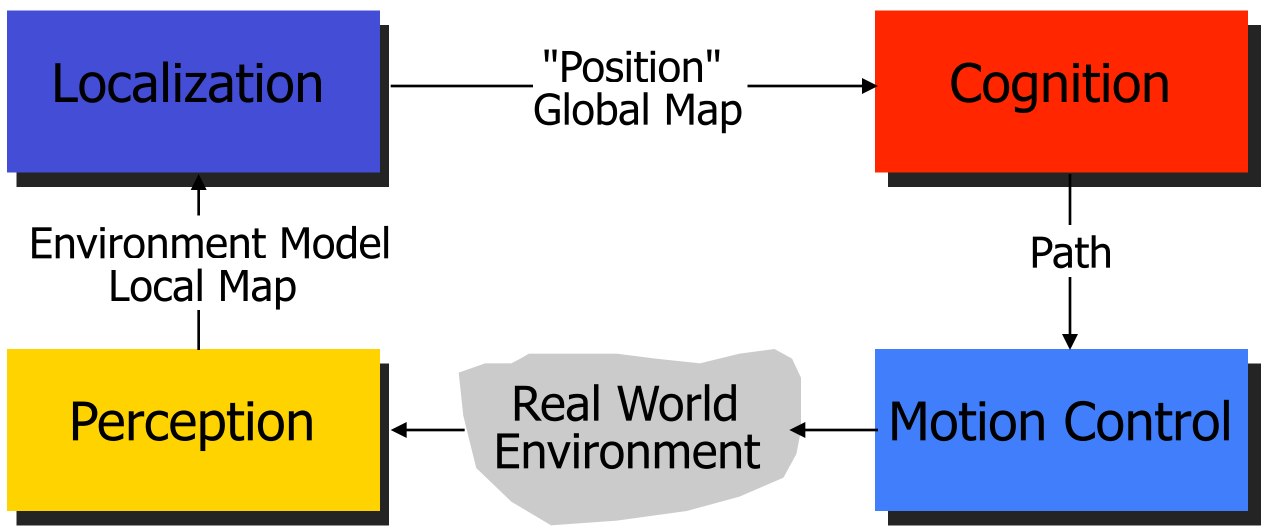
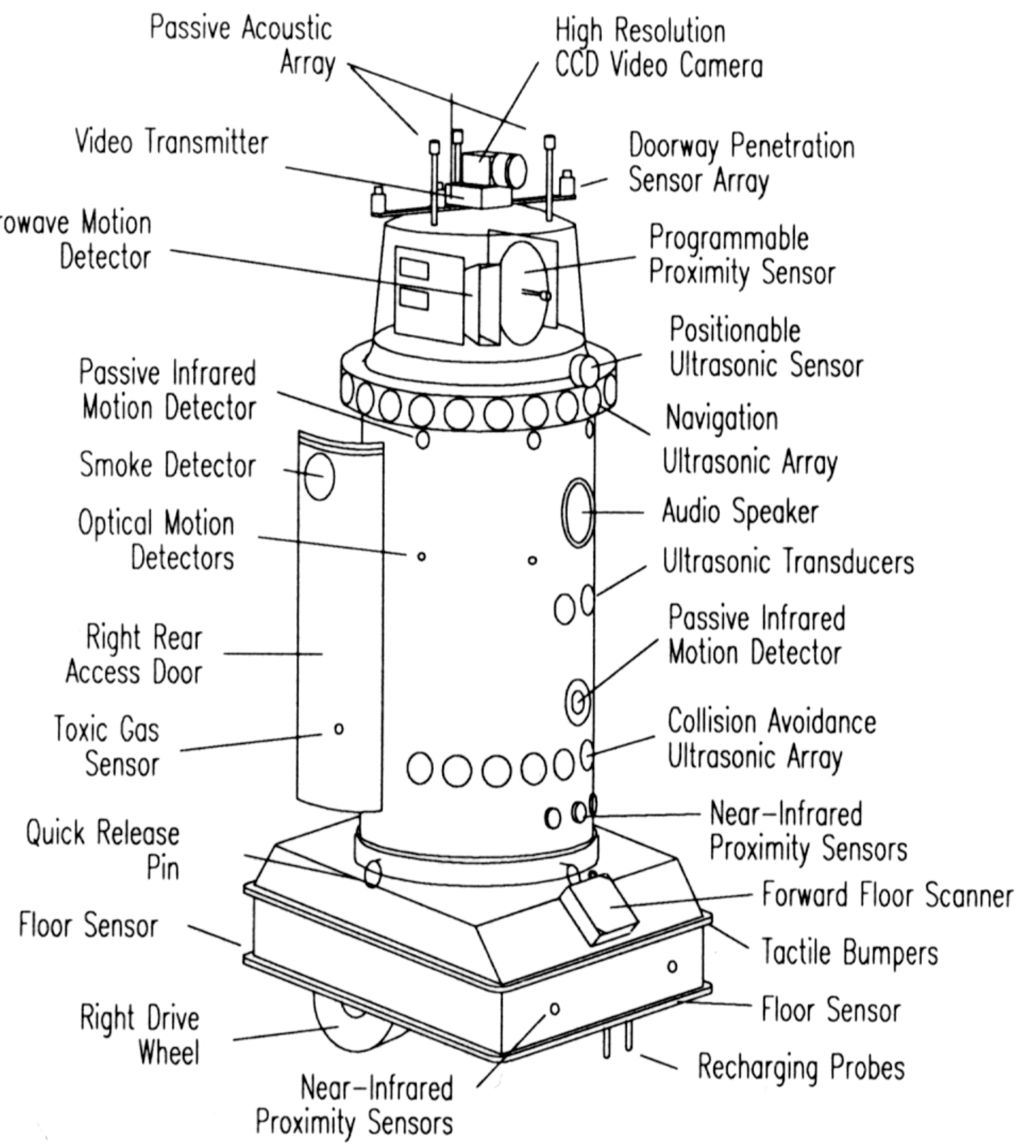
Perception

