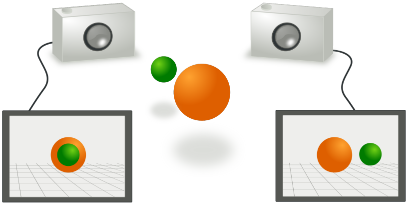
3D Machine Vision

Epipolar Geometry

Technical University of Applied Sciences Würzburg-Schweinfurt

Basics of Epipolar Geometry



3D Reconstruction from One View

Recovery of depth from one image only is inherently ambiguous

Stereo Vision

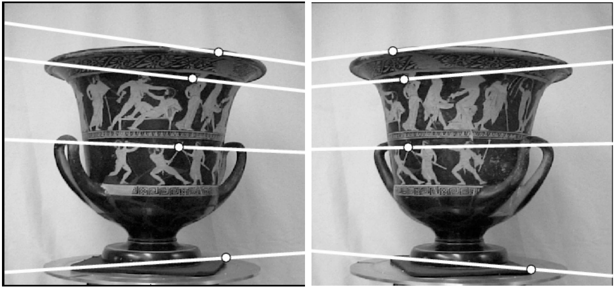
3D Reconstruction from One View

Recovery of depth from one image only is inherently ambiguous

Stereo Vision

3D Reconstruction from Two Views

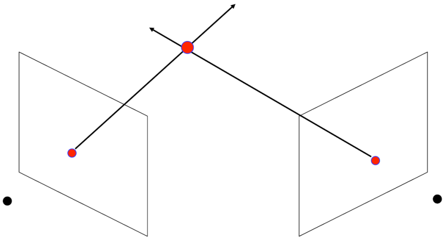
Now, we have a look at geometric methods that rely on two views



Triangulation

Requirements

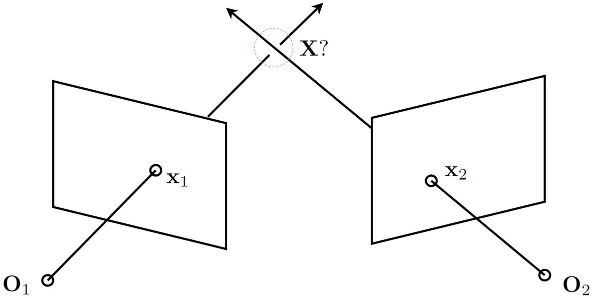
- Intersect two rays originating from the same point in the scene
- Requires 2D correspondences (knowledge which pixels are images of the same 3D point)
- Requires camera pose (in order to construct the 3D rays)



Triangulation

Goal

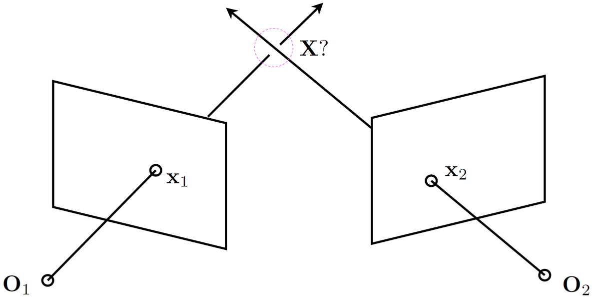
Given projections of a 3D point in two images (with known projection matrices), find the coordinates of the point



Triangulation

Strategy

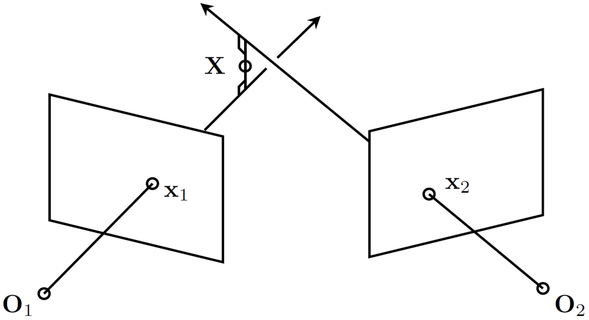
- We want to intersect the two viewing rays corresponding to $\mathbf{x}_1$ and $\mathbf{x}_2$
- Because of noise and numerical errors, they do not meet exactly
- Any ideas?



Triangulation

Geometric Construction

- Find shortest segment connecting the two viewing rays and let **X** be the midpoint of that segment
- find the segment direction (normal to both rays)
- construct 2 planes each containing the segment and one ray
- intersect planes with *other* rays to yield segment endpoints
- average points



Triangulation

Algebraic Linear Approach

$$
\begin{aligned}
Z_1 \bar{\mathbf{x}}_1 &= \mathbf{\Pi}_1 \bar{\mathbf{X}} \quad \rightarrow \quad \widehat{\bar{\mathbf{x}}}_1 \mathbf{\Pi}_1 \bar{\mathbf{X}} = \mathbf{0} \\
Z_2 \bar{\mathbf{x}}_2 &= \mathbf{\Pi}_2 \bar{\mathbf{X}} \quad \rightarrow \quad \widehat{\bar{\mathbf{x}}}_2 \mathbf{\Pi}_2 \bar{\mathbf{X}} = \mathbf{0}
\end{aligned}
$$

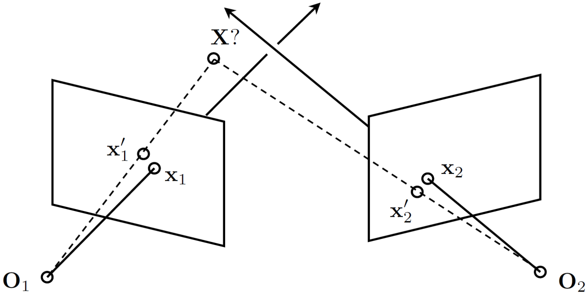
- Two independent equations per image point $\mathbf{x}_{1/2}$ for 3 unknown entries of $\mathbf{X}$
- ... again, a homogeneous overdetermined linear equation system
- ... again, solve with SVD
- Note: directly generalizes to > 2 views, just stack more equations

