VISUAL NAVIGATION

Visual odometry

Stereo cameras

Lecture outline:

- Monocular visual odometry
- Stereo camera geometry
- Stereo visual odometry

Visual odometry

- As seen in the previous lecture, features from two images are used to reconstruct the relative camera movements (rotations and translations) up to a scaling factory. Also, the 3D coordinates of the feature points can be reconstructed up to a scaling factor.

- Visual odometry is the process of estimating the relative movement of the camera using features points tracked across image sequences

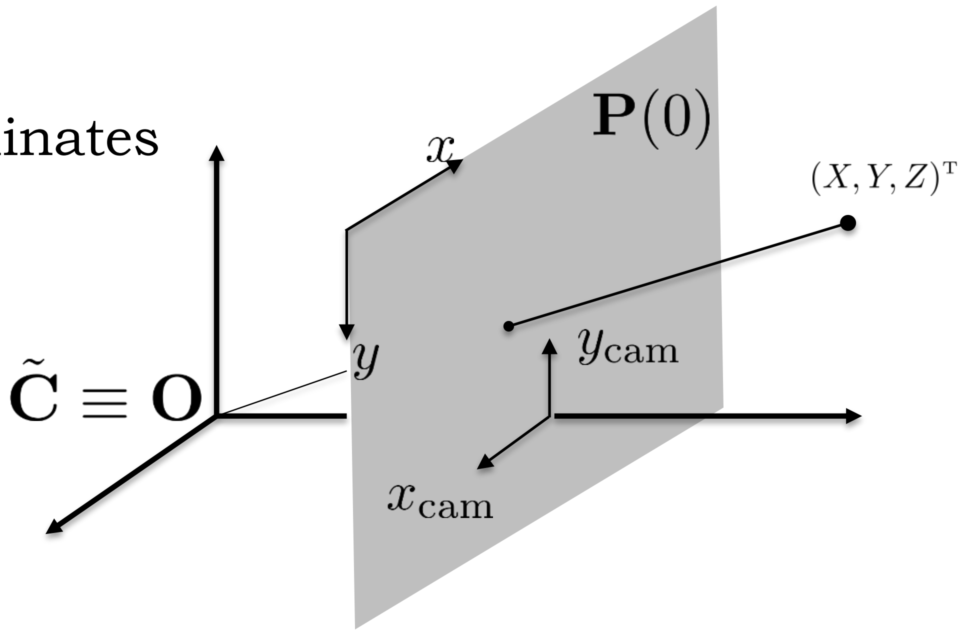
- We'll see in this lecture how to perform
  a) Visual odometry with a monocular camera
  b) Visual odometry with a stereo camera, in which the scale ambiguity is resolved

Monocular visual odometry

Workflow:

- I) Set the center of world-coordinates on the camera center at initial position:

$$\mathbf{P}(0) = \mathbf{K}[\mathbf{I} \mid \mathbf{0}]$$



** If calibrated cameras:*

$$\mathbf{P}(0) = [\mathbf{I} \mid \mathbf{0}]$$

- II) Remove radial and tangential distortions (if necessary)

Monocular visual odometry

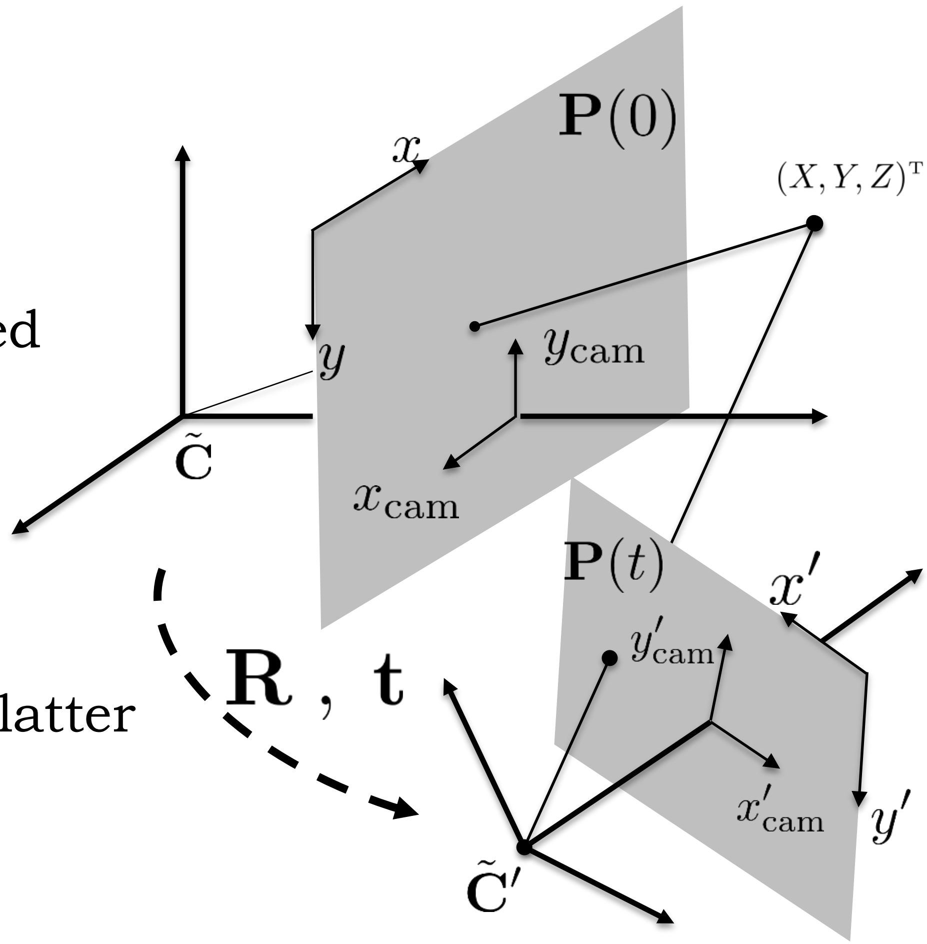
Workflow:

- III) Feature detection
- IV) Apply RANSAC to estimate essential matrix and matched features

$$\mathbf{x}'^{\mathrm{T}}\mathbf{E}^{(1)}\mathbf{x} = 0$$

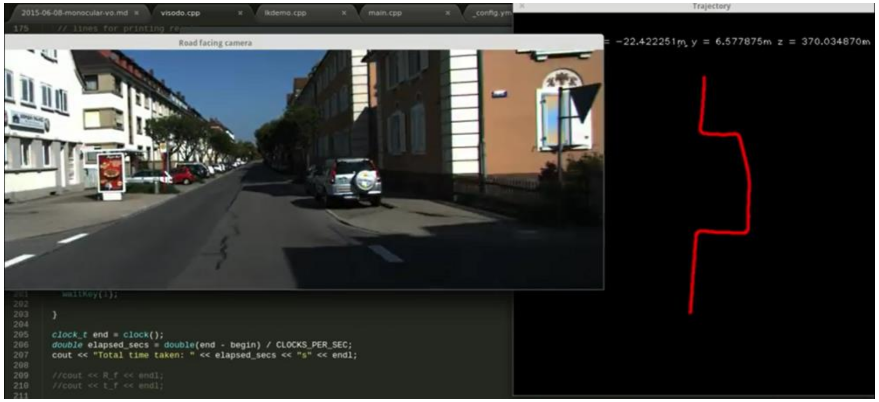
- V) Compute SVD of essential matrix to extract estimated rotation and translation, the latter only up to a scaling factor:

$$\hat{\mathbf{R}}\ ,\ \hat{\mathbf{t}}$$

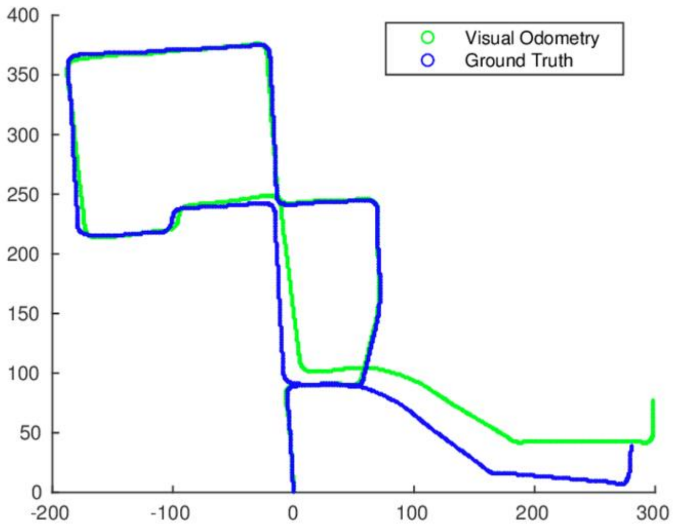


Monocular visual odometry

- Example of application:

