3D Vision – Assignment 4

魏家博 (Chia-Po Wei)

Department of Electrical Engineering

National Sun Yat-sen University

[TODO 1] Normalize image points.

1. The array `pts1` of shape (n, 2) denotes the image points in image1. Let n = 10.
2. Normalize `pts1` to `pts1_normal` of shape (n, 2) such that
   - The mean of image points (denoted by `pts1_normal`) is the origin.
   - The mean distance of the image points (denoted by `pts1_normal`) from the origin is $\sqrt{2}$.
3. In a similar fashion, normalize `pts2` to `pts2_normal` of shape (n,2).
4. Construct $\mathbf{T}_1 = \begin{bmatrix} s & 0 & -st_x \\ 0 & s & -st_y \\ 0 & 0 & 1 \end{bmatrix}$, where s, t_x, t_y are defined below.
5. In a similar fashion, construct $\mathbf{T}_2$ by replacing s, t_x, t_y with s', t'_x, t'_y, where s', t'_x, t'_y are from `pts2`.

- Hint:
  - Let $\mathbf{p}_i \in \mathbb{R}^2$ be the ith point in `pts1`.
  - Denote the mean of image points by $\overline{\mathbf{p}}$, which is given by $\overline{\mathbf{p}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{p}_i$. Let $\overline{\mathbf{p}} = \left[t_x, t_y\right]^T$.
  - Define $\mathbf{q}_i = \mathbf{p}_i - \overline{\mathbf{p}}$.
  - Find s such that $\frac{s\sum_{i=1}^{n}\|\mathbf{q}_i\|}{n} = \sqrt{2}$, where $\|\mathbf{q}_i\| = \sqrt{x_i^2 + y_i^2}$ if $\mathbf{q}_i = [x_i, y_i]^T$.
  - The ith point in `pts1_normal` is given by $s\mathbf{q}_i$.

[TODO 1]

6. Display the mean distance of the image points in `pts1_normal` from the origin. That is, display the value of $\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{r}_i\|$, where $\mathbf{r}_i$ is the ith point of `pts1_normal`.

[TODO 2] Compute the fundamental matrix $\mathbf{F}$.

1. Construct array A of shape (n,9) from `pts1_normal` and `pts2_normal` as in page 13 of 08 Two-View Geometry.pdf.
2. Implement the 8-point algorithm presented in page 14 of 08 Two-View Geometry.pdf to compute the fundament matrix $\mathbf{F}_n$, and $\mathbf{F}_n$ should have rank two.
3. Let $\mathbf{F} = \mathbf{T}_2^T \mathbf{F}_n \mathbf{T}_1$, where $\mathbf{T}_1$ and $\mathbf{T}_2$ are defined in the previous slide.
4. Display the epipolar error:
   - $\sum_{i=1}^{n}[x_i', y_i', 1]\ \mathbf{F}\ [x_i, y_i, 1]^T$
   - where $[x_i, y_i]$ is the ith point of `pts1`, and $[x_i', y_i']$ is the ith point of `pts2`.
   - You can check that the epipolar error should be less than one.
     - If this is not the case, then there is something wrong with your program.

[TODO 3] Draw epipolar lines in image1.

1. Let p_1', p_2', p_3' be the first three points of `pts2`. Plot p_1', p_2', p_3' in image2 as in Figure 2.
2. Plot the epipolar lines corresponding to p_1', p_2', p_3' in image1 as in Figure 1.

Hint:

- The epipolar line corresponding to p_1' is $\mathbf{F}^T p_1' = [a, b, c]^T$.
- Any point (x, y) on the epipolar line satisfies $ax + by + c = 0$.
- Select suitable x and then determine y from the above epipolar line equation.