Triangulation

Nonlinear Approach

Find **X** that minimizes the squared reprojection error:

$$\mathbf{X}^* = \operatorname{argmin}_{\mathbf{X}} \left\| \mathbf{x}_1 - \underbrace{h(\mathbf{\Pi}_1 \overline{\mathbf{X}})}_{\mathbf{x}_1'} \right\|^2 + \left\| \mathbf{x}_2 - \underbrace{h(\mathbf{\Pi}_2 \overline{\mathbf{X}})}_{\mathbf{x}_2'} \right\|^2$$



Triangulation

Nonlinear Approach

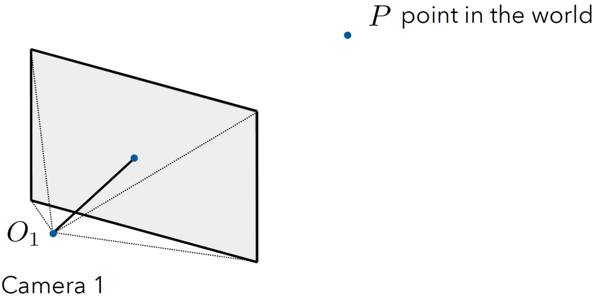
Find $\mathbf{X}$ that minimizes the squared reprojection error:

$$\mathbf{X}^* = \mathrm{argmin}_{\mathbf{X}} \left\| \mathbf{x}_1 - \underbrace{h(\mathbf{\Pi}_1 \overline{\mathbf{X}})}_{\mathbf{x}_1'} \right\|^2 + \left\| \mathbf{x}_2 - \underbrace{h(\mathbf{\Pi}_2 \overline{\mathbf{X}})}_{\mathbf{x}_2'} \right\|^2$$

- The most accurate method, but is more complex than the other two
- With 2 cameras: find roots of a 6th degree polynomial
- With > 2 cameras: initialize with linear estimate, optimize with iterative methods (Gauss-Newton, Levenberg-Marquardt)

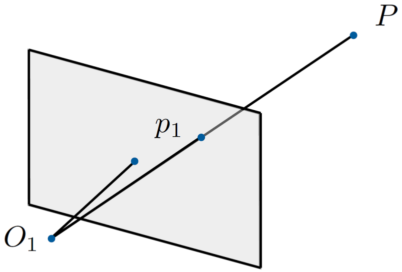
Epipolar Geometry

Step by Step Construction



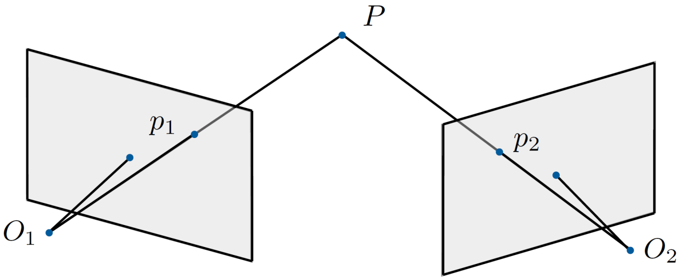
Epipolar Geometry

Step by Step Construction



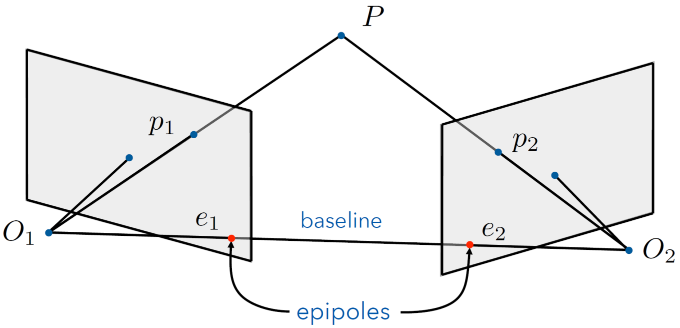
Epipolar Geometry

Step by Step Construction



Epipolar Geometry

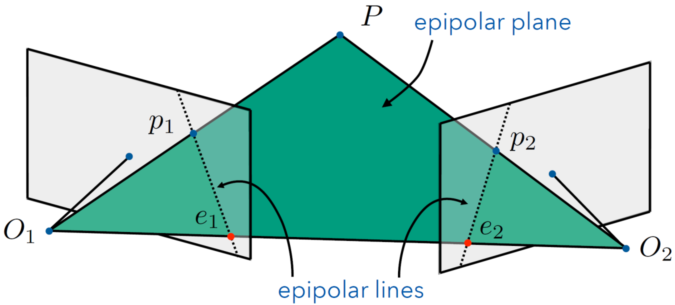
Step by Step Construction



Epipole: Image location of the optical center of the other camera.
Can be outside of the visible area.

Epipolar Geometry

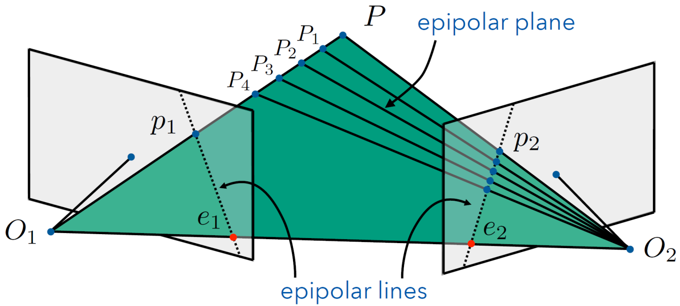
Step by Step Construction



Epipolar plane: Plane through both camera centers and world point.