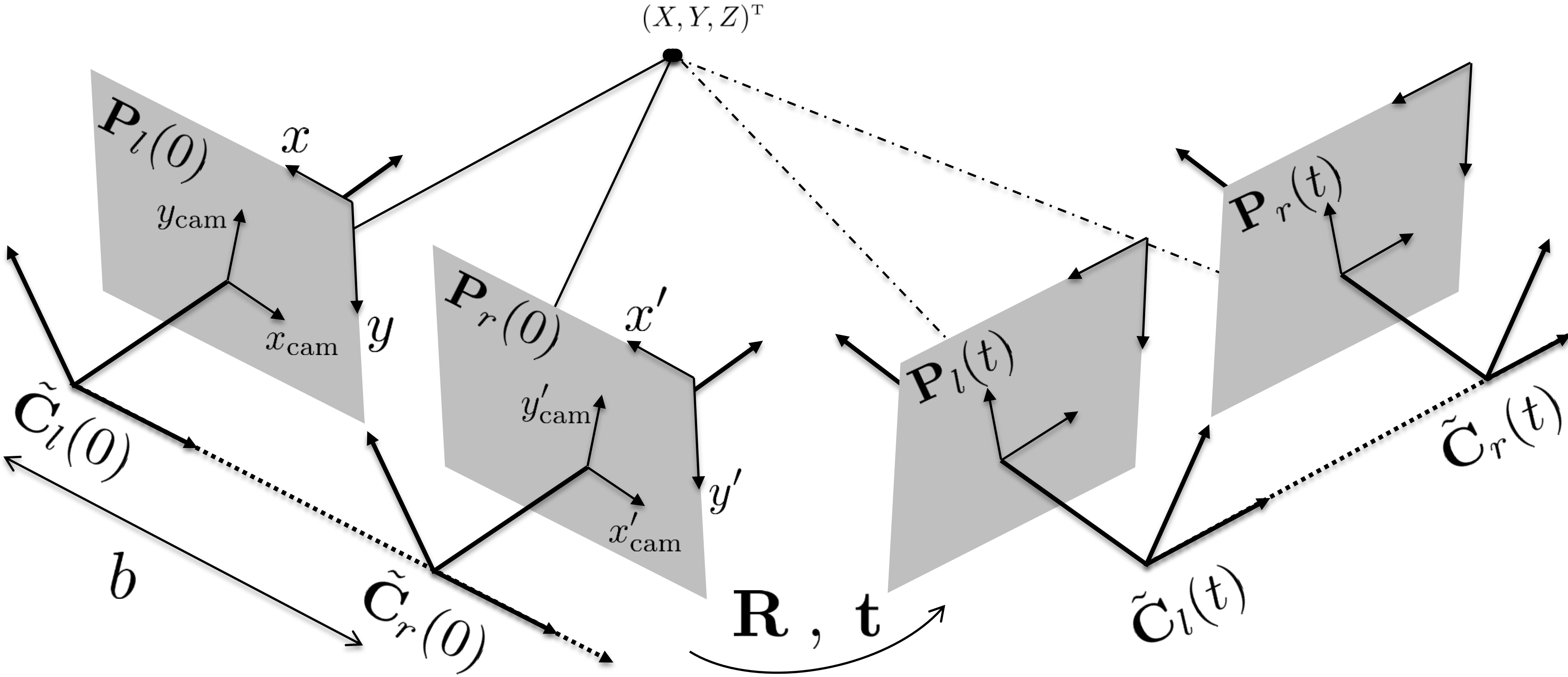


- Growing drift due to dead-reckoning approach: small errors tend to accumulate and grow to unacceptable levels



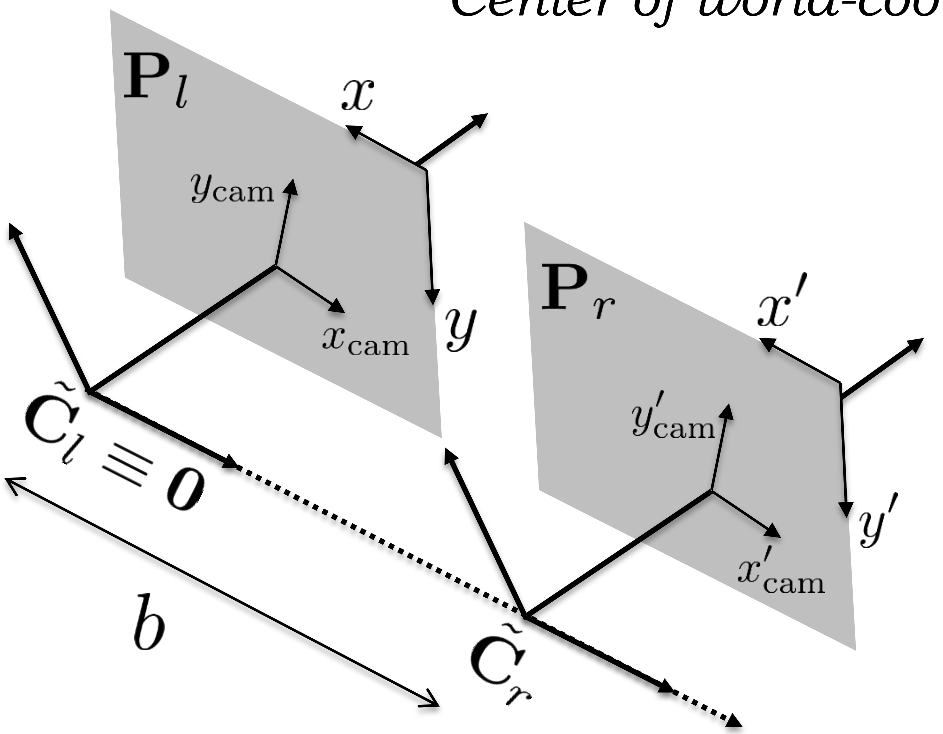
- How to restore scaling? Information from other sensors (INS, GNSS, wheel odometry) or from surrounding environment (control points whose coordinates are known)

Visual odometry with stereo cameras



Stereo camera geometry

- Couple of image sensors with (ideally) coplanar image planes:



Center of world-coordinates centered on the center of the left-camera:

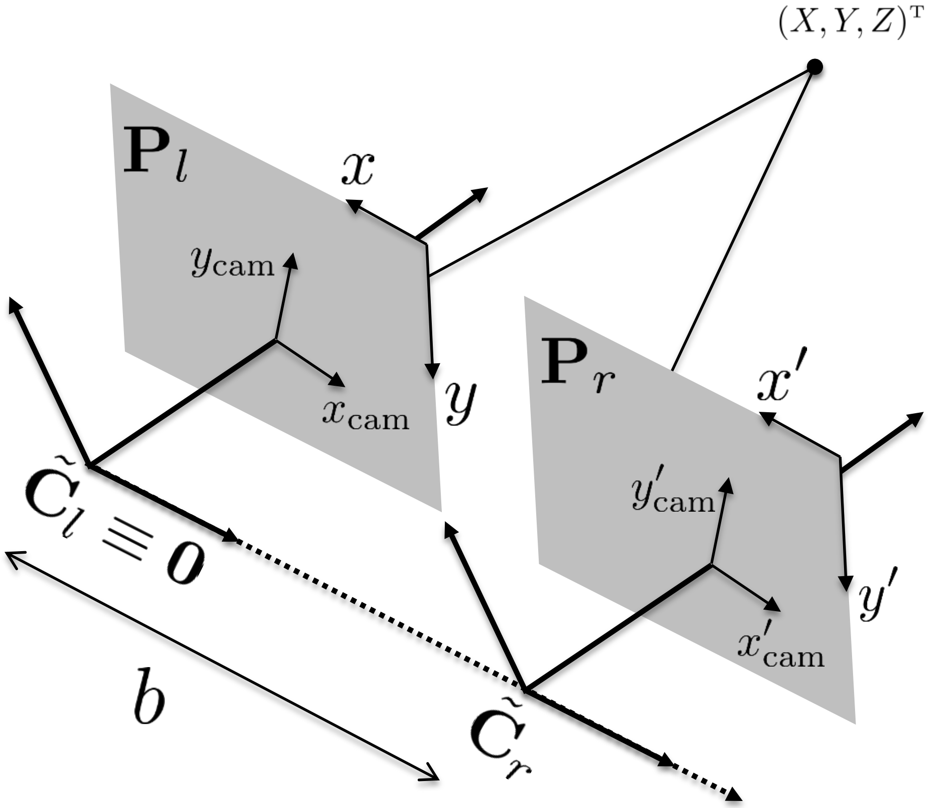
$$\mathbf{P}_l = \mathbf{K}_l[\mathbf{I} \mid \mathbf{0}]$$

$$\begin{aligned}\mathbf{P}_r &= \mathbf{K}_r[\mathbf{I} \mid \mathbf{t}_b] \\ &= \mathbf{K}_r[\mathbf{I} \mid (-b, 0, 0)^{\mathrm{T}}]\end{aligned}$$

Baseline distance (norm): b

Stereo camera geometry

- Geometry of image projection



Camera matrices:

$$\mathbf{P}_l = \mathbf{K}_l[\mathbf{I} \mid \mathbf{0}]$$
$$\mathbf{P}_r = \mathbf{K}_r[\mathbf{I} \mid \mathbf{t}_b]$$

Epipoles at infinity!

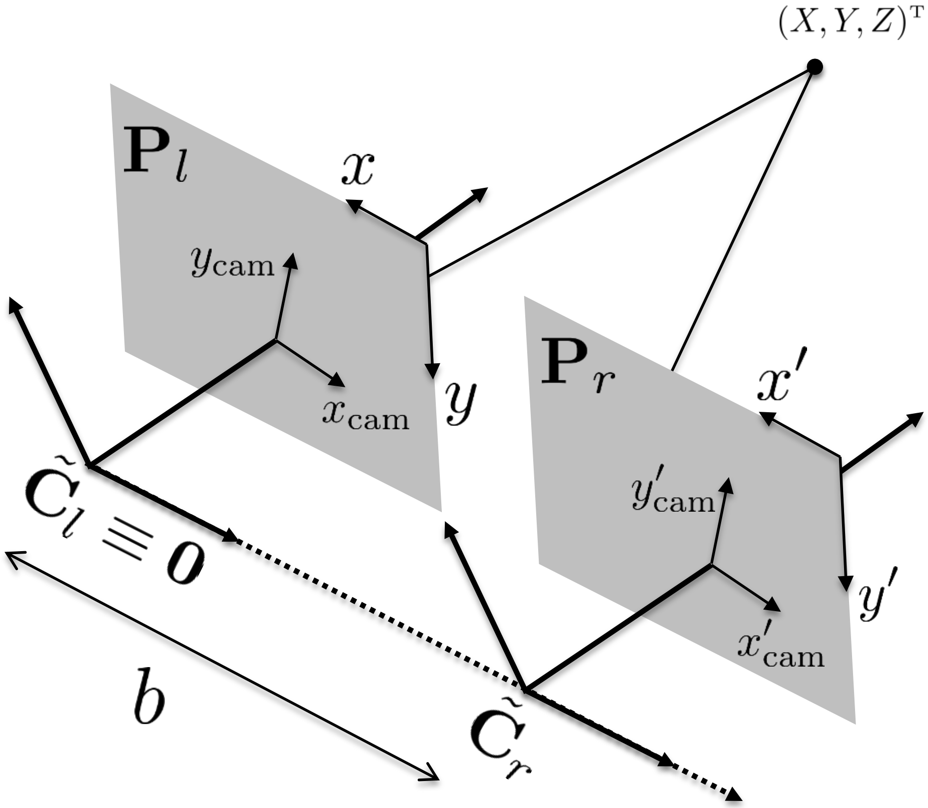
$$\mathbf{e}_r = \mathbf{P}_r\mathbf{C}_l = \mathbf{K}_r\mathbf{t}_b$$
$$\mathbf{e}_l = \mathbf{P}_l\mathbf{C}_r = -\mathbf{K}_l\mathbf{t}_b$$

Fundamental matrix:

$$\begin{aligned}\mathbf{F} &= \boldsymbol{\Omega}_{\mathbf{e}_r}\mathbf{P}_r\mathbf{P}_l^{+} \\ &= \boldsymbol{\Omega}_{\mathbf{e}_r}\mathbf{K}_r\mathbf{K}_l^{-1}\end{aligned}$$

