University at Buffalo
Department of Computer Science and Engineering
CSE 473/573 - Computer Vision and Image Processing

Fall 2025

Project #1

Due Date: 9/30/2025, 11:59PM EST

1 Q. Computing Rotation Matrix (4 points)

Figure 1 illustrates the transformation from coordinate xyz to XYZ: 1) rotate around z axis with α (the current coordinate is x'y'z'); 2) rotate around $x\prime$ axis with β (the current coordinate is x"y"z"); 3) rotate around $z\prime\prime$ axis with γ (the current coordinate is XYZ). α, β, and γ are all given in degrees (not radians), and $0° < \alpha, \beta, \gamma < 90°$.

- Design a program to get the rotation matrix from xyz to XYZ. (2 points)
- Design a program to get the rotation matrix from XYZ to xyz. (2 points)

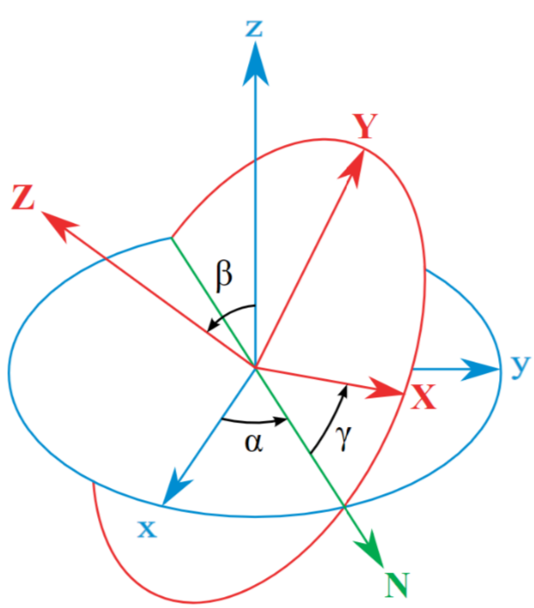

Figure 1: Illustration of Euler angles

We may test your code using different α, β, and γs in grading. Please check instructions on implementation at the end of this document.

2 Q. Conduct Camera Calibration (6 points)

Preliminary.1.
The projection from world coordinate to image plane can be indicated by intrinsic parameters (Camera) and extrinsic parameters(World). From world coordinate to camera coordinate, the extrinsic parameters can be used as

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

$$\begin{gathered} M = M_{in} \cdot M_{ex} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \\ \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} = \begin{bmatrix} f_x r_{11} + o_x r_{31} & f_x r_{12} + o_x r_{32} & f_x r_{13} + o_x r_{33} & f_x T_x + o_x T_z \\ f_y r_{21} + o_y r_{31} & f_y r_{22} + o_y r_{32} & f_y r_{23} + o_y r_{33} & f_y T_y + o_y T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}. \end{gathered} \tag{1}$$

Here, M is projection matrix. Let's define $\mathbf{m_1} = (m_{11}, m_{12}, m_{13})^T, \mathbf{m_2} = (m_{21}, m_{22}, m_{23})^T, \mathbf{m_3} = (m_{31}, m_{32}, m_{33})^T, \mathbf{m_4} = (m_{14}, m_{24}, m_{34})^T$. Also define $\mathbf{r_1} = (r_{11}, r_{12}, r_{13})^T, \mathbf{r_2} = (r_{21}, r_{22}, r_{23})^T, \mathbf{r_3} = (r_{31}, r_{32}, r_{33})^T$. Observe that $(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3})$ is the rotation matrix, then

$$(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}) \begin{pmatrix} \mathbf{r_1}^T \\ \mathbf{r_2}^T \\ \mathbf{r_3}^T \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Then we have $\mathbf{r}_i^T \mathbf{r}_i = 1, \quad \mathbf{r}_i^T \mathbf{r}_j = 0 \ (i \neq j)$.