Epipolar Geometry

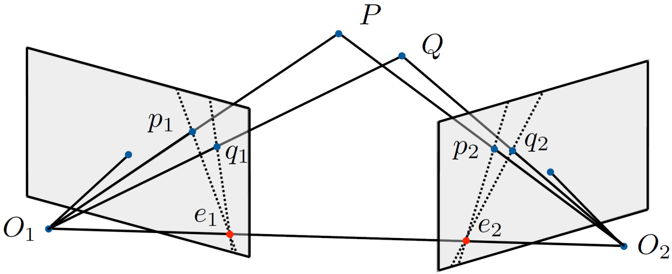
Step by Step Construction



Epipolar line: Constrains the location where a particular feature from one view can be found in the other. Feature Matching is constraint to a line!

Epipolar Geometry

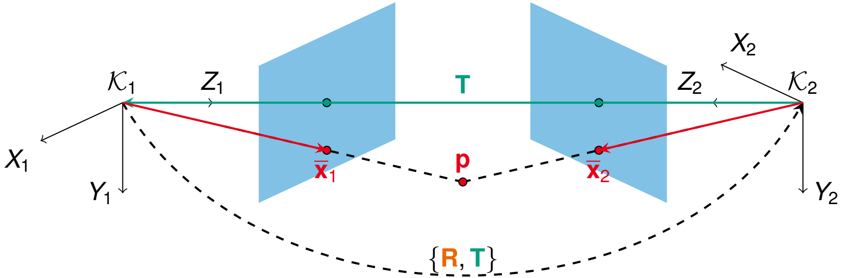
Step by Step Construction



Epipolar lines: Intersect at the epipoles. They are in general not parallel.

Epipolar Geometry

Discrete Modelling - Large Baseline



Looking at the projections of the same static 3D scene from two viewpoints $\mathcal{K}_1, \mathcal{K}_2$ with large baseline, and describing the dependencies between the coordinates $\mathbf{x}_1, \mathbf{x}_2$ of the projections and the relative pose $(\mathbf{R}, \mathbf{T})$ of the viewing angles, then one speaks of discrete epipolar geometry.

Technical University of Applied Sciences Würzburg-Schweinfurt

Discrete Epipolar Constraint

The coordinates of the point $\mathbf{p}$ with respect to the coordinate systems $\mathcal{K}_1$ and $\mathcal{K}_2$ are related by the relative pose $(\mathbf{R}, \mathbf{T})$ in the following way

$$\mathbf{X}_2 = \mathbf{R}\mathbf{X}_1 + \mathbf{T}\,.$$

Replacing these coordinates by the projections $\mathbf{X}_1 = Z_1\bar{\mathbf{x}}_1$ and $\mathbf{X}_2 = Z_2\bar{\mathbf{x}}_2$ we get

$$Z_2\bar{\mathbf{x}}_2 = \mathbf{R}Z_1\bar{\mathbf{x}}_1 + \mathbf{T}\,.$$

The cross product of both sides of this vector equation with $\mathbf{T}$ gives

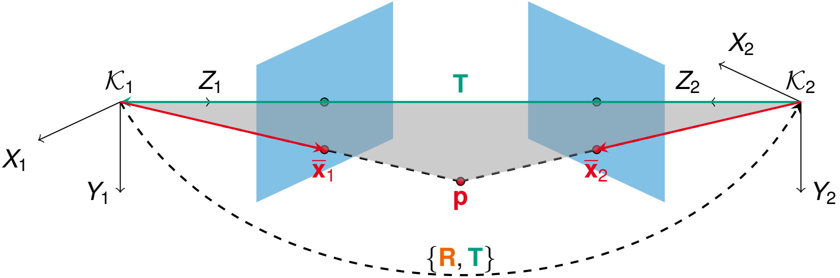
$$Z_2\widehat{\mathbf{T}}\,\bar{\mathbf{x}}_2 = \widehat{\mathbf{T}}\mathbf{R}Z_1\bar{\mathbf{x}}_1\,, \quad \text{because} \quad \widehat{\mathbf{T}}\mathbf{T} = \mathbf{0}\,.$$

Via the scalar product of both sides of the vector equation with $\bar{\mathbf{x}}_2$ one obtains the discrete epipolar constraint

$$0 = \bar{\mathbf{x}}_2^\top\widehat{\mathbf{T}}\mathbf{R}\,\bar{\mathbf{x}}_1\,, \quad \text{because} \quad \bar{\mathbf{x}}_2^\top\widehat{\mathbf{T}}\,\bar{\mathbf{x}}_2 = 0\,, \text{ with the essential matrix } \mathbf{E} := \widehat{\mathbf{T}}\mathbf{R}\,.$$