Stereo camera geometry

- Geometry of image projection



Skew-matrix $\boldsymbol{\Omega}_{\mathbf{e}_r}$ *:*

$$\boldsymbol{\Omega}_{\mathbf{K}_r\mathbf{t}_b} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -e \\ 0 & e & 0 \end{bmatrix}$$

Often (but not always!):

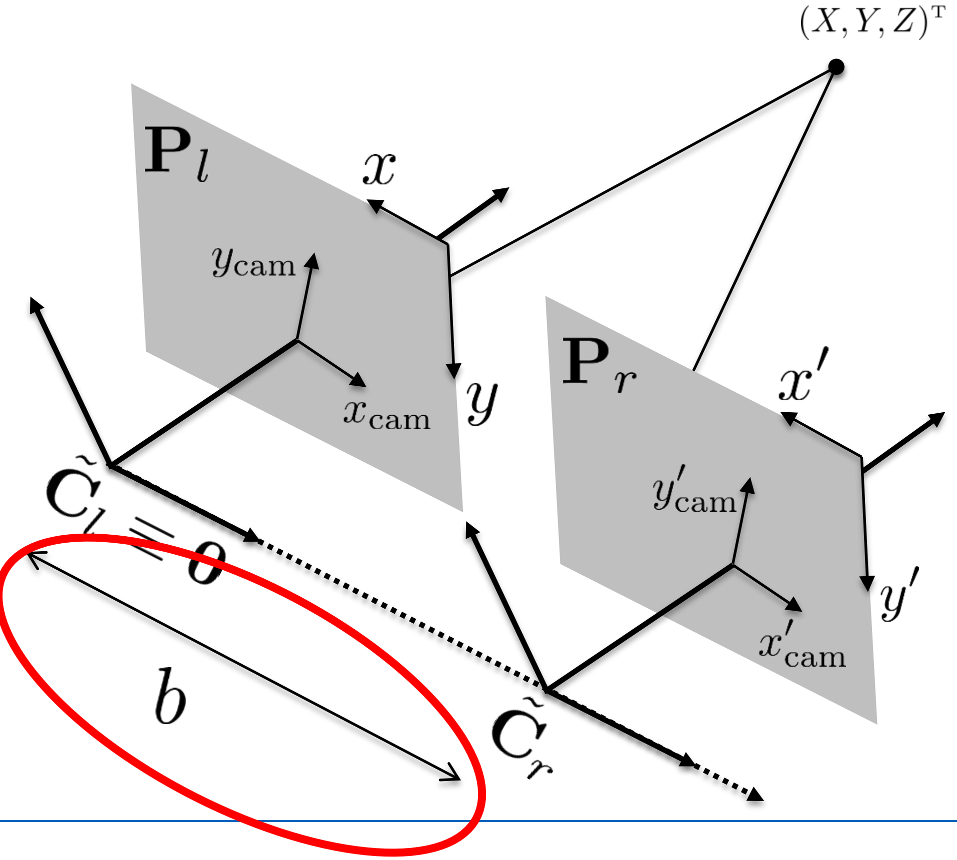
$$\mathbf{K}_r = \mathbf{K}_l \Rightarrow \mathbf{F} = \boldsymbol{\Omega}_{\mathbf{K}_r\mathbf{t}_b}$$

If calibrated, essential matrix is

$$\mathbf{E} = \boldsymbol{\Omega}_{\mathbf{t}_b}$$

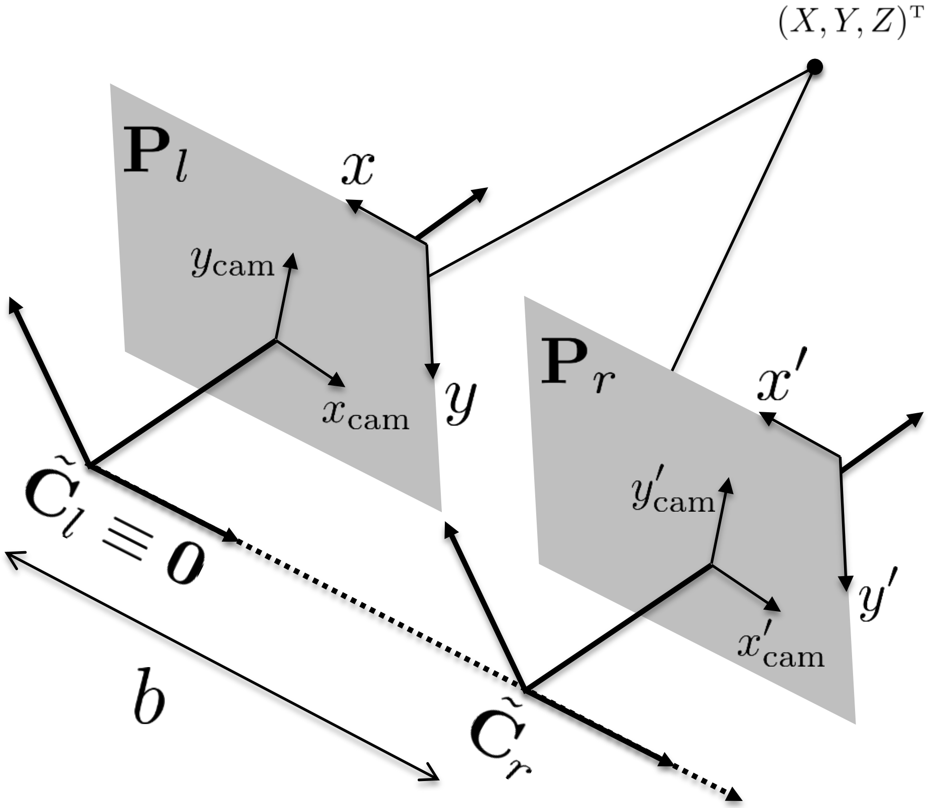
Stereo camera geometry

- Scaling is resolved! Knowledge of baseline enables reconstructing without scale ambiguity, contrary to the monocular case.



Stereo camera geometry

- Disparity map



Projection on image planes:

$$\mathbf{x}_l = \begin{pmatrix} -f\frac{X}{Z} + P_x \\ -f\frac{Y}{Z} + P_y \\ 1 \end{pmatrix}$$

$$\mathbf{x}_r = \begin{pmatrix} -f\frac{X-b}{Z} + P_x \\ -f\frac{Y}{Z} + P_y \\ 1 \end{pmatrix}$$

<u>Stereo disparity:</u>

$$d = (x_{r,1} - x_{l,1})^{\mathrm{T}} = f\frac{b}{Z}$$

Computation of the disparity enables reconstructing instantaneously the depth of feature points

