Comprehensive Guide to 2D-3D Geometry and Camera Projection

David Padilla Orenga, Nacho Pastore

September 2024

1. Camera Projection Model and Projection Matrix

The camera projection model describes how a 3D point in the world is projected onto a 2D image plane. This is done via the **projection matrix** P.

Projection Equation

To project a 3D world point $\mathbf{X} = (X, Y, Z, 1)^\top$ to a 2D image point $\mathbf{x} = (x, y, 1)^\top$, we use:

$$\mathbf{x} = P\mathbf{X}, \quad P \in \mathbb{R}^{3\times 4}$$

Where P is the **projection matrix** and is formed by the camera's intrinsic and extrinsic parameters:

$$P = K[R|t]$$

Where:

- K: **Intrinsic matrix** (3×3), defines camera-specific parameters such as focal length.
- R: **Rotation matrix** (3×3), describes the camera's orientation in the world.
- t: **Translation vector** (3×1), describes the camera's position in the world.

Intrinsic Matrix Example

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Where f_x, f_y are the focal lengths and c_x, c_y are the coordinates of the principal point.

Tip: Projection Matrix Construction
Given a transformation matrix $T_{w_{c1}}$ (4×4) for camera 1's pose:

$$T_{w_{c1}} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}, \quad R = \text{rotation}, \, t = \text{translation}$$

The inverse T_{c1_w} for the world-to-camera transformation is:

$$T_{c1_w} = T_{w_{c1}}^{-1} = \begin{bmatrix} R^\top & -R^\top t \\ 0 & 1 \end{bmatrix}$$

This technique is essential for changing coordinate frames.

2. Epipolar Geometry and Fundamental Matrix

Epipolar geometry defines the geometric relationship between two camera views of the same scene. The **fundamental matrix** F describes this relationship between corresponding points in two images.

Fundamental Matrix F

The fundamental matrix satisfies the **epipolar constraint**:

$$\mathbf{x'}^\top F\mathbf{x} = 0$$

Where:

- $\mathbf{x} = (x, y, 1)^\top$: Point in the first image.
- $\mathbf{x}' = (x', y', 1)^\top$: Corresponding point in the second image.
- F: Fundamental matrix (3×3).

8-Point Algorithm for F

The **8-point algorithm** estimates F given at least 8 pairs of corresponding points:

$$A\mathbf{f} = 0, \quad A \in \mathbb{R}^{N\times 9}, \quad \mathbf{f} = \text{flatten}(F)$$

Solve this using SVD, and then enforce the **rank-2 constraint**:

$$U, S, Vt = \text{SVD}(F), \quad S[-1] = 0, \quad F = U \cdot S \cdot Vt$$

Epipolar Line

Given a point $\mathbf{x}$ in the first image, the corresponding epipolar line l_2 in the second image is computed as:

$$l_2 = F\mathbf{x}$$

The epipolar line $l_2 = [a, b, c]^\top$ represents the line equation $ax + by + c = 0$. To plot the line, use the image's width to find the x coordinates and solve for the corresponding y values:

$$y = -\frac{ax + c}{b}$$

Tip: Computing the Epipole The **epipole** is the point where all epipolar lines intersect. It lies in the null space of F. Compute the epipoles as:

$$e_2 = \text{null space}(F), \quad e_1 = \text{null space}(F^\top)$$

Use SVD:

$$U, S, Vt = \text{SVD}(F), \quad e_2 = Vt[-1], \quad e_1 = U[:, -1]$$

Normalize to homogeneous coordinates.

3. Essential Matrix and Decomposition

The **essential matrix** is a calibrated version of the fundamental matrix that relates points between two cameras with known intrinsic parameters:

$$E = K_2^\top F K_1$$

Where K_1 and K_2 are the intrinsic matrices of the two cameras.