Epipolar Geometry

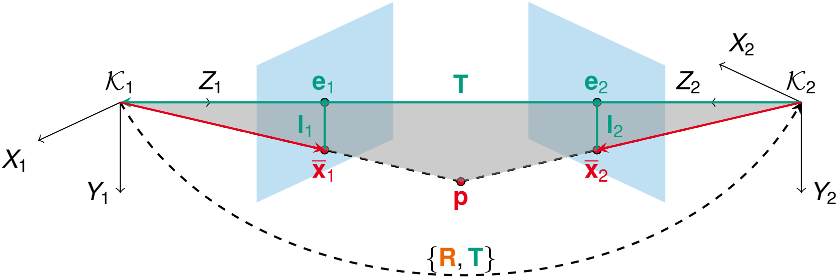
Discrete Epipolar Constraint



The discrete epipolar constraint $\bar{\mathbf{x}}_2^\top \widehat{\mathbf{T}}\mathbf{R}\,\bar{\mathbf{x}}_1 = 0$ corresponds to the triple product $< \bar{\mathbf{x}}_2, \mathbf{T} \times \mathbf{R}\bar{\mathbf{x}}_1 >$ of the vectors $\bar{\mathbf{x}}_2$, $\mathbf{T}$ and $\mathbf{R}\bar{\mathbf{x}}_1$ which all lie in the epipolar plane and thus the volume of the spanned parallelepiped is zero. If at least five or more point correspondences $(\mathbf{x}_1^i, \mathbf{x}_2^i)$ are given, the relative pose $(\mathbf{R}, \mathbf{T}/\|\mathbf{T}\|)$ can be determined down to the amount of translation $\|\mathbf{T}\|$, e.g. via the eight-point algorithm.

Epipolar Geometry

Discrete Epipolar Constraint



The intersections of the baseline $\overline{\mathcal{K}_1\mathcal{K}_2}$ with the image planes are called epipoles $\mathbf{e}_1, \mathbf{e}_2$ and the projections of all 3D points on an epipolar plane result in the so-called epipolar lines $\mathbf{l}_1, \mathbf{l}_2$, the intersection lines of the epipolar plane with the image planes.

Epipolar Geometry

Discrete Epipolar Constraint

Two views of the same 3D point from calibrated cameras must satisfy:

$$\overline{\mathbf{x}}_2^\top \mathbf{E}\, \overline{\mathbf{x}}_1 = 0\,, \quad \mathbf{E} = \widehat{\mathbf{T}}\mathbf{R}\,.$$

Other important properties:

- Epipolar lines: $\mathbf{l}_2 = \mathbf{E}\overline{\mathbf{x}}_1$ and $\mathbf{l}_1 = \mathbf{E}^\top \overline{\mathbf{x}}_2$
- Epipoles are the (left/right) null-space of the essential matrix:

$$\overline{\mathbf{e}}_2^\top \mathbf{E} = \mathbf{0}^\top\,, \quad \mathbf{E}^\top \overline{\mathbf{e}}_2 = \mathbf{0}\,, \quad \mathbf{E}\overline{\mathbf{e}}_1 = \mathbf{0}\,.$$

- Essential matrix is singular; has rank 2.
- The two remaining eigenvalues are equal.
- 5 degrees of freedom (translation + rotation have 6, but scale is arbitrary)

Eight-Point Algorithm

A simple algorithm to recover $(\mathbf{R}, \mathbf{T})$ from $N \geq 8$ corresponding pairs $(\mathbf{x}_1^j, \mathbf{x}_2^j)$, $j = 1, \ldots, N$. Three Steps:

1. Compute a first approximation of the essential matrix
2. Project onto the essential space
3. Recover the relative pose from the essential matrix

Epipolar Geometry

Eight-Point Algorithm

Technical University of Applied Sciences Würzburg-Schweinfurt

A simple algorithm to recover $(\mathbf{R}, \mathbf{T})$ from $N \geq 8$ corresponding pairs $(\mathbf{x}_1^j, \mathbf{x}_2^j)$, $j = 1, \ldots, N$.

1. Compute a first approximation of the essential matrix

N equations :
$$\begin{bmatrix} x_2^j & y_2^j & 1 \end{bmatrix} \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} \begin{bmatrix} x_1^j \\ y_1^j \\ 1 \end{bmatrix} = 0 \,.$$

Rearrange N equations (DLT):
$$\begin{bmatrix} x_1^j x_2^j & y_1^j x_2^j & x_2^j & x_1^j y_2^j & y_1^j y_2^j & y_2^j & x_1^j & y_1^j & 1 \end{bmatrix} \begin{bmatrix} E_{11} \\ E_{12} \\ E_{13} \\ E_{21} \\ E_{22} \\ E_{23} \\ E_{31} \\ E_{32} \\ E_{33} \end{bmatrix} = 0 \,.$$

Stack all N equations to form a system of linear equations:
$$\mathbf{A}\mathbf{e} = \mathbf{0} \,.$$

Solve it, s.t. $\|\mathbf{e}\| = 1$ using SVD.

Technical University of Applied Sciences Würzburg-Schweinfurt

Eight-Point Algorithm

A simple algorithm to recover $(\mathbf{R}, \mathbf{T})$ from $N \geq 8$ corresponding pairs $(\mathbf{x}_1^j, \mathbf{x}_2^j)$, $j = 1, \ldots, N$.

2. Project onto the essential space

Rearrange $\mathbf{e} \rightarrow \tilde{\mathbf{E}}$ and decompose it using SVD:

$$\tilde{\mathbf{E}} = \mathbf{U}\tilde{\mathbf{S}}\mathbf{V}^\top \quad \text{rank 3}$$

Replace $\tilde{\mathbf{S}} \rightarrow \mathbf{S}$ to fullfill constraints of an Essential-Matrix:

$$\mathbf{S} = diag\{1, 1, 0\}$$

Project $\tilde{\mathbf{E}} \rightarrow \mathbf{E}$ to the essential space:

$$\mathbf{E} = \mathbf{U}\mathbf{S}\mathbf{V}^\top \quad \text{rank 2}$$

Epipolar Geometry

Technical University of Applied Sciences Würzburg-Schweinfurt

Eight-Point Algorithm

A simple algorithm to recover $(\mathbf{R}, \mathbf{T})$ from $N \geq 8$ corresponding pairs $(\mathbf{x}_1^j, \mathbf{x}_2^j)$, $j = 1, \dots, N$.