From M we have

$$\begin{aligned} \mathbf{m}_1^T \mathbf{m}_3 &= r_{31}(f_x r_{11} + o_x r_{31}) + r_{32}(f_x r_{12} + o_x r_{32}) + r_{33}(f_x r_{13} + o_x r_{33}) \\ &= f_x(r_{11} r_{31} + r_{12} r_{32} + r_{13} r_{33}) + o_x(r_{31}^2 + r_{32}^2 + r_{33}^2) \\ &= f_x(\mathbf{r}_1^T \mathbf{r}_3) + o_x(\mathbf{r}_3^T \mathbf{r}_3) \\ &= o_x \end{aligned} \tag{2}$$

Similarly, Next, from M we have

$$\begin{aligned} \mathbf{m}_1^T \mathbf{m}_1 &= (f_x r_{11} + o_x r_{31})^2 + (f_x r_{12} + o_x r_{32})^2 + (f_x r_{13} + o_x r_{33})^2 \\ &= f_x^2 \cdot \mathbf{r_1}^T \mathbf{r_1} + 2 f_x o_x \cdot \mathbf{r_1}^T \mathbf{r_3} + o_x^2 \cdot \mathbf{r_3}^T \mathbf{r_3} = f_x^2 + o_x^2 \end{aligned} \tag{3}$$

So $f_x = \sqrt{\mathbf{m}_1^T \mathbf{m}_1 - o_x^2}$. Similarly we have $o_y = \mathbf{m}_2^T \mathbf{m}_3$, $f_y = \sqrt{\mathbf{m}_2^T \mathbf{m}_2 - o_y^2}$. Overall, we come to the conclusion as follows

$$o_x = \mathbf{m}_1^T \mathbf{m}_3 \quad o_y = \mathbf{m}_2^T \mathbf{m}_3 \tag{4}$$

$$f_x = \sqrt{\mathbf{m}_1^T \mathbf{m}_1 - o_x^2} \quad f_y = \sqrt{\mathbf{m}_2^T \mathbf{m}_2 - o_y^2} \tag{5}$$

Preliminary.2.
Let $X_w Y_w Z_w$ be the world coordinate and xy be the image coordinate, we have the transformation matrix $M \in \mathbb{R}^{3\times 4}$:

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{6}$$

$$
\begin{aligned}
sx &= m_{11}X_w + m_{12}Y_w + m_{13}Z_w + m_{14}, \\
sy &= m_{21}X_w + m_{22}Y_w + m_{23}Z_w + m_{24}, \\
s &= m_{31}X_w + m_{32}Y_w + m_{33}Z_w + m_{34}.
\end{aligned} \tag{7}
$$

We can solve m_{ij} with the equation below:

$$
\begin{bmatrix}
X_w^1 & Y_w^1 & Z_w^1 & 1 & 0 & 0 & 0 & 0 & -x^1X_w^1 & -x^1Y_w^1 & -x^1Z_w^1 & -x^1 \\
0 & 0 & 0 & 0 & X_w^1 & Y_w^1 & Z_w^1 & 1 & -y^1X_w^1 & -y^1Y_w^1 & -y^1Z_w^1 & -y^1 \\
 & & & & & & . & & & & & \\
 & & & & & & . & & & & & \\
 & & & & & & . & & & & & \\
 & & & & & & . & & & & & \\
 & & & & & & . & & & & & \\
 & & & & & & . & & & & & \\
X_w^n & Y_w^n & Z_w^n & 1 & 0 & 0 & 0 & 0 & -x^nX_w^n & -x^nY_w^n & -x^nZ_w^n & -x^n \\
0 & 0 & 0 & 0 & X_w^n & Y_w^n & Z_w^n & 1 & -y^nX_w^n & -y^nY_w^n & -y^nZ_w^n & -y^n
\end{bmatrix}
\cdot
\begin{bmatrix}
m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34}
\end{bmatrix}
= \mathbf{0}, \tag{8}
$$

