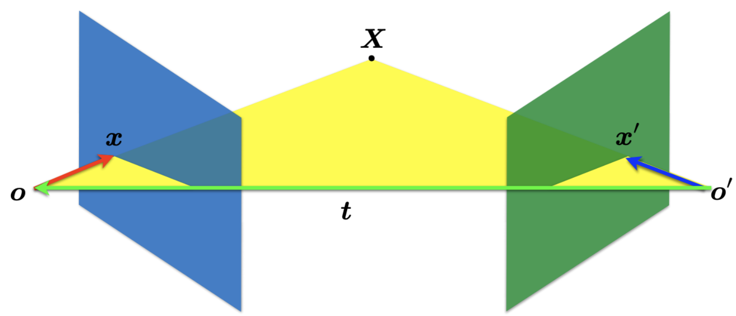
Homework 7: Introduction to SLAM
CSE 455: Computer Vision

In order to fully achieve simultaneously localization and mapping, we need to constantly calculation the movement of the camera and the map of our environment at the same time. Thus understanding the decomposition of the essential matrix into rotation and translation and using triangulation to locate points in 3D becomes the key to understand MonoSLAM systems. We will practice calculating them by hand.

1 Theoretical Part

1.1 Essential Matrix Review (*5pts*)

Before we dive into decomposing the essential matrix, let's review what essential matrix is and what does it encode. Let x be the point from one frame, x' be the corresponding point in the next frame, and t be the translation vector



of the camera. We can tell from the graph that these three vectors are coplanar.
(i) Given this information, calculate:

$$x^T(t \times x)$$

(ii) Recall that addition and subtraction of two coplanar vectors are still coplanar. Compute:

$$(x-t)^T(t \times x)$$

(iii) Now given the rigid motion of the camera, i.e. the motion consists of rotation and translation, we have $x' = \mathbf{R}(x-t)$, $\mathbf{R}$ being the rotation matrix. Compute $(x-t)^T$ based on this given equation.

Plug in the answer to the coplanarity constrain, we have $(x'^T\mathbf{R})(t \times x) = 0$.

Let's write the cross product in a matrix multiplication form:

$$a \times b = [a]_{\times} b = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

(iv) Verify that this is equal to compute the cross product directly.
Thus we have $(x'^T \mathbf{R})(t \times x) = (x'^T \mathbf{R})([t_{\times}]x) = x'^T(\mathbf{R}[t_{\times}])x = 0$. We call this matrix $\mathbf{R}[t_{\times}]$ the essential matrix.

Answer:
(i) & (ii) $x^T(t \times x) = (x - t)^T(t \times x) = 0$ because the dot product of two orthogonal vectors is 0.
(iii) $x' = \mathbf{R}(x - t)$. $\mathbf{R}^{-1}x' = (x - t)$. We use the fact that the transpose of the rotation matrix is the same as its inverse, $\mathbf{R}^T x' = (x - t)$. $(x - t)^T = x'^T \mathbf{R}$
(iv) $a \times b = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix}$

1.2 Understanding the Decomposition of Essential Matrix (*10pts*)

We decompose the essential matrix by using Singluar Value Decomposition. We have derived that $\mathbf{E} = [t_\times]\mathbf{R}$. Let's call these two matrices $\mathbf{S}$ and $\mathbf{R}$. Now we can start the decomposition process.
$\mathbf{E} = \mathbf{U}(diag)\mathbf{V}^T$
Given the coplanar and rigid transformation, we fill out the diagonal matrix with values $[1, 1, 0]$. In order to decompose the matrix into two parts, let's assume the diagonal matrix can be divided into two parts: $\mathbf{Z}$ and $\mathbf{W}$.

$$\mathbf{E} = \mathbf{U}diag(1, 1, 0)\mathbf{V}^T = \mathbf{U}(\mathbf{ZW})\mathbf{V}^T = (\mathbf{UZU}^T)(\mathbf{UWV}^T)$$

Here are two matrices $\mathbf{Z}$ and $\mathbf{W}$ that have this property.
$\mathbf{Z} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
For $\mathbf{E}$ there is another solution where $\mathbf{E} = -\mathbf{U}(-diag(1, 1, 0))\mathbf{V}^T$.
We find that $-(diag(1, 1, 0)) = \mathbf{ZW}^T$. Thus we have $\mathbf{E} = -\mathbf{U}(\mathbf{ZW}^T)\mathbf{V}^T = (-\mathbf{UZU}^T)(\mathbf{UW^TV}^T)$
We have successfully extracted the rotation matrix! It can take one two possible values $\mathbf{UWV}^T$ or $\mathbf{UW^TV}^T$. We will use the triangulation method we are going to learn in the next question to determine which one is the correct one. But how do we get the translation vector?
First we know that the matrix $\mathbf{S}$ is the matrix multiplication form of the cross product of t. Recall that $t \times t = 0$. So we have $[t_\times]t = 0$. $\mathbf{S}t = 0$. We can expand the formula for $\mathbf{S}$ and solve for t. Without complicating the calculation too much, we assume the answer $t = \mathbf{U}_3$ is given.

So far, we have completely decomposed the essential matrix and fully extract all possible rotations and translations from it. It can in total take on these four values:
$[\mathbf{UWV^T}|u_3]$, $[\mathbf{UW^TV^T}|u_3]$, $[\mathbf{UWV^T}|-u_3]$, $[\mathbf{UW^TV^T}|-u_3]$.

