{bmatrix} R & T \\ 0_3^T & 1 \end{bmatrix}$ to the world coordinates $[X_s, Y_s, Z_s, 1]^T$ gives a camera coordinates $[x_s, y_s, z_s, 1]^T$

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} a_x & s & x_0 & 0 \\ 0 & a_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix}$$

By multiplying the camera coordinate points with calibration matrix, the result is in the image plane coordinates. After that we can do u'/w' and v'/w' to get the pixel coordinates

2.2 Intrinsic parameters

K is a camera matrix, or a matrix of intrinsic parameters $K = \begin{bmatrix} a_x & s & x_0 \\ 0 & a_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$ Where

- a_x and a_y are focal lengths in pixel
- x_0 and y_0 are the image center
- s the skew coefficient between the x,y axis which is 0 in most cases [6][7]

K can transform image plane to pixels and $\begin{bmatrix} I_3 & | & O_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ can be use for perspective projection which can be combined to K matrix and facilitate the matrix multiplication

2.3 Extrinsic parameters

$$extrinsicParams = \begin{bmatrix} R & T \\ 0_3^T & 1 \end{bmatrix}$$

This extrinsic parameters is used to explain the camera motion along the scene. It translates world-coordinates of a point $[X, Y, Z, 1]$ to a camera coordinate system

$$R = \begin{bmatrix} I_i & J_i & K_i \\ I_j & J_j & K_j \\ I_k & J_k & K_k \end{bmatrix} T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

R is a rotational matrix such that it helps rotating the point from world to camera. T is a translational metric to translate the point from world to camera. Motivated from [8]

3 Implementation

The final implementation task can be found in `main.py`.

Concerning output (images in *output* directory), projected points without correction of distortion are shown in red, those with correction of radial distortion in green; see Figure 1.

References

[1] matrix transpose * itself = identity. [Online]. Available: https://math.stackexchange.com/questions/768098/matrix-transpose-itself-identity

[2] Why is inverse of orthogonal matrix is its transpose. [Online]. Available: https://math.stackexchange.com/questions/1097422/why-is-inverse-of-orthogonal-matrix-is-its-transpose

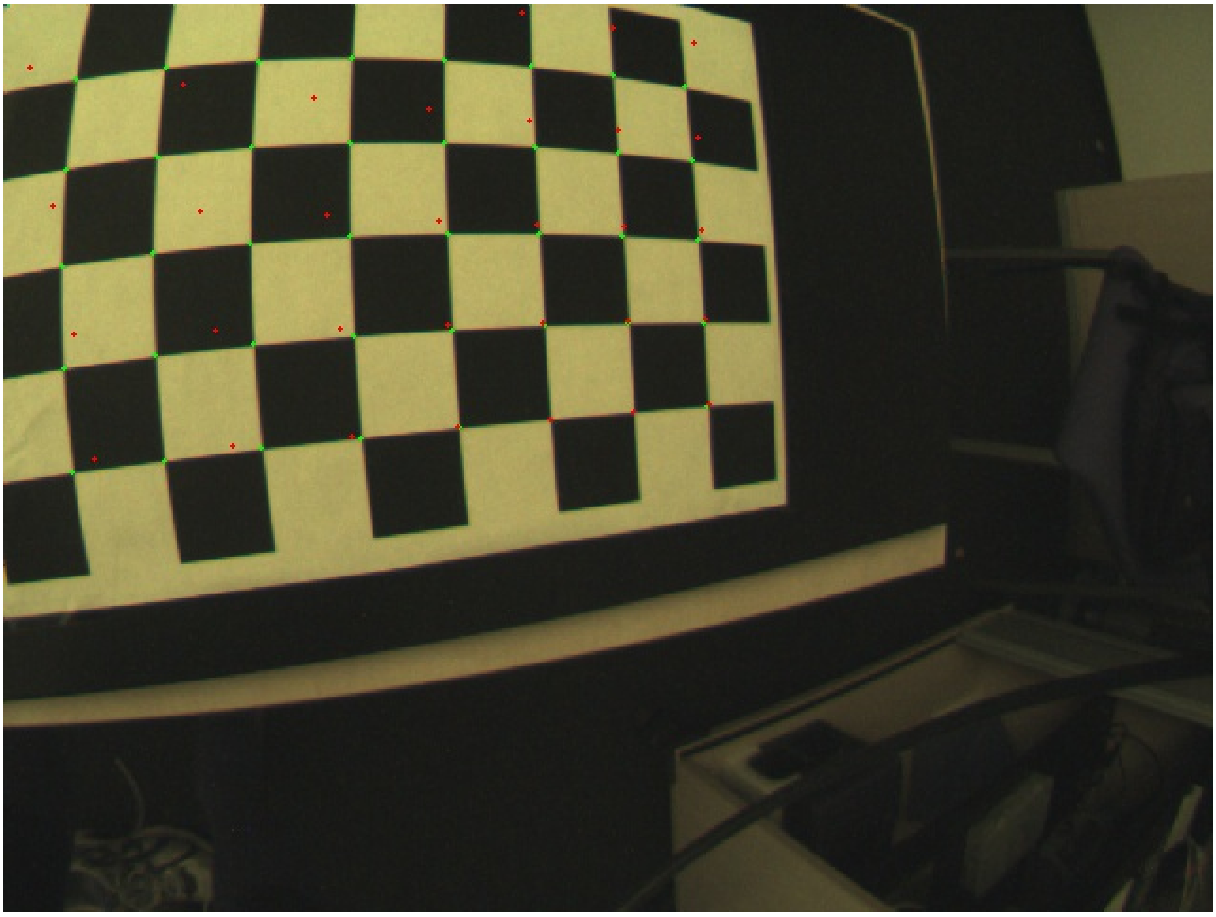

Figure 1: Sample output image

[3] Orthogonale matrix. [Online]. Available: https://de.wikipedia.org/wiki/Orthogonale_Matrix

[4] The determinant | essence of linear algebra, chapter 5. [Online]. Available: https://www.youtube.com/watch?v=Ip3X9LOh2dk

[5] The truth behind homogeneous coordinates. [Online]. Available: http://deltaorange.com/2012/03/08/the-truth-behind-homogenous-coordinates/

[6] Camera resectioning. [Online]. Available: https://en.wikipedia.org/wiki/Camera_resectioning

[7] Camera calibration and 3d reconstruction. [Online]. Available: https://docs.opencv.org/2.4.13/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

[8] Some linear algebra. [Online]. Available: http://math.hws.edu/graphicsbook/c3/s5.html#gl1geom.5.3