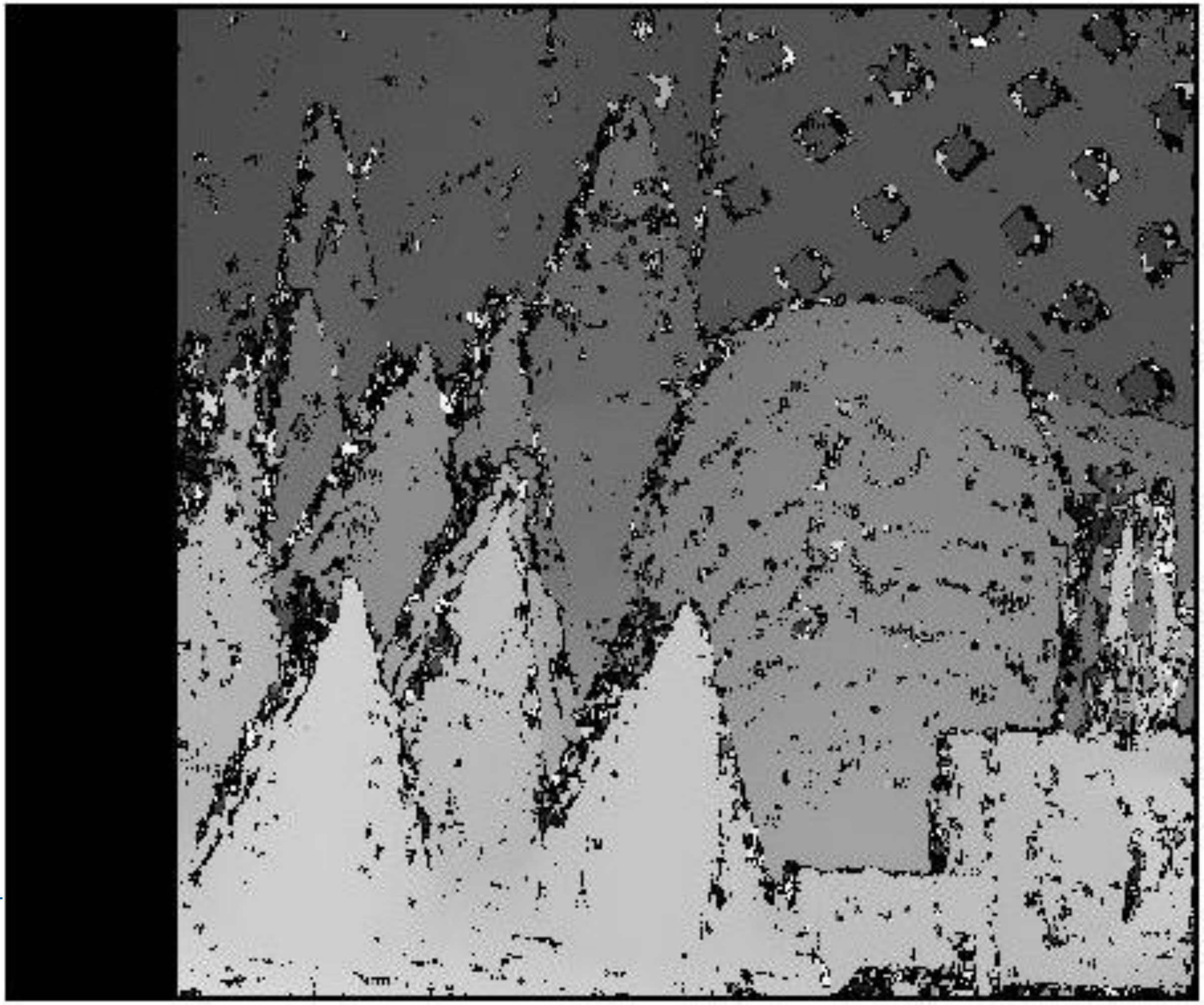
Stereo camera geometry

- Disparity map: example

Original images

Stereo camera geometry

- Disparity map: example



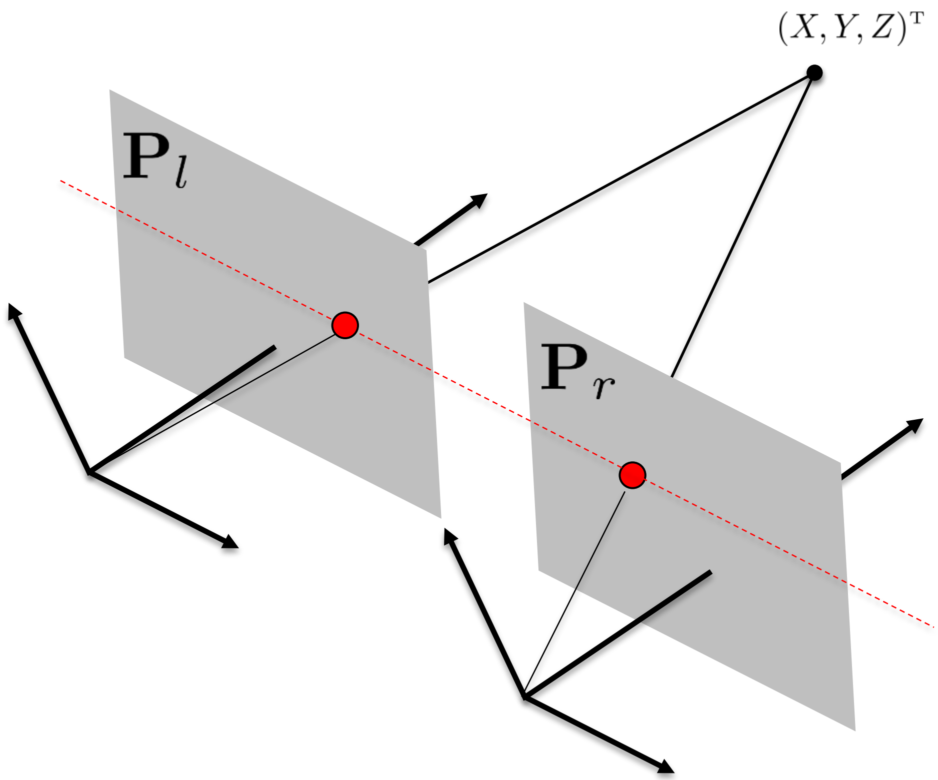
Color coding: brighter pixels are closer objects

Stereo camera geometry

- Example of disparity map-based obstacle avoidance

Stereo camera geometry

➢ The search for candidate matches is performed in <u>one dimension</u>
(on the corresponding to the epipolar line)



Stereo camera geometry

- The search for candidate matches is performed in <u>one dimension</u>

- Viable fast matching criteria: <u>block matching</u> through

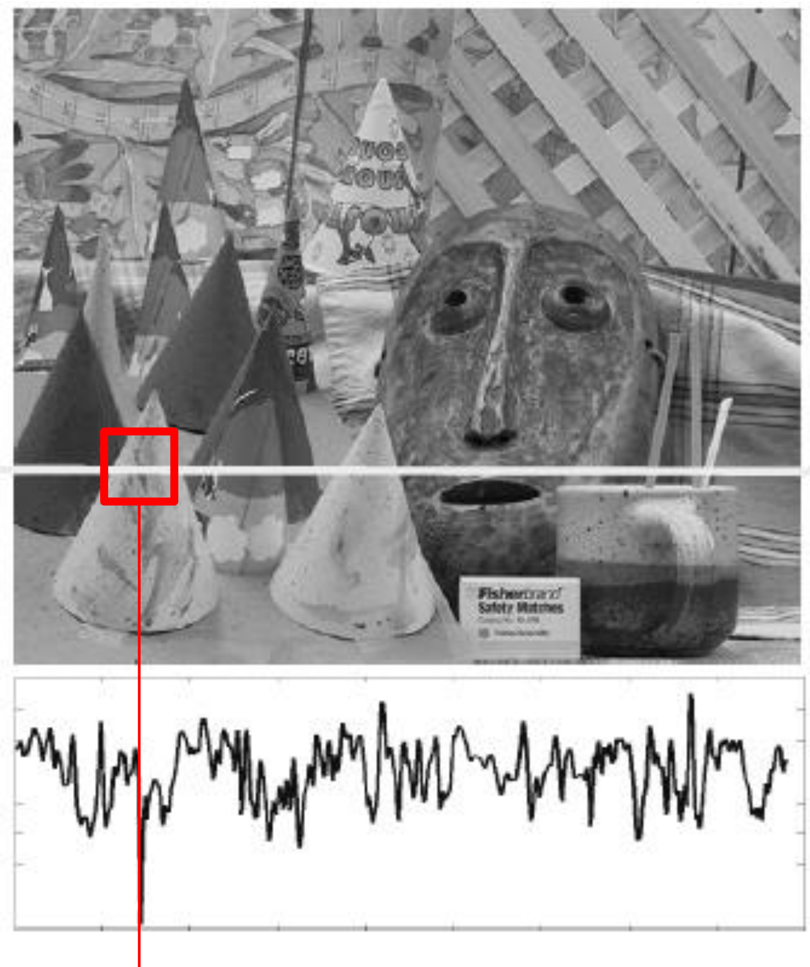
  - *minimization of SSD* $$f_{\mathrm{SSD}}(x) = \sum_{(u,v)\in\mathcal{W}} \left(I_l(u,v) - I_r(u-x,v)\right)^2$$

  - *maximization of cross-correlation* $$f_{\mathrm{CC}}(x) = \sum_{(u,v)\in\mathcal{W}} I_l(u,v) I_1(u-x,v)$$

Stereo camera geometry

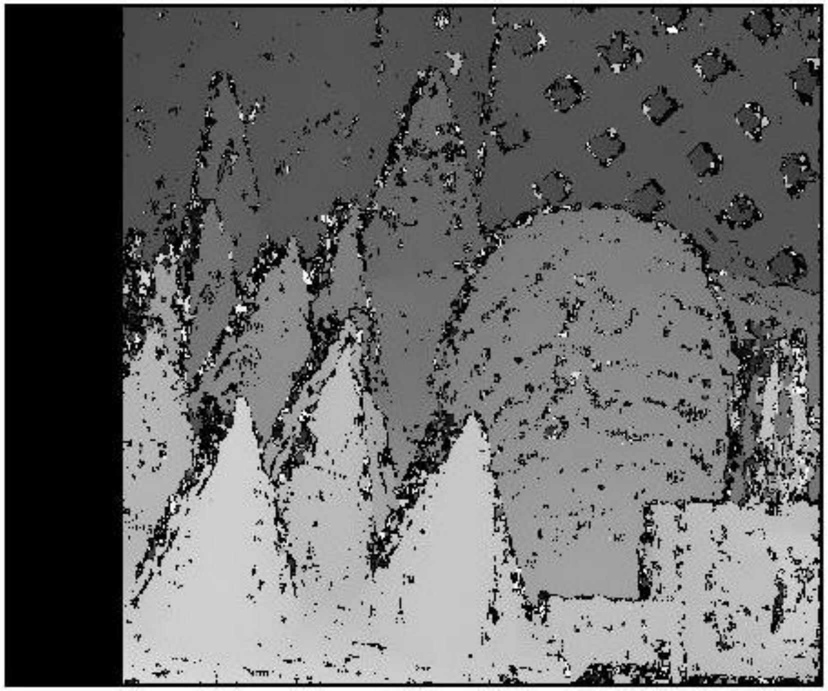
➢ The search for candidate matches is performed in <u>one dimension</u>

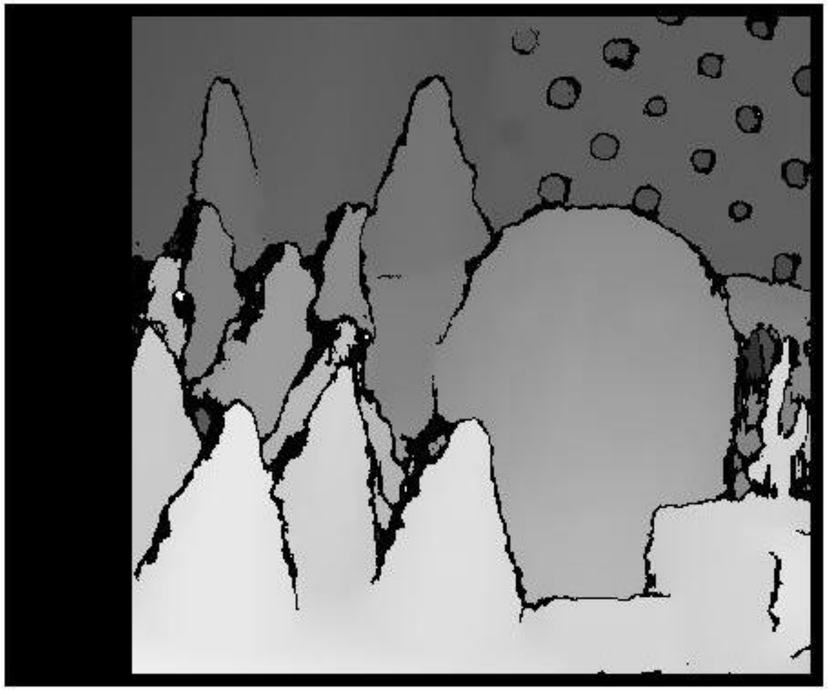
Minimization of SSD:



Stereo camera geometry

➢ Influence of patch size





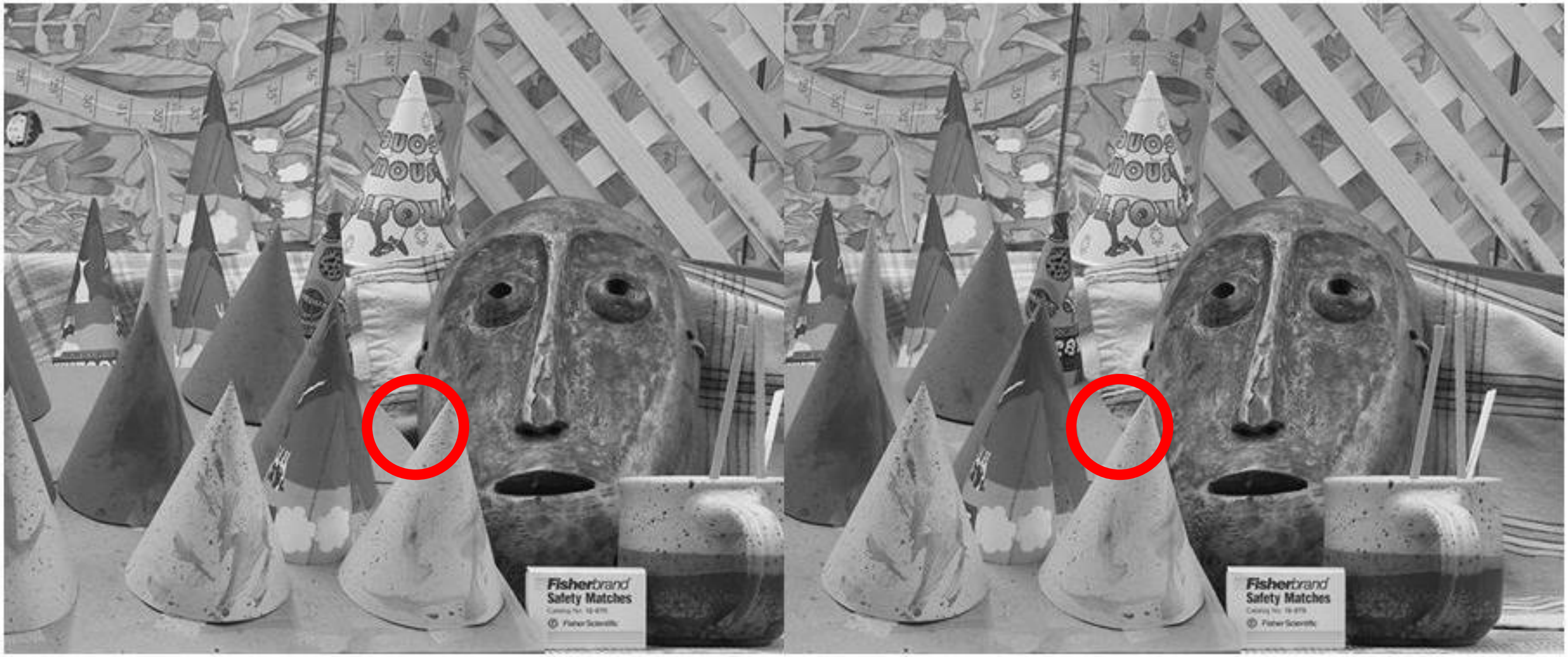
Small: higher definition, larger noise

Large: lower definition, smaller noise

Stereo camera geometry

- Example of occlusion (no match can be found)

Original images

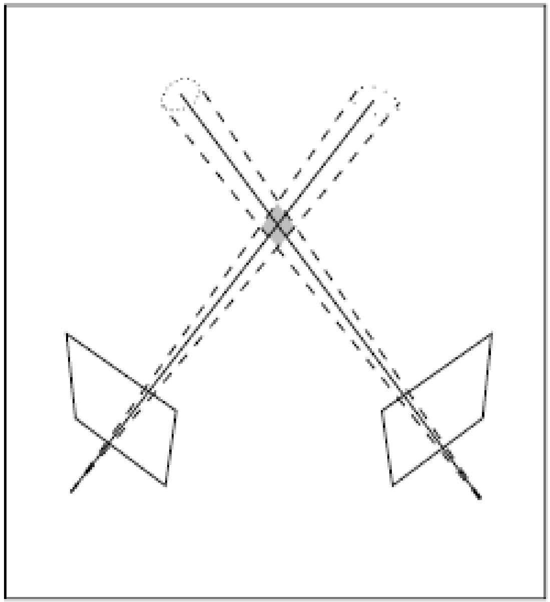


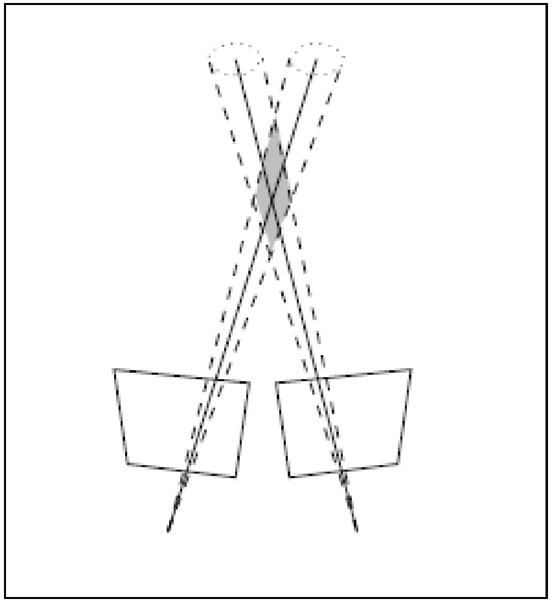
Stereo camera geometry

- Influence of baseline length

Short: larger depth errors, less problems with occlusions

Long: better depth definition, but larger chance of occlusions



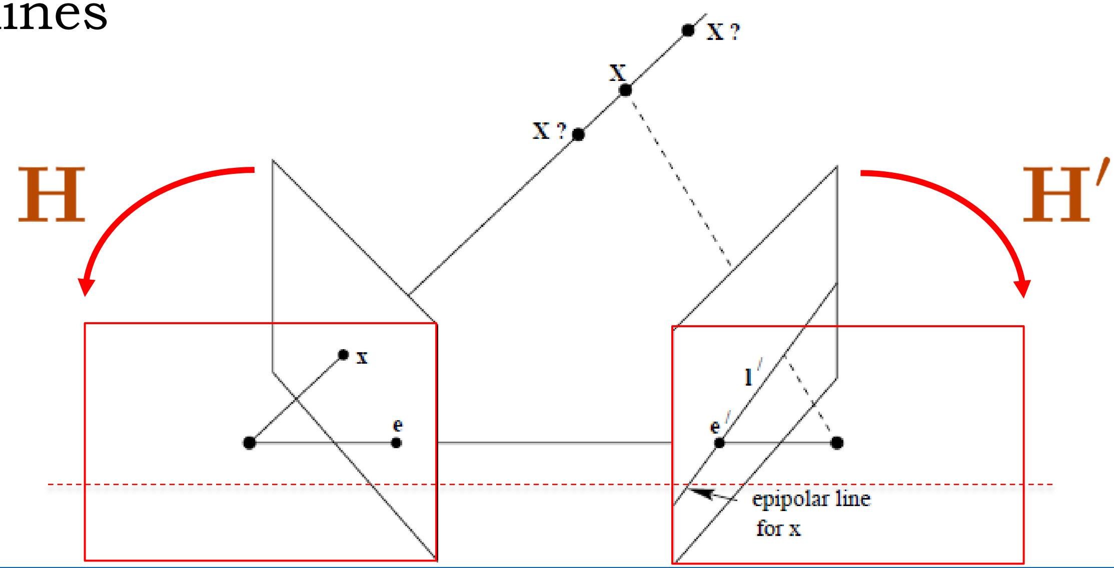


- Nota: stereo cameras are short-sighted! Depth of points can be reconstructed only as long as their images on the left and right focal planes show a minimum separation.

Stereo camera geometry

- If image planes not coplanar, search of matches between left and right images must be performed on non-parallel epipolar lines (higher computational load)

- <u>Rectification</u> of left-right images enables restoring one-dimensional search, by applying an homography that restore parallelism of epipolar lines



Stereo camera geometry

- The transformation is a planar homography

- Constraints (we assume $\mathbf{F}$ known):
  - *Correspondences on epipolar lines*

$$\mathbf{l}' = \mathbf{F}\mathbf{x} = \mathbf{\Omega}_{\mathbf{e}'}\mathbf{P}'\mathbf{P}^{+}\mathbf{x}$$

  - *Epipoles:*

$$\mathbf{F}\mathbf{e} = \mathbf{F}^{\mathrm{T}}\mathbf{e}' = \mathbf{0}$$

- Following rectification, we must obtain

$$\bar{\mathbf{F}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Stereo camera geometry

- The transformation is a planar homography
- Applying homographies $\mathbf{H}$ and $\mathbf{H}'$,points are transformed as

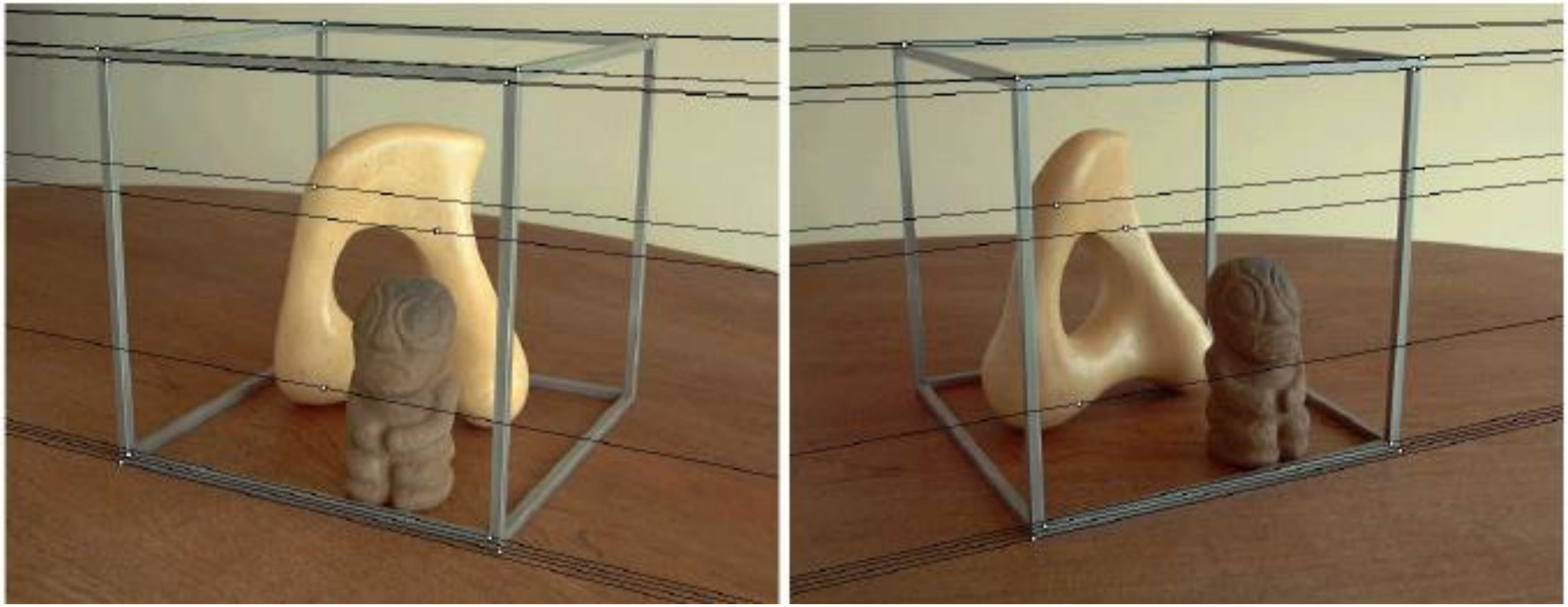
$$\bar{\mathbf{x}} = \mathbf{H}\mathbf{x} \qquad \bar{\mathbf{x}}' = \mathbf{H}'\mathbf{x}'$$