where the first matrix is with size $2n \times 12$ (n is the number of available points).
We apply Eq. (9) to calculate $\mathbf{m} = \{m_{ij}\}$. Stacking n correspondences yields a homogeneous linear system $\mathbf{A}\,\mathbf{m} = 0$ of size $2n \times 12$ for the stacked projection parameters $\mathbf{m} = \text{vec}(M)$. This system determines $\mathbf{m}$ only up to a nonzero scalar i.e. $\mathbf{x} = \lambda\mathbf{m}$. $\mathbf{x}$ denotes the direction of $\mathbf{m}$, where $\|\mathbf{x}\| = 1$. If we know the values of λ and $\mathbf{x}$, we can get $\mathbf{m}$. To obtain the value of $\mathbf{x}$, we apply Eq. (9). To obtain the value of λ, we have $||\mathbf{m}_3|| = ||\frac{1}{\lambda}\mathbf{x}_3|| = 1$.

Preliminary.3.

Solve the homogeneous linear equation $\mathbf{Ax} = 0$, where $\mathbf{x}$ is the vector of N unknowns, and $\mathbf{A}$ is the matrix of $M \times N$ coefficients. A quick observation is that there are infinite solutions for $\mathbf{Ax} = 0$, since we can randomly scale x with a scalar λ such that $\mathbf{A}(\lambda\mathbf{x}) = 0$. Therefore, we assume $\|\mathbf{x}\| = 1$. Solving the equation can be approximated to

$$
\min \|\mathbf{Ax}\| \tag{9}
$$

The minimization problem can be solved with Singular Value Decomposition (SVD). Assume that $\mathbf{A}$ can be decomposed to $\mathbf{U\Sigma V}^T$, we have

$$
\min \|\mathbf{Ax}\| = \|\mathbf{U\Sigma V}^T\mathbf{x}\| = \|\mathbf{\Sigma V}^T\mathbf{x}\|. \tag{10}
$$

Note that $\|\mathbf{V}^T\mathbf{x}\| = \|\mathbf{x}\| = 1$, then let $\mathbf{y} = \mathbf{V}^T\mathbf{x}$, so we have

$$
\begin{aligned}
\min \|\mathbf{Ax}\| &= \|\mathbf{\Sigma y}\| \\
&= \left\| \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ 0 & \cdots & 0 \\ \vdots & \vdots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \right\|,
\end{aligned} \tag{11}
$$

where $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$. Recall that $\|\mathbf{y}\| = 1$, we can set

$$
\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix}. \tag{12}
$$

So $\mathbf{x}$ should be the last row of $\mathbf{V}^T$.

Question

Figure 2 shows an image of the checkerboard, where XYZ is the world coordinate and xy is marked as the image coordinate. The edge length of each grid on the checkerboard is $10mm$ in reality. You can calculate the projection matrix from world coordinate to image coordinate based on the 32 corners (marked points) on the checkerboard (16 corners in each side of the checkerboard). From the projection matrix you can get the intrinsic matrix which is indicated as $\begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$ (f_x and f_y need not to be equal).

- Design a function to get the image coordinates of the 32 corners from the image. You can decide the order of output points for yourself. (2 point)
- Manually (or design a program) to get the world coordinate of the 32 corners. Note that the output order should be the SAME with the last question. (1 point)
- Design a function to get the intrinsic parameters f_x, f_y, o_x, o_y from the image coordinates and world coordinates acquired above. (2 points)
- Design a function to get the extrinsic parameters R, T from the image coordinates and world coordinates acquired above. (1 points)