(i) Now, given the essential matrix:

$$\mathbf{E} = \begin{bmatrix} -\frac{5}{2} & 0 & \frac{5\sqrt{3}}{2} \\ \frac{5}{2} & 0 & -\frac{5\sqrt{3}}{2} \\ -\frac{5\sqrt{3}}{2} & 5 & -\frac{5}{2} \end{bmatrix}$$

Use your favorite programming language and SVD libraries to solve for all possible rotation matrices and translation vectors.

Answer:

(i) We first send the matrix to a SVD solver. We recover the translation vector by looking at the last column of matrix $\mathbf{U}$. $t = \begin{bmatrix} \frac{1}{\sqrt{2}} \cdot 5\sqrt{2} \\ \frac{1}{\sqrt{2}} \cdot 5\sqrt{2} \\ 0 \cdot 5\sqrt{2} \end{bmatrix} = \begin{bmatrix} 5 \\ 5 \\ 0 \end{bmatrix}$
Then we recover the rotation matrix by computing the $\mathbf{UWV^T}$ and $\mathbf{UW^TV^T}$. We get the rotation of 30 degrees around the y axis.

$$M = \begin{pmatrix} -\frac{5}{2} & 0 & \frac{5\sqrt{3}}{2} \\ \frac{5}{2} & 0 & -\frac{5\sqrt{3}}{2} \\ -\frac{5\sqrt{3}}{2} & 5 & -\frac{5}{2} \end{pmatrix}$$

$$U = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{-\frac{5}{2\sqrt{3}} - \frac{5\sqrt{3}}{2}}{\sqrt{2}\left(\frac{5}{2\sqrt{3}} + \frac{5\sqrt{3}}{2}\right)} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 5\sqrt{2} & 0 & 0 \\ 0 & 5\sqrt{2} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$V = \begin{pmatrix} -\frac{1}{2} & -\frac{\sqrt{\frac{3}{2}}}{2} & \frac{\sqrt{\frac{3}{2}}}{2} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{\sqrt{3}}{2} & -\frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} \end{pmatrix}$$

1.3 Triangulation Step-by-Step (*20pts*)

After we computed the rotation and translation of the camera from the previous frame to the current one, we can simply recover the camera's pose by multiplying the transformation matrices. Now the problem becomes, given a set of matched points $\{x, x'\}$, and camera matrices $\mathbf{P}, \mathbf{P}'$, estimate the corresponding point in 3D.
First let's write the equation that describe the mapping from a 3D point to a 2D point: $x = \mathbf{P}\mathbf{X}$.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \cdot \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(i) Explain why there is a scalar factor α.

We solve the equation by using a technique called Direct Linear Transform. We will practice deriving the math of it from scratch.
First let's recall $x \times x = 0$, that is, the cross product of two same vectors is 0. Also remember $x \times \alpha x = \alpha x \times x = 0$, that is, scaling a vector doesn't change its direction.

(ii) Write the equation above $x = \mathbf{P}\mathbf{X}$ in the form of a cross product of two vectors.

(iii) Use what you know about calculating the cross product to solve the answer you got for the previous question.

(iv) Remember we have not just one but two points. Plug the other point into the equation as well. The final equation should be in the form of $\mathbf{A}x = b$.

After we have this equation, we can easily solve it by using any of the least squares solver.

Answer:

(i) There are an infinite amount of points on the same ray that will be mapped into the same point in the 2D plane.

(ii) $x \times \mathbf{P}\mathbf{X} = 0$

(iii) $x = \alpha \mathbf{P}\mathbf{X}$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \alpha \begin{bmatrix} \mathbf{p}_1^T\mathbf{X} \\ \mathbf{p}_2^T\mathbf{X} \\ \mathbf{p}_3^T\mathbf{X} \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_1^T\mathbf{X} \\ \mathbf{p}_2^T\mathbf{X} \\ \mathbf{p}_3^T\mathbf{X} \end{bmatrix} = \begin{bmatrix} y\mathbf{p}_3^T\mathbf{X} - \mathbf{p}_2^T\mathbf{X} \\ \mathbf{p}_1^T\mathbf{X} - x\mathbf{p}_3^T\mathbf{X} \\ x\mathbf{p}_2^T\mathbf{X} - y\mathbf{p}_1^T\mathbf{X} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Third equation is just a linear combination of the first and second lines.

Thus we have $\begin{bmatrix} y\mathbf{p}_3^T\mathbf{X} - \mathbf{p}_2^T\mathbf{X} \\ \mathbf{p}_1^T\mathbf{X} - x\mathbf{p}_3^T\mathbf{X} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

(iv) $$\begin{bmatrix} y\mathbf{p}_3^T\mathbf{X} - \mathbf{p}_2^T\mathbf{X} \\ \mathbf{p}_1^T\mathbf{X} - x\mathbf{p}_3^T\mathbf{X} \\ y'\mathbf{p'}_3^T\mathbf{X} - \mathbf{p'}_2^T\mathbf{X} \\ \mathbf{p'}_1^T\mathbf{X} - x'\mathbf{p'}_3^T\mathbf{X} \end{bmatrix} = \begin{bmatrix} y\mathbf{p}_3^T - \mathbf{p}_2^T \\ \mathbf{p}_1^T - x\mathbf{p}_3^T \\ y'\mathbf{p'}_3^T - \mathbf{p'}_2^T \\ \mathbf{p'}_1^T - x'\mathbf{p'}_3^T \end{bmatrix} \mathbf{X} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$\mathbf{A}x = 0$