- Epipolar constraint: $(\bar{\mathbf{x}}')^{\mathrm{T}}\bar{\mathbf{F}}\bar{\mathbf{x}} = \mathbf{0} \quad \Rightarrow \quad (\mathbf{x}')^{\mathrm{T}}\underbrace{(\mathbf{H}')^{\mathrm{T}}\bar{\mathbf{F}}\mathbf{H}}_{\mathbf{F}}\mathbf{x} = \mathbf{0}$
- Relationship $\mathbf{F} = (\mathbf{H}')^{\mathrm{T}}\bar{\mathbf{F}}\mathbf{H}$ gives nine identities, but we have 16 independent elements to fix
- Viable approach: use remaining degrees of freedom to minimize distortions. Note that $\mathbf{H}$ and $\mathbf{H}'$ must be a general projections, since affine (and lower grade) transformations cannot reinstate points at infinity

Stereo camera geometry

- Example of a stereo rectification algorithm:
  C. Loop, Z. Zhang, "Computing Rectifying Homographies for Stereo Vision"

- *Original images:*



Stereo camera geometry

- Decompose as $\mathbf{H} = \mathbf{H}_s \mathbf{H}_a \mathbf{H}_p$ and $\mathbf{H}' = \mathbf{H}_s \mathbf{H}'_a \mathbf{H}'_p$
- Compute $\mathbf{H}_p$ and $\mathbf{H}'_p$ such that parallelism of epipolar lines is re-instated (<u>with minimal distortion</u>)

Stereo camera geometry

- Compute the rotations $\mathbf{H}_r$ and $\mathbf{H}_r'$ such that corresponding epipolar lines are aligned horizontally

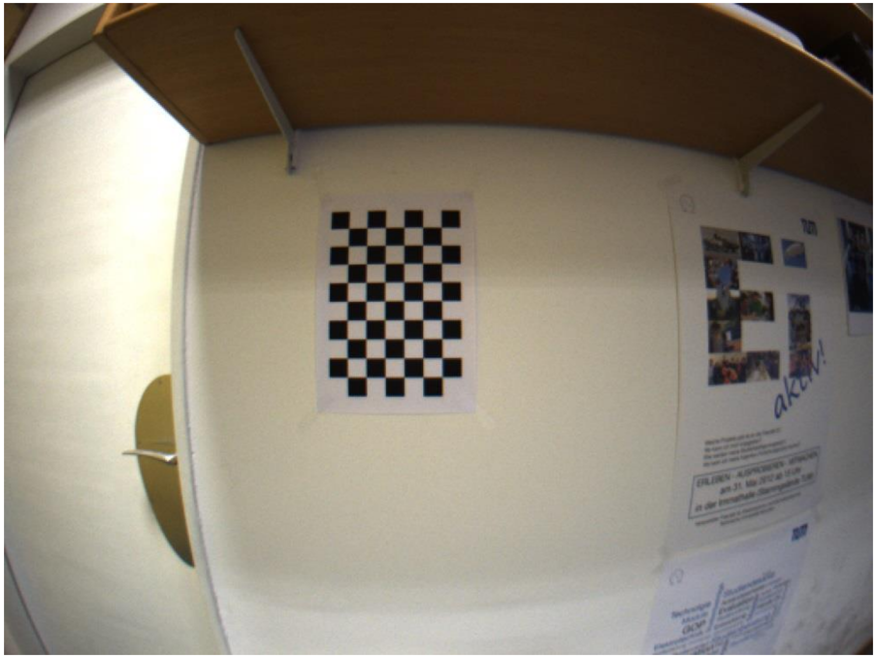
Stereo camera geometry

- Compute the affine transformation that reduces distortions (optional

Stereo camera geometry

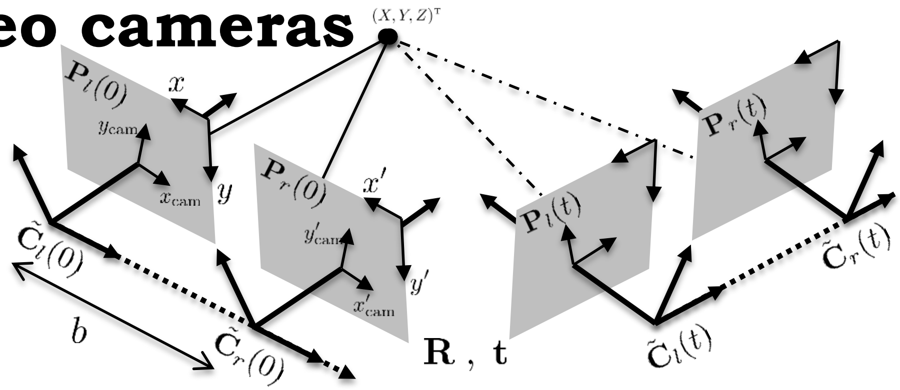
- Limitations on block-matching

Textureless surfaces



Repetitions and occlusions

Visual odometry with stereo cameras



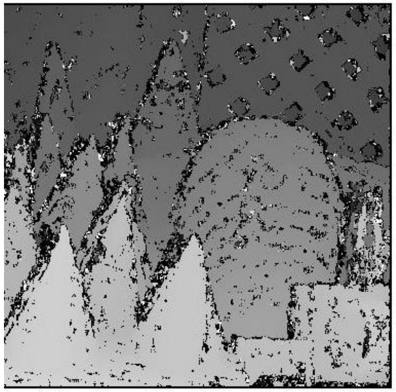
➢ Workflow:

I) Extract features from stereo pair

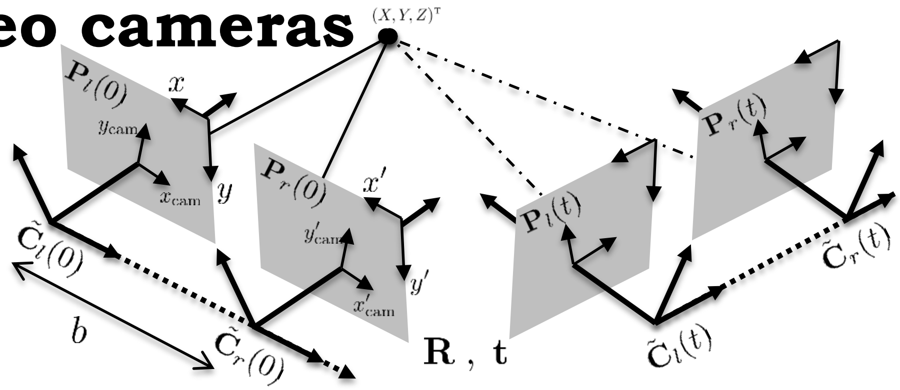
II) Fix (arbitrarily) initial camera matrices

$$\mathbf{P}_l(0) = \mathbf{K}_l[\mathbf{I} \mid \mathbf{0}]$$
$$\mathbf{P}_r(0) = \mathbf{K}_r[\mathbf{I} \mid \mathbf{t}_b]$$

and form disparity map.



Visual odometry with stereo cameras



- Workflow:

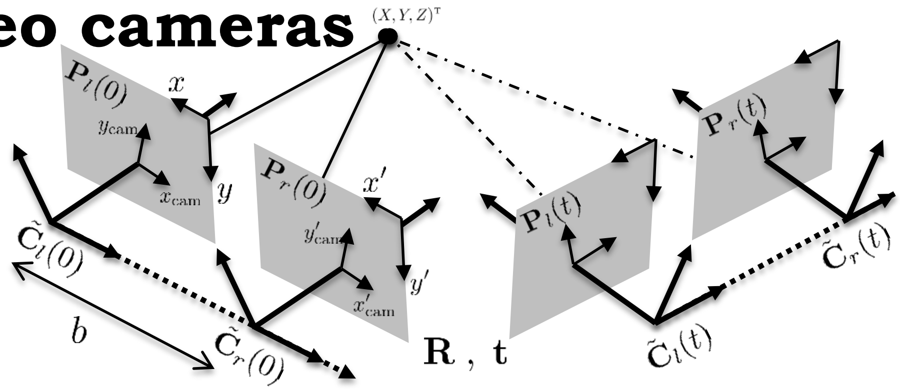
  III) Triangulate to obtain 3D points from imaged points

$$\mathbf{x}_{l,i}(0)\ ,\ \mathbf{x}_{r,i}(0)\ ,\ \mathbf{P}_l(0)\ ,\ \mathbf{P}_r(0) \quad \Rightarrow \quad \mathbf{X}_i$$

- *Fast triangulation is quickly obtained from disparity map as follows:*

$$\begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} = \begin{bmatrix} -1 & 0 & 0 & P_{x_l} \\ 0 & -1 & 0 & P_{y_l} \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{1}{b} & 0 \end{bmatrix} \begin{pmatrix} x_l \\ y_l \\ d \\ 1 \end{pmatrix} \Leftrightarrow \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} (P_x - x_l)\frac{b}{d} \\ (P_y - y_l)\frac{b}{d} \\ \frac{fb}{d} \end{pmatrix}$$

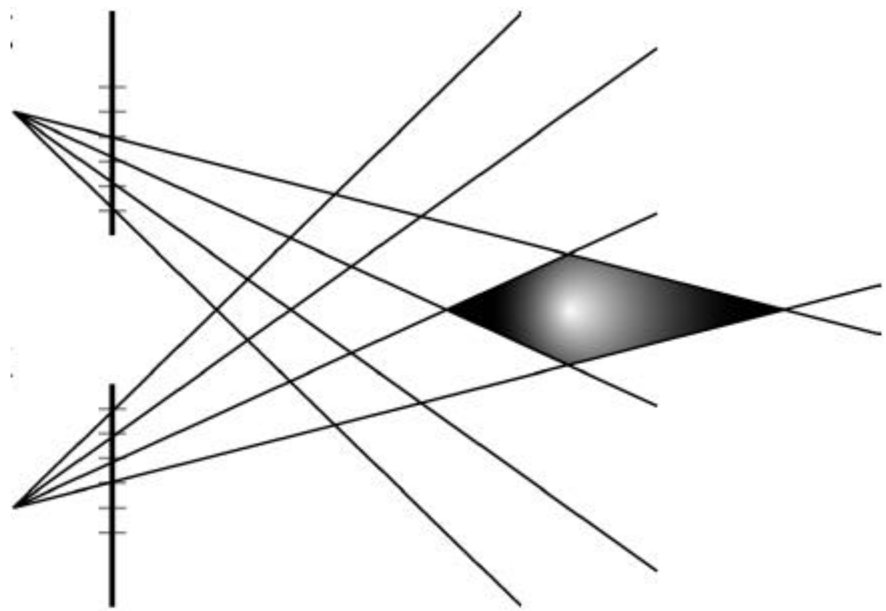
Visual odometry with stereo cameras



- Workflow:

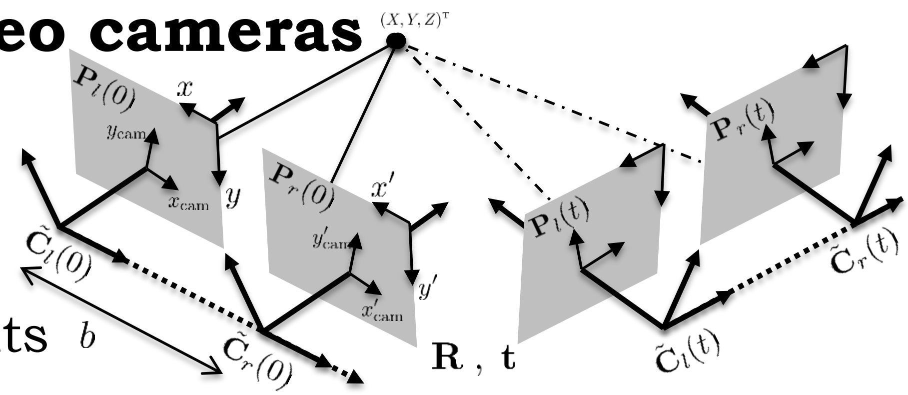
  III) Triangulate to obtain 3D points from imaged points

- *Pixel-precision gets 'diluted' when back-projecting pixel-regions for triangulation* →



- *Points further away are triangulated with very poor precision in depth direction*

Visual odometry with stereo cameras



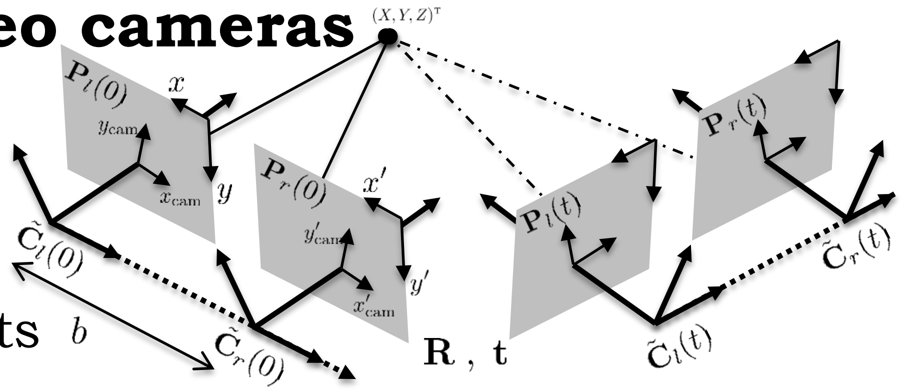
- Workflow:

  III) Triangulate to obtain 3D points from imaged points

- *Characterization of error*

  - *Depth is* $d = x_l - x_r$
  - *Variance-covariance of pixel (Gaussian) noise:* $\boldsymbol{\Sigma}_{\mathbf{x}} = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$
  - *Non-linear function for triangulation:* $\mathbf{X} = \mathbf{f}(\mathbf{p}) = \mathbf{f}(\mathbf{x}_l, x_r)$
  - *Jacobian of* $\mathbf{f}(\mathbf{p})$: $\mathbf{J}_{\mathbf{p}}$

Visual odometry with stereo cameras



- Workflow:

  III) Triangulate to obtain 3D points from imaged points

- *Characterization of error*

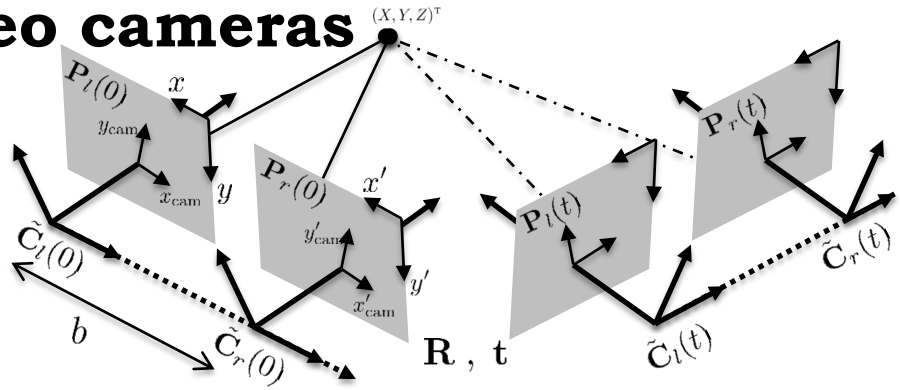
  - *Dispersion (error) on* $\mathbf{p}$: $\Sigma_{\mathbf{p}} = \begin{bmatrix} \Sigma_{\mathbf{x}_l} & \mathbf{0} \\ \mathbf{0}^T & \sigma^2_{x_r} \end{bmatrix}$

  - *Approximation of 3D estimate from triangulation:*

$$\Sigma_{\mathbf{X}} \approx \mathbf{J}_{\mathbf{p}} \Sigma_{\mathbf{p}} \mathbf{J}_{\mathbf{p}}^T$$

- *This evaluation (although approximated) becomes extremely useful at a later step*

Visual odometry with stereo cameras



- Workflow:

  IV) Acquire next stereo pair

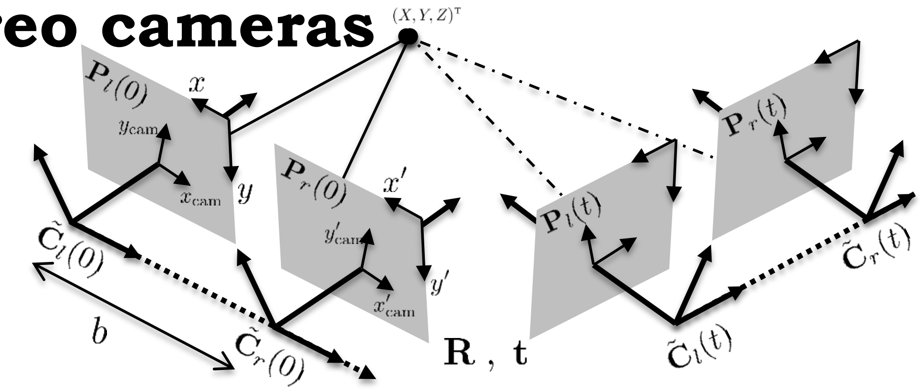
  V) Find features in common with previous pair: $\mathbf{x}_{l,i}(t)$, $\mathbf{x}_{r,i}(t)$

  a) by tracking points in corresponding left-left and right-right images (example: apply KLT algorithm)

  b) by extracting features and matching from previous set of features

- At this point, two different approaches are available (VIa and VIb): see next slides

Visual odometry with stereo cameras



- Workflow:

  VIa) Re-triangulate with <u>same previous camera matrices</u>

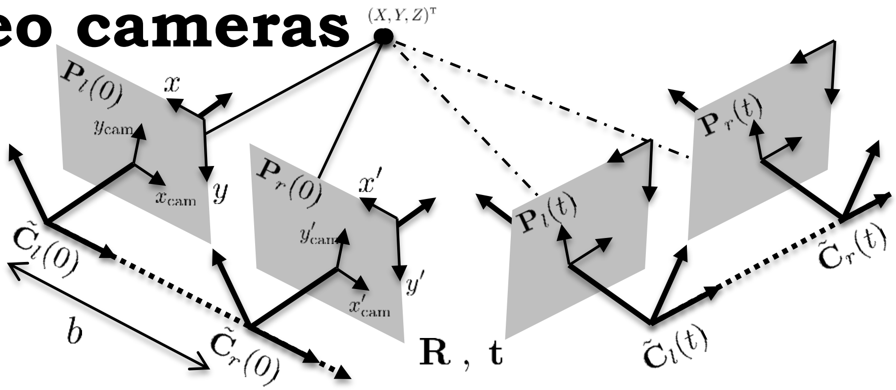
$$\mathbf{x}_{l,i}(t)\ ,\ \mathbf{x}_{r,i}(t) \quad \Rightarrow \quad \mathbf{X}_i(t)$$

We have now two sets of n points $\{\mathbf{X}_i(0)\}$, $\{\mathbf{X}_i(t)\}$ *related by a rigid body transformation (plus error):*

$$\mathbf{X}_i(t) = \mathbf{R}\mathbf{X}_i(0) + \mathbf{t} + \boldsymbol{\varepsilon}_i \quad , \quad i = 1, \ \ldots \ , n$$

- *MLE:* $\{\hat{\mathbf{R}}, \hat{\mathbf{t}}\} = \arg\max P(\mathbf{X}_1(t), \ldots, \mathbf{X}_n(t) \mid \mathbf{R}, \mathbf{t})$

Visual odometry with stereo cameras



- Workflow:

  VIa) Re-triangulate with <u>same previous camera matrices</u>

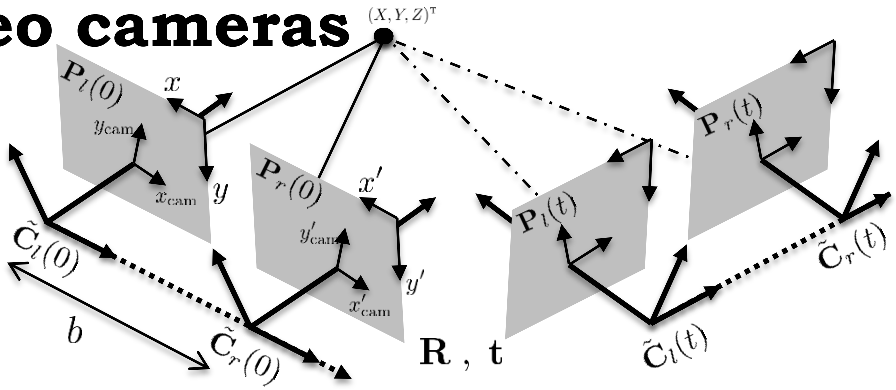
- *Under Gasussian hypothesis, we use the previously derived variance-covariance matrix* $\mathbf{\Sigma}_{\mathbf{X}_i(t)} \approx \mathbf{J}_{\mathbf{p}} \mathbf{\Sigma}_{\mathbf{p}} \mathbf{J}_{\mathbf{p}}^{\mathrm{T}}$ *for each "observation"* $\mathbf{X}_i(t)$