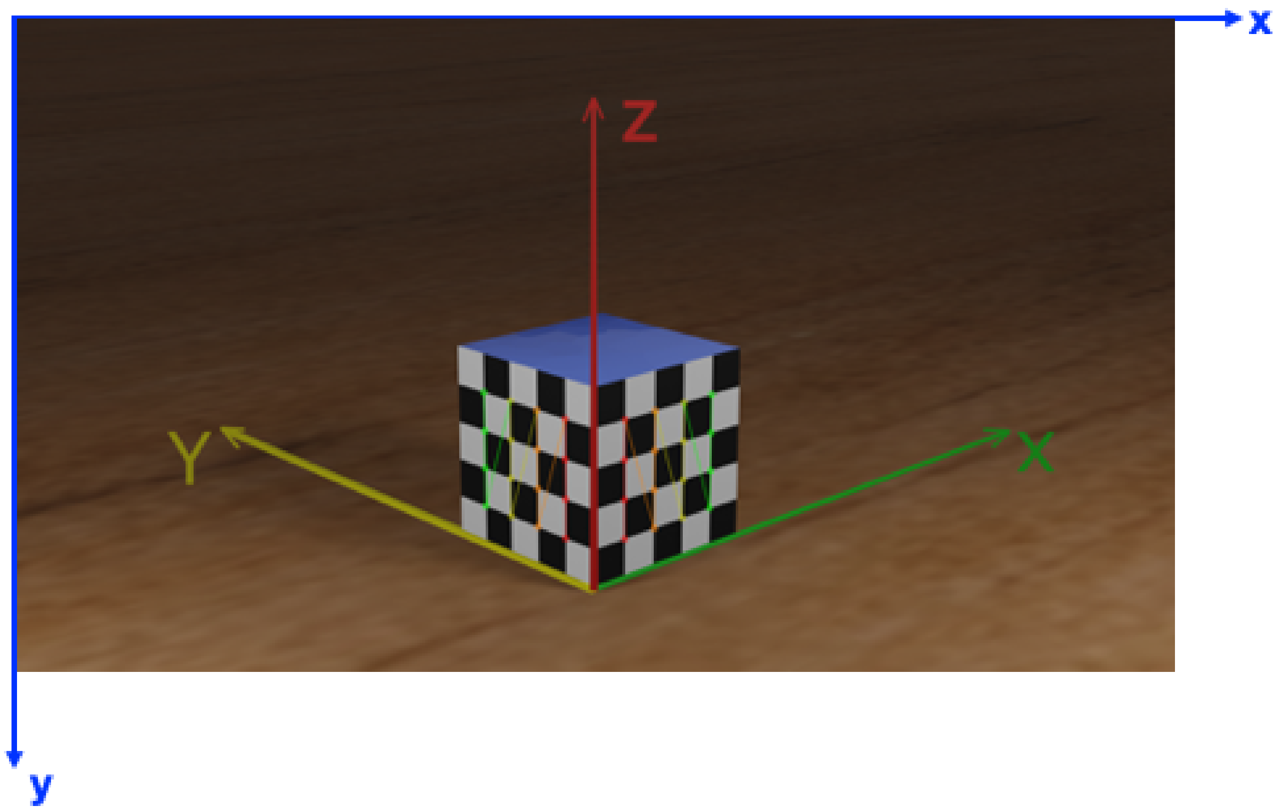

Figure 2: Image of the checkerboard

Instructions (Please read this very carefully!):

- You are only allowed to use OpenCV version 4.5.4 for this project. Please check requirements.txt.
- Please implement all your code in file **"UB_Geometry.py"**. Please do NOT make any changes to any file except **"UB_Geometry.py"**.
- To submit your code and result, Please run **"pack_submission.sh "** to pack your code and result into a zip file. You can find the command line in **"README.md"** Note that when packing your submission, the script would run your code before packing. The resulting zip file is the only file you need to submit. You should upload the resulting zip file in UBlearns.
- The packed submission file should be named **"submission_<** *Your_UBIT_name* **>.zip"**, and it should contain 3 files, named **"result_task1.json"**, **"result_task2.json"**, and **"UB_Geometry.py"**. If not, there is something wrong with your code/filename, please go back and check.
- You are ONLY allowed to use the libraries that are already imported in the scripts.
- We grade this project based on the results we get from running your code. If you do not give correct final results, you are very likely to get NO partial points for that step, even if it may be because of mistakes in the former parts.
- Late submissions are NOT accepted.
- Anyone whose code raises "RuntimeError", your grade will be 0 for that task.
- Anyone whose code is flagged for plagiarism, your grade will be 0 for this project. Consequently, you will be reported to UB's Academic Integrity office.