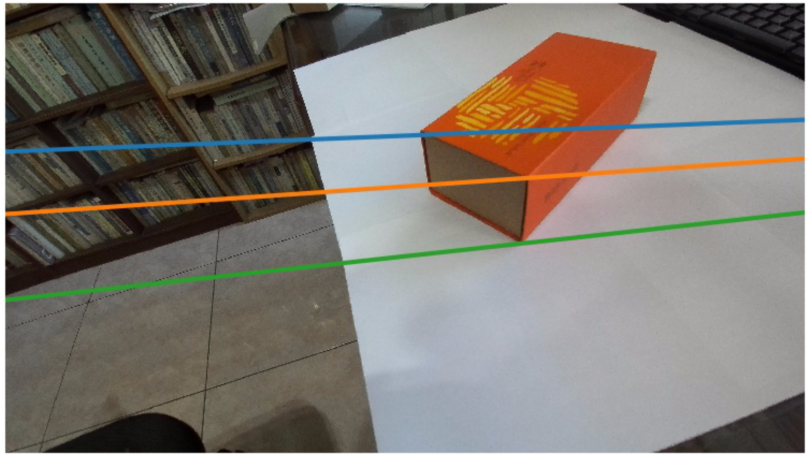

Figure 1: left image

Figure 2: right image

[TODO 4] Draw the epipole and epipolar lines in image1.

1. Compute the epipole in image1. The epipole is a nonzero vector in the null space of $\mathbf{F}$.
2. Plot the epipole and epipolar lines in image1 as shown in Figure 3.

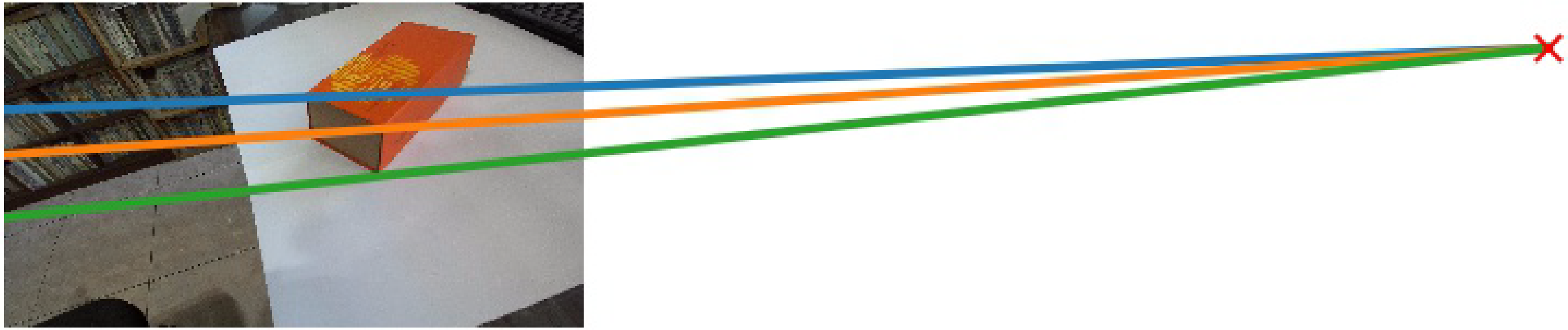

Figure 3

[TODO 5] Compute the 4 possible camera matrices.

1. Let $\mathbf{K}_1$ and $\mathbf{K}_2$ be intrinsic matrices for cameras 1 and 2, respectively.
2. Let $\mathbf{E} = \mathbf{K}_2^T \mathbf{F} \mathbf{K}_1$.
3. Let `Rt2_list` be a list of extrinsic matrices for camera 2 defined in page 22 of 08 Two-View Geometry.pdf
    - `Rt2_list[0]` is $[\mathbf{R}_1, \mathbf{t}_1]$, `Rt2_list[1]` is $[\mathbf{R}_1, \mathbf{t}_2]$, `Rt2_list[2]` is $[\mathbf{R}_2, \mathbf{t}_1]$, `Rt2_list[3]` is $[\mathbf{R}_2, \mathbf{t}_2]$

[TODO 6] Triangulation.