3. Recover the relative pose from the essential matrix

$$\mathbf{R} = \mathbf{U}\mathbf{R}_Z^\top\left(\pm\frac{\pi}{2}\right)\mathbf{V}^\top\,, \quad \widehat{\mathbf{T}} = \mathbf{U}\mathbf{R}_Z\left(\pm\frac{\pi}{2}\right)\mathbf{S}\mathbf{V}^\top\,,$$

with rotation by $\pm\pi/2$ about Z-axis and skew-symmetric translation $\widehat{\mathbf{T}}$:

$$\mathbf{R}_Z\left(\pm\frac{\pi}{2}\right) = \begin{bmatrix} 0 & \pm 1 & 0 \\ \mp 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \widehat{\mathbf{T}} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}.$$

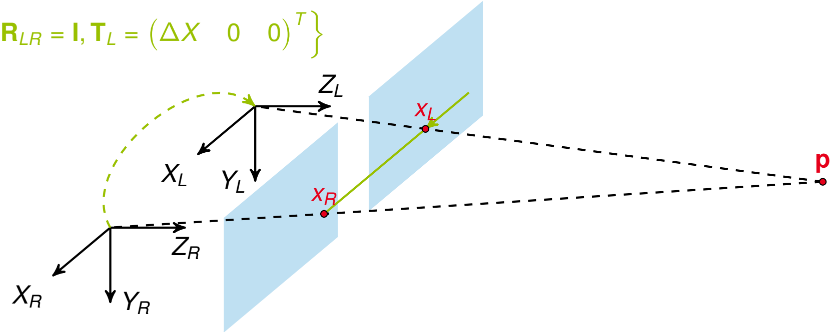
Hence, we get four possible decompositions $(\mathbf{R}, \widehat{\mathbf{T}})$ for $\mathbf{E}$ and the translation $\mathbf{T}$ is recovered up to a scalar factor (i.e. it is normalized to unit norm).

Epipolar Geometry

Technical University of Applied Sciences Würzburg-Schweinfurt

Stereo Vision

$$\left\{ \mathbf{R}_{LR} = \mathbf{I}, \mathbf{T}_L = \begin{pmatrix} \Delta X & 0 & 0 \end{pmatrix}^T \right\}$$



If the image planes of two cameras are aligned parallel to each other in the same plane, which corresponds to a relative pose of $(\mathbf{R}_{LR}, \mathbf{T}_L)$ (see above), then the distance of each point $\mathbf{p}$ can be reconstructed via the point correspondence of the projected coordinates (x_L, x_R) and the known relative pose. (Note: rectification!)

Triangulation via Stereo Vision

For the simplified case of identical intrinsic parameters $\mathbf{K}_L = \mathbf{K}_R = \mathbf{K}$, under the assumption of $s_\theta = 0$, the following relation is obtained

$$\mathbf{R}_{LR} \underbrace{Z_R \mathbf{K}^{-1} \overline{\mathbf{x}}'_R}_{\mathbf{X}_R} + \mathbf{T}_L = \underbrace{Z_L \mathbf{K}^{-1} \overline{\mathbf{x}}'_L}_{\mathbf{X}_L} \ .$$

This simplifies in the case of a rectified stereo system to

$$Z_R \begin{pmatrix} \frac{x'_R - o_x}{cs_x} \\ \frac{y'_R - o_y}{cs_y} \\ 1 \end{pmatrix} + \begin{pmatrix} \Delta X \\ 0 \\ 0 \end{pmatrix} = Z_L \begin{pmatrix} \frac{x'_L - o_x}{cs_x} \\ \frac{y'_L - o_y}{cs_y} \\ 1 \end{pmatrix} \ .$$

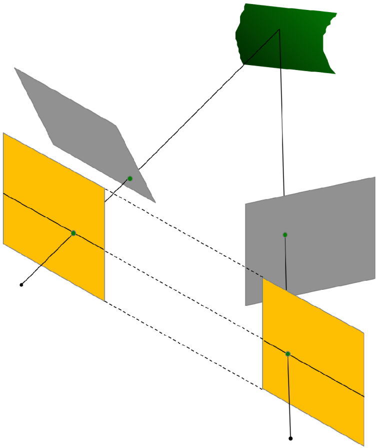
From the third line of the vector equation follows $Z_L = Z_R = Z$. Substituting into the second line follows $y'_R = y'_L$. Finally, the first line gives the depth reconstruction

$$Z = cs_x \Delta X / (x'_L - x'_R) \, .$$

Epipolar Geometry

Rectification

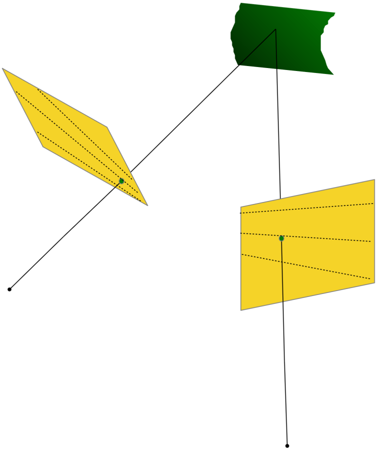
Reproject image planes onto a common plane parallel to the line between camera centers