Decomposing E into Rotation and Translation

The essential matrix can be decomposed into **rotation** and **translation** using SVD:

$$E = U\Sigma V^\top$$

From this, we extract two possible rotations R_1, R_2 and two possible translations $t, -t$:

$$R_1 = UWV^\top, \quad R_2 = UW^\top V^\top, \quad t = U[:, 2]$$

Where:

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

This gives four possible camera poses. To find the correct pose, we check which configuration places the triangulated points in front of both cameras (i.e., with positive depth).

Tip: Recovering Pose from E
Use the essential matrix E to recover the relative pose (rotation R and translation t) between two cameras. You can decompose E to obtain two possible solutions for each. Use triangulation to validate which pose is correct.

4. Triangulation of 3D Points

Triangulation allows us to recover the 3D position of a point given its 2D projections in two images and the corresponding projection matrices P_1 and P_2.

Triangulation Algorithm

Given two projection matrices P_1 and P_2 and corresponding points $(\mathbf{x}_1, \mathbf{x}_2)$ in two images:

$$A\mathbf{X} = 0$$

Where:

$$A = \begin{bmatrix} x_1 P_3^1 - P_1^1 \\ y_1 P_3^1 - P_2^1 \\ x_2 P_3^2 - P_1^2 \\ y_2 P_3^2 - P_2^2 \end{bmatrix}, \quad \mathbf{X} = (X, Y, Z, 1)^\top$$

Solve for $\mathbf{X}$ using SVD:

$$\mathbf{X} = Vt[-1], \quad \mathbf{X}/ = \mathbf{X}[-1]$$

The result $\mathbf{X}$ gives the 3D coordinates in homogeneous form.

Projection of 3D Points

To project a 3D point $\mathbf{X}$ onto an image:

$$\mathbf{x} = P\mathbf{X}$$

Where P is the projection matrix and $\mathbf{X}$ is the 3D point in homogeneous coordinates.

Tip: Using the Projection Matrix
For triangulation using two images:

- Compute the projection matrices $P_1 = K_1[R_1|t_1]$ and $P_2 = K_2[R_2|t_2]$.
- Solve $A\mathbf{X} = 0$ to get the 3D coordinates $\mathbf{X}$ of the point.

5. Direct Linear Transform (DLT)

The **Direct Linear Transform (DLT)** algorithm estimates the homography matrix H_{21} that maps points from one image to another. Given N corresponding points $(\mathbf{x}_1, \mathbf{x}_2)$ between two images:

$$A\mathbf{h} = 0, \quad A \in \mathbb{R}^{2N\times 9}, \quad \mathbf{h} = \text{flatten}(H)$$

Construct A from point correspondences, and solve using SVD:

$$H = Vt[-1].reshape(3,3), \quad H/ = H[2,2]$$

Inverse and Transpose of Homography

To find the inverse homography H_{12} (mapping points back to the first image), compute the inverse:

$$H_{12} = H_{21}^{-1}$$

You **cannot** simply transpose the homography matrix H to invert it because it is not an orthogonal matrix.

6. RMSE for Homography Estimation

Once the homography matrix H_{21} is estimated, the RMSE (Root Mean Square Error) between the projected points and ground-truth points is calculated to evaluate accuracy.

RMSE Formula for Homography

To compute RMSE for homography:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} ||\mathbf{x}'_i - \hat{\mathbf{x}}_i||^2}$$

Where:

- $\mathbf{x}'_i$: Actual point in the second image.
- $\hat{\mathbf{x}}_i = H_{21}\mathbf{x}_i$: Projected point using homography.
- N: Number of point correspondences.

7. Summary of Key Equations and Tips

- **Projection Equation**: $\mathbf{x} = P\mathbf{X}$.
- **Fundamental Matrix Constraint**: $\mathbf{x}'^{\top} F\mathbf{x} = 0$.
- **Essential Matrix from** F: $E = K_2^{\top} F K_1$.
- **Triangulation Equation**: $A\mathbf{X} = 0$, where A is formed from projection matrices and image points.
- **Homography Estimation**: H_{21} from $A\mathbf{h} = 0$ using DLT.
- **Decomposing** E: $E = U\Sigma V^{\top}$ to get R_1, R_2, t.