1. Let $[x_i, y_i]$ be the ith point of `pts1`, and $[x_i', y_i']$ be the ith point of `pts2`, and `X_list = []`

`2. for j in range(4):`

- Let $\mathbf{P}_1 = \mathbf{K}_1[\mathbf{I}, \mathbf{0}]$, and $\mathbf{P}_2 = \mathbf{K}_2[\mathbf{R}, \mathbf{t}]$, where $[\mathbf{R}, \mathbf{t}]$ is `Rt2_list[j]`.
- `for i in range(n):`
    - Given $\mathbf{P}_1$, $\mathbf{P}_2$, $[x_i, y_i]$, $[x_i', y_i']$, implement the triangulation method in page 8 of 08 Two-View Geometry.pdf to obtain $[X_i, Y_i, Z_i]$.
    - Save $[X_i, Y_i, Z_i]$ in the ith row of `X`, where `X` is an array of shape (n,3).
- `X_list.append(X)`

[TODO 6] Triangulation.

3. Find index `j` such that `X_list[j][:,2]` are all positive, and denote `X_list[j]` by `X_best`.
4. Display the reconstruction error for `pts1` and `pts2`, respectively.
   - Let $\mathbf{P}_1$ and $\mathbf{P}_2$ be defined as in Step 2 of TODO 6, where `j` is from Step 3 of TODO 6.
   - Let $\mathbf{M}_i \in \mathbb{R}^3$ be the ith point of `X_best`, and define
     - $\widetilde{\mathbf{M}}_i = \begin{bmatrix} \mathbf{M}_i \\ 1 \end{bmatrix} \in \mathbb{R}^4$, $\widetilde{\mathbf{m}}_i = \mathbf{P}_1\widetilde{\mathbf{M}}_i$, $\widetilde{\mathbf{m}}_i' = \mathbf{P}_2\widetilde{\mathbf{M}}_i$
     - Convert $\widetilde{\mathbf{m}}_i \in \mathbb{R}^3$ to $\mathbf{m}_i \in \mathbb{R}^2$ as in Step 3.3 of Assignment 1 (Convert homogeneous coordinates to Cartesian coordinates).
     - Similarly, Convert $\widetilde{\mathbf{m}}_i' \in \mathbb{R}^3$ to $\mathbf{m}_i' \in \mathbb{R}^2$.
   - Let $\mathbf{p}_i \in \mathbb{R}^2$ be the ith point of `pts1`, and $\mathbf{p}_i' \in \mathbb{R}^2$ be the ith point of `pts2` .
   - The reconstruction error for `pts1` is given by $\frac{\sum_{i=1}^{n}\|\mathbf{p}_i - \mathbf{m}_i\|}{n}$.
   - The reconstruction error for `pts2` is given by $\frac{\sum_{i=1}^{n}\|\mathbf{p}_i' - \mathbf{m}_i'\|}{n}$.
   - Note that the two norm of $\mathbf{d}_i = [x_i, y_i]^T \in \mathbb{R}^2$ is given by $\|\mathbf{d}_i\| = \left(x_i^2 + y_i^2\right)^{1/2}$.

[TODO 7] Visualize the 3D reonstruction.

1. Plot the 3D points `X_best` from Step 3 of TODO 6 as in Figures 5 and 6.
2. You need to draw the nine edges: (v0, v1), (v1, v2), (v2, v3), (v3,v0), (v4, v5), (v5, v6), (v0, v4), (v1, v5), (v2, v6), where the vertex number is defined as in Figure 4.