Epipolar Geometry

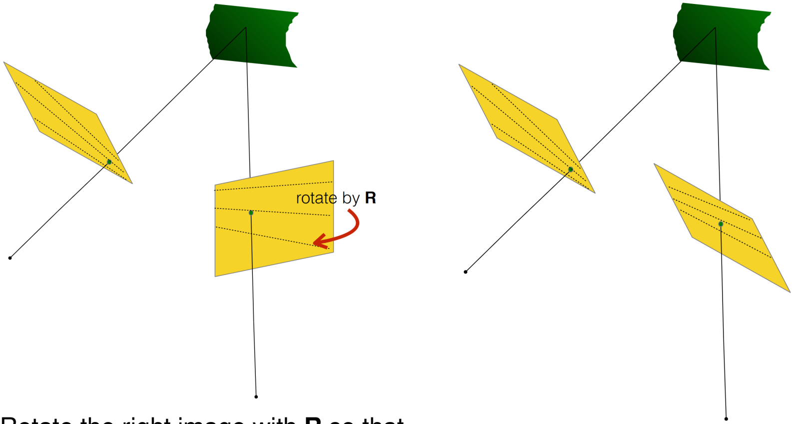
Rectification

Calculate the essential matrix **E** to obtain the rotation **R** between both image planes.



Epipolar Geometry

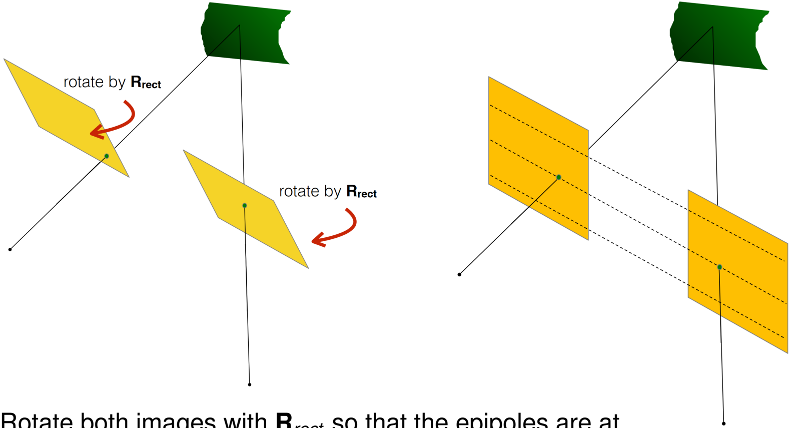
Rectification



Rotate the right image with **R** so that both images are oriented the same way

Epipolar Geometry

Rectification



Rotate both images with $\mathbf{R}_{rect}$ so that the epipoles are at infinity and the epipolar lines are parallel.

Epipolar Geometry

Rectification

Construction of $\mathbf{R}_{rect} = \begin{bmatrix} \mathbf{r}_1^\top \\ \mathbf{r}_2^\top \\ \mathbf{r}_3^\top \end{bmatrix}$:

1. epipole direction = translational direction: $\mathbf{e}_1 \overset{!}{=} \frac{\mathbf{T}}{\|\mathbf{T}\|} \quad \rightarrow \quad \mathbf{r}_1 = \frac{\mathbf{T}}{\|\mathbf{T}\|}$

2. y-axis $\perp$ optical axis & epipole:

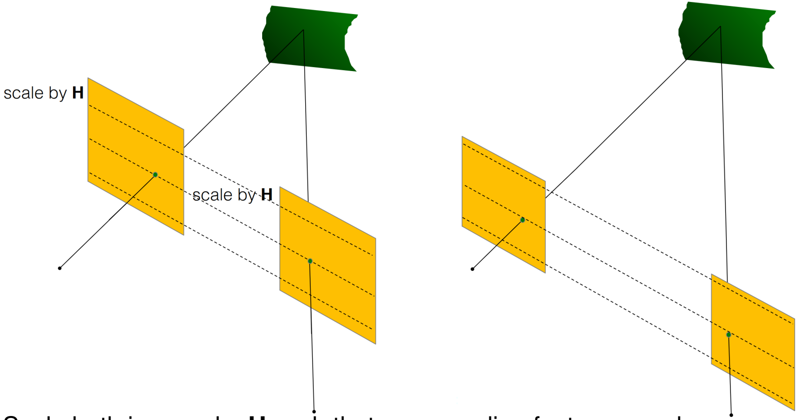
$$\|\mathbf{r}_2\| = \left\| \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \right\| \overset{!}{=} 1 \quad \rightarrow \quad \mathbf{r}_2 = \frac{1}{\sqrt{T_x^2 + T_y^2}} \begin{bmatrix} -T_y \\ T_x \\ 0 \end{bmatrix}$$

3. $\mathbf{r}_1 \perp \mathbf{r}_2 \perp \mathbf{r}_3 \wedge \|\mathbf{r}_1\| = \|\mathbf{r}_2\| = \|\mathbf{r}_3\| = 1 \quad \rightarrow \quad \mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$

Epipolar Geometry

thws Technical University of Applied Sciences Würzburg-Schweinfurt

Rectification



Scale both images by **H** such that corrsponding features can be found on the same image scanline.