- *The probability* $P(\mathbf{X}_1(t), \ldots, \mathbf{X}_n(t) \mid \mathbf{R}, \mathbf{t})$ *is proportional to*

$$P(\mathbf{X}_1(t), \ldots, \mathbf{X}_n(t) \mid \mathbf{R}, \mathbf{t}) \propto e^{-\frac{1}{2} \sum_{i=1}^{n} \mathbf{r}_i^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{X}_i}^{-1} \mathbf{r}_i}$$

with residuals $\mathbf{r}_i = \mathbf{X}_i(t) - \mathbf{R}\mathbf{X}_i(0) - \mathbf{t}$

Visual odometry with stereo cameras



- Workflow:

  VIa) Re-triangulate with <u>same previous camera matrices</u>

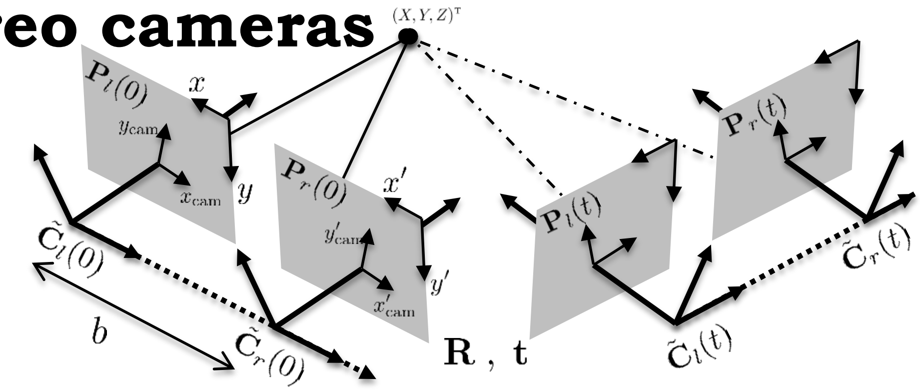
- *Maximizing the exponential equals to minimizing expression*

$$\{\hat{\mathbf{R}}, \hat{\mathbf{t}}\} = \arg\min \sum_{i=1}^{n} \mathbf{r}_i^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathbf{X}_i}^{-1} \mathbf{r}_i$$

- *Non-linear minimization of the squared weighted norm of residuals provides the sought MLE of rotation matrix and translation vector*

- *Apply RANSAC to remove outliers in the dataset (wrong triangulations leading to mismatches between* $\mathbf{X}_i(t)$ *and* $\mathbf{X}_i(0)$ *)*

Visual odometry with stereo cameras



- Workflow:

  VIb) Compute left (and/or right) camera pose only from 3D to 2D correspondences

$$\mathbf{x}_{l,i}(t) = \mathbf{P}_l \mathbf{X}_i \quad \Rightarrow \quad \mathbf{A}_i \mathbf{p}_l = \mathbf{0}$$

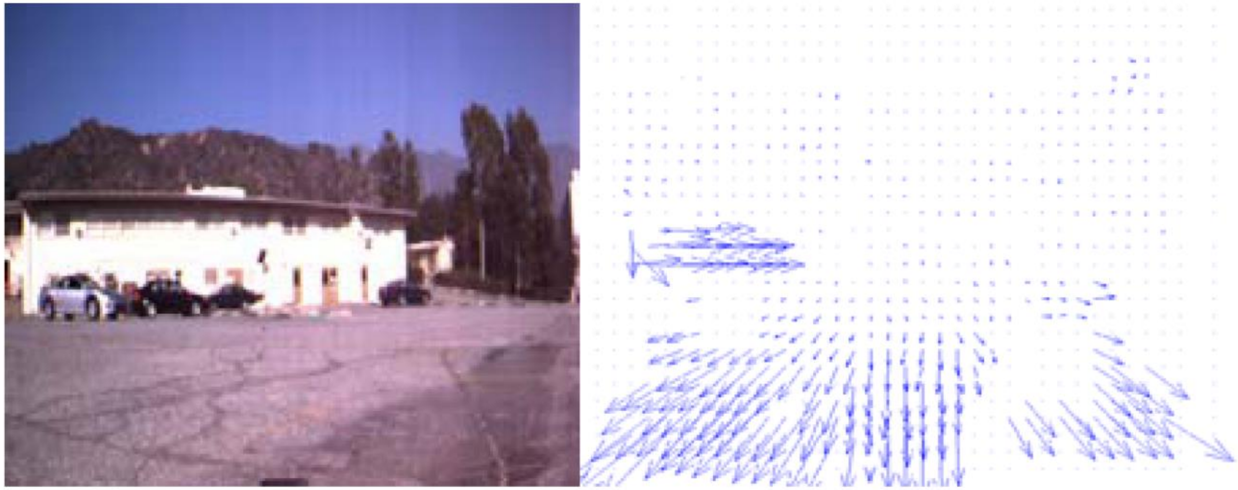
$$\hat{\mathbf{p}}_l = \arg \min_{\mathbf{p} \in \mathbb{R}^{12}\ ,\ \|\mathbf{p}_l\| = l} \|\mathbf{A}\mathbf{p}_l\|^2_{\mathbf{\Sigma}}$$

- Trade off between periodic triangulations (to acquire new features replacing points that move out from the camera field of view) and propagation from points $\mathbf{X}_i$ triangulated as back as possible (to limit drift)

Visual odometry with stereo cameras

- Nota: previous approaches operate under the assumption of rigid world, i.e., imaged objects/points are static. This is of course not always the case, and methods for detecting objects/points moving across images need to be implemented.

- Example (right): detection of the prevalent direction of motion and removal of those parts of the image that seem to be moving differently.



Example: Visual Odometry for Ground Vehicles

- Following results taken from
  D. Nister, O. Naroditsky, J. Bergen "Visual Odometry for Ground Vehicle Application"

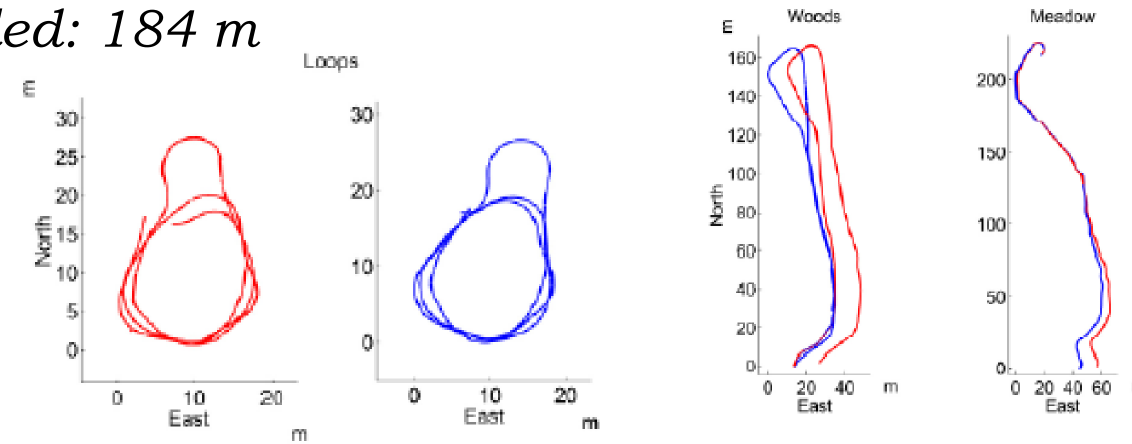
- *Uses* VIb) *approach*

- Stereo set-up:
  - 720 x 240 resolution
  - 13 Hz frame rate
  - 50 deg Horizontal field of view
  - Stereo baseline: 28 cm

Example: Visual Odometry for Ground Vehicles

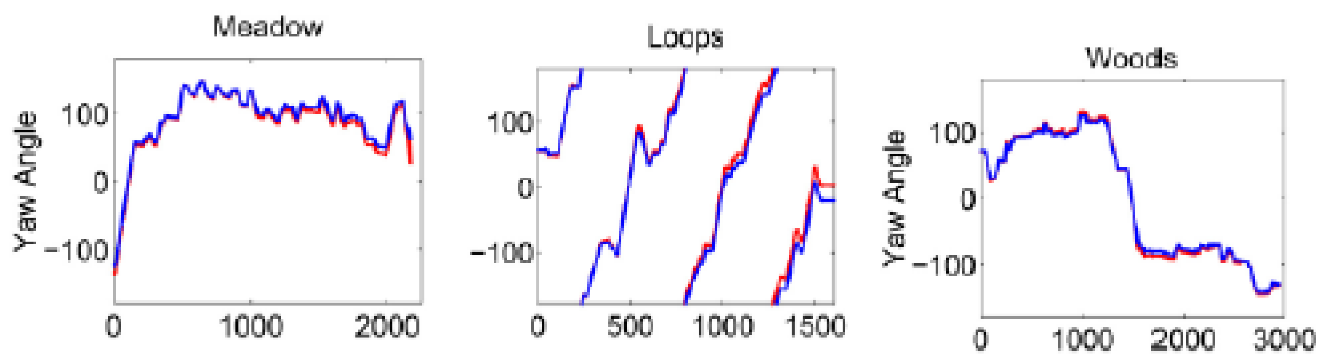
- Visual odometry versus Differential GPS positioning

  -Full 3D trajectory estimation (Red: VO ; Blue: GPS)
  - Linear distance travelled: 184 m
  - Final error: 4.1 m



- Visual odometry (attitude) versus INS

  - Sub-degree angular res. achieved



Monocular versus Stereo cameras

- Monocular cameras offer several advantages in terms of weight, cost, scalability, power consumption. These characteristics make monocular sensors ideal for a wide range of autonomous small platforms, e.g., quadrocopters. However, the scale ambiguity is an unacceptable draw-back for most applications. S

- Stereo cameras, although more costly in terms of weight and power consumption, enable timely trajectory estimation and map construction without scale ambiguity.

Further reading

- Stereo algorithms and rectification:
  M. Pollefeys, R. Koch, L. Van Gool, "A simple and efficient rectification method for general motion"
  C. Loop, Z. Zhang, "Computing Rectifying Homographies for Stereo Vision"
  D. Oram, "Rectification for Any Epipolar Geometry"
  H. Hirschmüller, "Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information"
- Visual odometry
  D. Nistér, O. Naroditsky, J. Bergen, "Visual Odometry for Ground Vehicle Applications"
  N. Sünderhauf, P. Protzel, "Stereo Odometry – A Review of Approaches"
  C.F. Olson, L.H. Matthies, M. Schoppers, M.W. Maimoneb, "Rover navigation using stereo ego-motion"