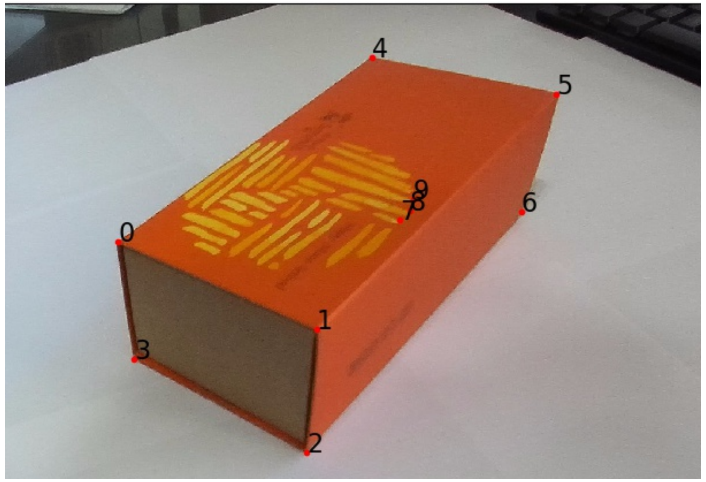

Figure 4

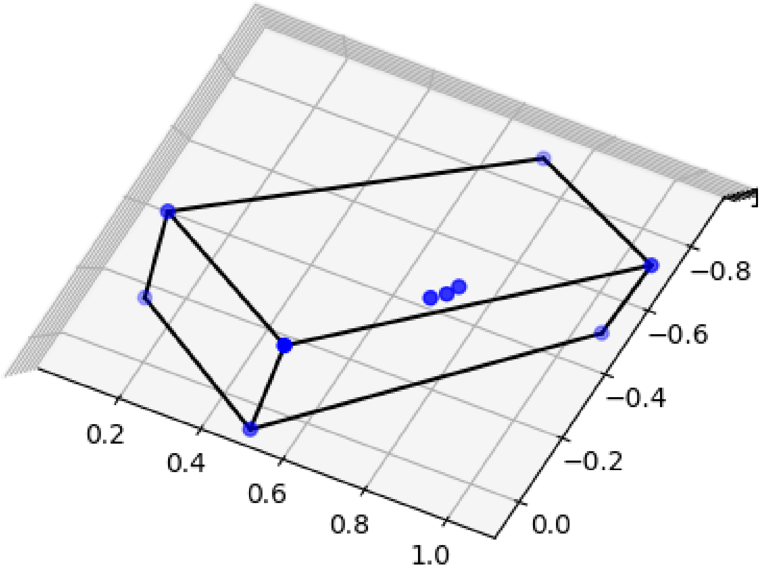

Figure 5:
ax.view_init(elev=-90, azim=-115)

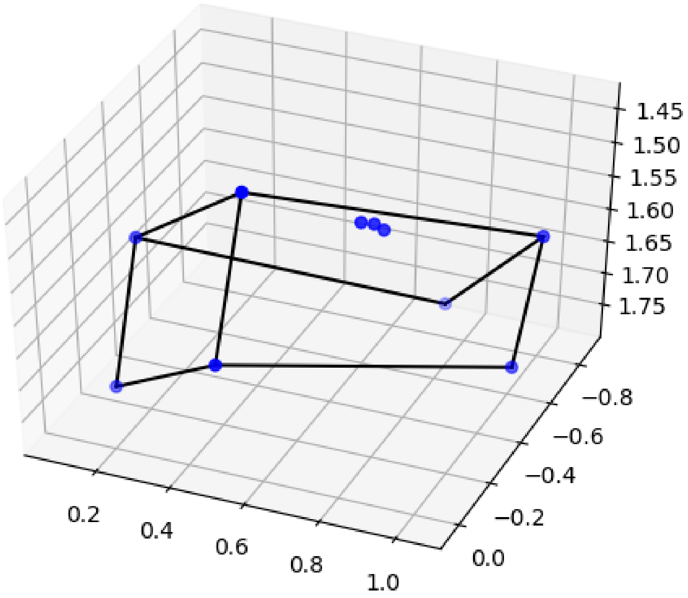

Figure 6:
ax.view_init(elev=-139, azim=-113)