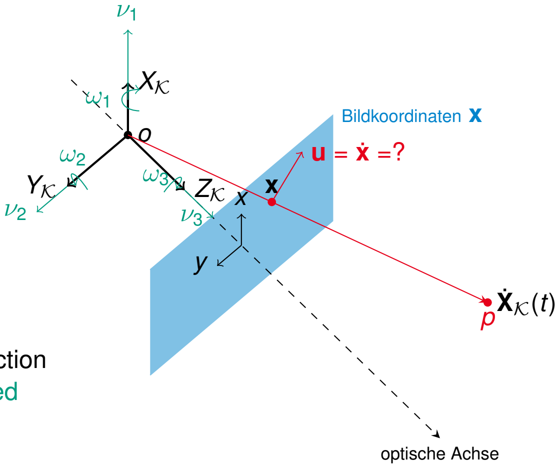
Epipolar Geometry

thws
Technical University
of Applied Sciences
Würzburg-Schweinfurt

Moving Camera

Considering the temporal change $\dot{\mathbf{x}}$ of the projection of a static 3D scene onto the image plane of a moving camera and describe the dependencies between the coordinates of the projection $\mathbf{x}$ and the twist (ω, ν), then one speaks of continuous epipolar geometry.

The resulting vector field of projection changes is also called self-induced optical flow $\mathbf{u}$.



Epipolar Geometry

Moving Camera

The following relationships are known so far:

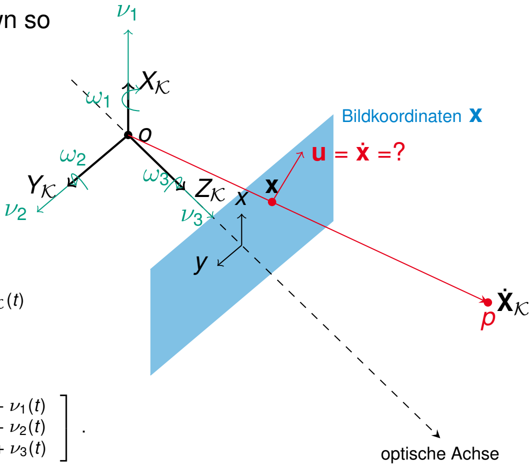
I. Normalised projection: $Z_{\mathcal{K}}\overline{\mathbf{x}} = \mathbf{X}_{\mathcal{K}}$

$$\longleftrightarrow \qquad \mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{Z_{\mathcal{K}}} \begin{bmatrix} X_{\mathcal{K}} \\ Y_{\mathcal{K}} \end{bmatrix}$$

II. Rogod body motion

$$\begin{aligned} \dot{\mathbf{X}}_{\mathcal{K}}(t) &= \dot{\mathbf{R}}(t)\mathbf{R}^{T}(t)\mathbf{X}_{\mathcal{K}}(t) + \dot{\mathbf{T}}_{\mathcal{K}}(t) - \mathbf{R}^{(}t)\mathbf{T}_{\mathcal{K}}(t) \\ &= \hat{\omega}(t)\mathbf{X}_{\mathcal{K}}(t) + \nu(t)\,, \end{aligned}$$

$$\begin{bmatrix} \dot{X}_{\mathcal{K}}(t) \\ \dot{Y}_{\mathcal{K}}(t) \\ \dot{Z}_{\mathcal{K}}(t) \end{bmatrix} = \begin{bmatrix} \omega_2(t)Z_{\mathcal{K}}(t) - \omega_3(t)Y_{\mathcal{K}}(t) + \nu_1(t) \\ \omega_3(t)X_{\mathcal{K}}(t) - \omega_1(t)Z_{\mathcal{K}}(t) + \nu_2(t) \\ \omega_1(t)Y_{\mathcal{K}}(t) - \omega_2(t)X_{\mathcal{K}}(t) + \nu_3(t) \end{bmatrix}.$$



Epipolar Geometry

Moving Camera

Technical University of Applied Sciences Würzburg-Schweinfurt

Derivative from the normalized projection with respect to time, equation I, gives:

III. $\dot{\mathbf{X}}_{\mathcal{K}} = \dot{Z}_{\mathcal{K}}\overline{\mathbf{x}} + Z_{\mathcal{K}}\dot{\overline{\mathbf{x}}}\,.$

Substituting equations I and III into the rigid body kinematics, equation II, yields:

IV. $$\dot{\overline{\mathbf{x}}} = \begin{bmatrix} u \\ v \\ 0 \end{bmatrix} = \widehat{\omega}\overline{\mathbf{x}} + \frac{1}{Z_{\mathcal{K}}}\nu - \frac{\dot{Z}_{\mathcal{K}}}{Z_{\mathcal{K}}}\overline{\mathbf{x}}\,.$$

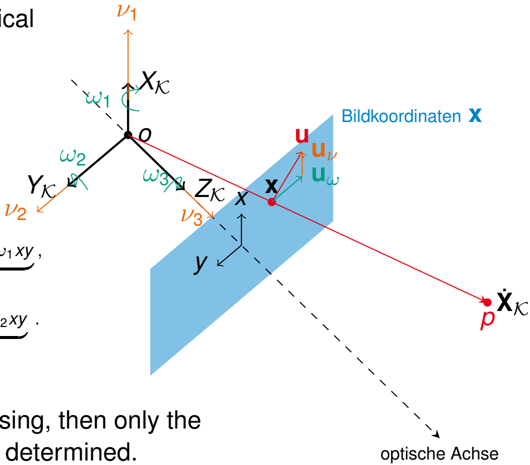
Solving the third line of equation IV after the depth derivative

V. $\dot{Z}_{\mathcal{K}} = \omega_1 y Z_{\mathcal{K}} - \omega_2 x Z_{\mathcal{K}} + \nu_3$

and substituting equation V into the first and second lines of IV,

Self-induced Optical Flow

leads to the intrinsically induced optical flow $\mathbf{u} = [u, v]^T$, which is additively composed of a rotational $\mathbf{u}_\omega$ and translational $\mathbf{u}_\nu$ part where only the translational part depends on the distance $Z_\mathcal{K}$:



$$
\begin{aligned}
u &= \underbrace{(\nu_1 - x\nu_3)/Z_\mathcal{K}}_{u_\nu} + \underbrace{\omega_2(1 + x^2) - \omega_3 y - \omega_1 xy}_{u_\omega}, \\
v &= \underbrace{(\nu_2 - y\nu_3)/Z_\mathcal{K}}_{v_\nu} + \underbrace{\omega_1(1 - y^2) + \omega_3 x + \omega_2 xy}_{v_\omega}.
\end{aligned}
$$

If the distance information $Z_\mathcal{K}$ is missing, then only the translation direction $\mathbf{u}_\nu / \|\mathbf{u}_\nu\|$ can be determined.

Continuous Epipolar Constraint

Eliminating the dependence of the optical flow on the depth $Z_{\mathcal{K}}$ by applying to both sides of the vector equation IV

$$\dot{\overline{\mathbf{x}}} = \widehat{\omega}\overline{\mathbf{x}} + \frac{1}{Z_{\mathcal{K}}}\nu - \frac{\dot{Z}_{\mathcal{K}}}{Z_{\mathcal{K}}}\overline{\mathbf{x}}$$

the scalar product with cross-product $\nu \times \mathbf{x} = \widehat{\nu}\,\mathbf{x}$

$$\dot{\overline{\mathbf{x}}}^\top \widehat{\nu}\,\overline{\mathbf{x}} = \overline{\mathbf{x}}^\top \widehat{\omega}^\top \widehat{\nu}\,\overline{\mathbf{x}}\,,$$

leads to the so-called continuous epipolar confinement considering $\widehat{\omega} = -\widehat{\omega}^\top$

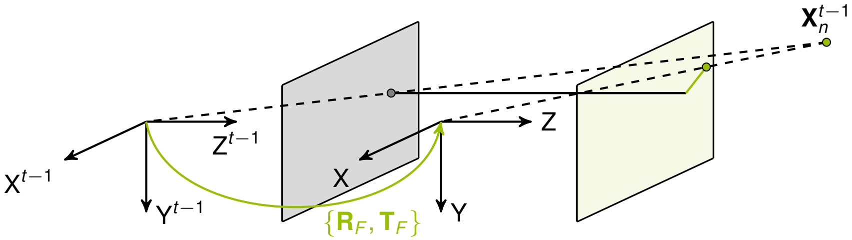
$$\overline{\mathbf{u}}^\top \widehat{\nu}\,\overline{\mathbf{x}} + \overline{\mathbf{x}}^\top \widehat{\omega}\widehat{\nu}\,\overline{\mathbf{x}} = 0\,.$$

Using this equation, the motion of the camera $(\omega, \nu/\|\nu\|)$ can be estimated if the optical flow $\mathbf{u}_i$ of at least five different coordinates $\mathbf{x}_i$ has been measured and no pure rotational motion $\nu \neq 0$ prevails.

Epipolar Geometry

Reprojection Error

For a moving stereo camera system, the coordinates $\mathbf{X}_n^t$ of each 3D point $\mathbf{p}_n$ can be calculated at any time t. If one calculates the projections $\mathbf{x}_n^{t-1}$ and $\mathbf{x}_n^t$, the so called optical flow, then one can calculate via the reprojection error the relative pose $(\mathbf{R}_F, \mathbf{T}_F)$ between the camera images of the stereo system. Considering one camera as a reference (either left or right camera of the stereo system), the following relation is obtained:



Reprojection Error

Approach of the reprojection error:

$$\underbrace{\left(\mathbf{x}_n^{t-1} - \mathbf{x}_n\right)}_{\text{Messung}} - \underbrace{\left(\mathbf{x}_n^{t-1} - \mathbf{K}\pi\left(\mathbf{R}_\mathrm{W}\mathbf{X}_n^{t-1} + \mathbf{T}_\mathrm{W}\right)\right)}_{\text{Modell}}$$

Only end-points:

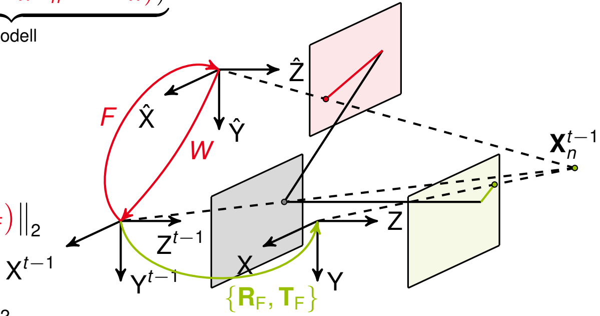
$$\mathbf{x}_n - \mathbf{K}\pi\left(\mathbf{R}_\mathrm{W}\mathbf{X}_n^{t-1} + \mathbf{T}_\mathrm{W}\right)$$

Backward mapping:

$$\epsilon_n = \left\|\mathbf{x}_n^{t-1} - \mathbf{K}\pi\left(\mathbf{R}_\mathrm{F}\mathbf{X}_n^{t} + \mathbf{T}_\mathrm{F}\right)\right\|_2$$

Motion estimation:

$$\{\hat{\mathbf{R}}_F, \hat{\mathbf{T}}_F\} = \mathrm{argmin}_{\mathbf{R}_F,\mathbf{T}_F} \sum_{n=1}^{N} \epsilon_n^2$$



Epipolar Geometrie

Reprojection Error

Show video about minimization of reprojection error!