- Sensors
- Uncertainty
- Features
- Introduction Chapter 4 [Nourbaksh & Siegwart]
- Introductory slides (courtesy [Nourbaksh & Siegwart])



Example Robart II, H.R. Everett



BibaBot, BlueBotics SA, Switzerland

IMU
Inertial Measurement Unit

Emergency Stop Button

Wheel Encoders

Omnidirectional Camera

Pan-Tilt Camera

Sonar Sensors

Laser Range Scanner

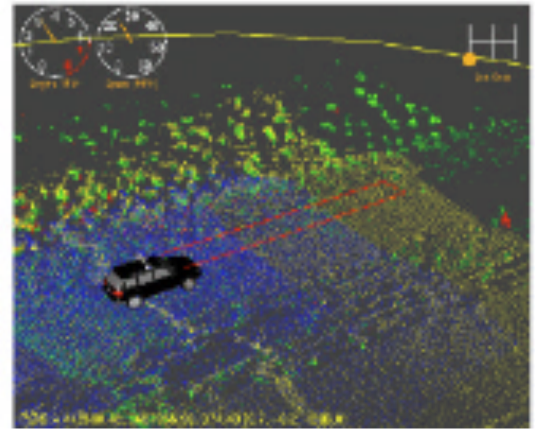
Bumper

Robotic Navigation

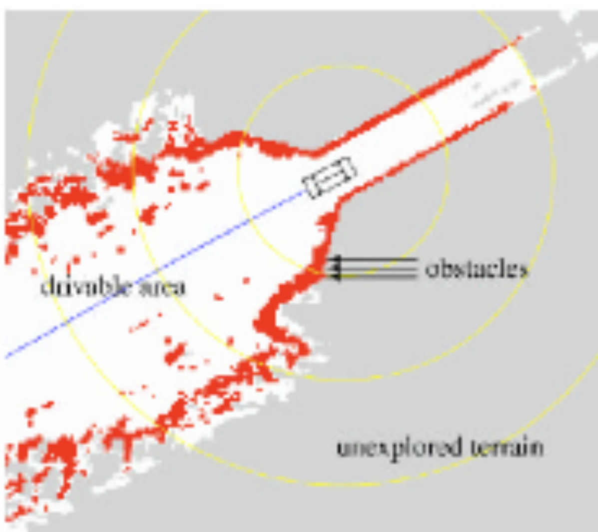
- Stanford Stanley Grand Challenge
- Outdoors unstructured env., single vehicle

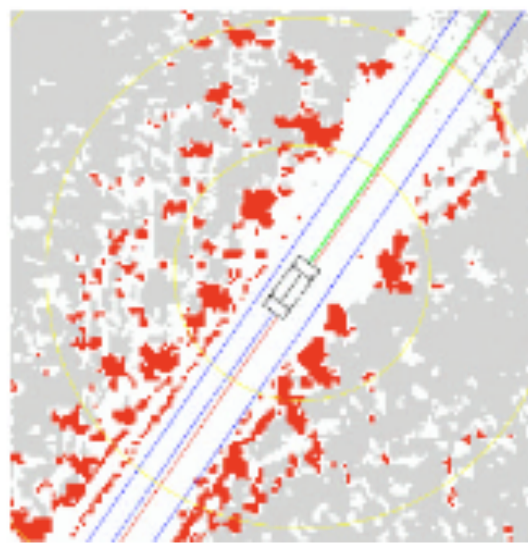
- Urban Challenge
- Outdoors structured env., mixed traffic, traffic rules

- Terrain mapping using lasers



- Determining obstacle course





Classification of Sensors

- Proprioceptive sensors
  - measure values internally to the system (robot),
  - e.g. motor speed, wheel load, heading of the robot, battery status
- Exteroceptive sensors
  - information from the robots environment
  - distances to objects, intensity of the ambient light, unique features.
- Passive sensors
  - energy coming for the environment
- Active sensors
  - emit their proper energy and measure the reaction
  - better performance, but some influence on environment

Role of Perception in Robotics

- Where am I relative to the world?
  - sensors: vision, stereo, range sensors, acoustics
  - problems: scene modeling/classification/recognition
  - integration: localization/mapping algorithms (e.g. SLAM)

- What is around me?
  - sensors: vision, stereo, range sensors, acoustics, sounds, smell
  - problems: object recognition, structure from x, qualitative modeling
  - integration: collision avoidance/navigation, learning

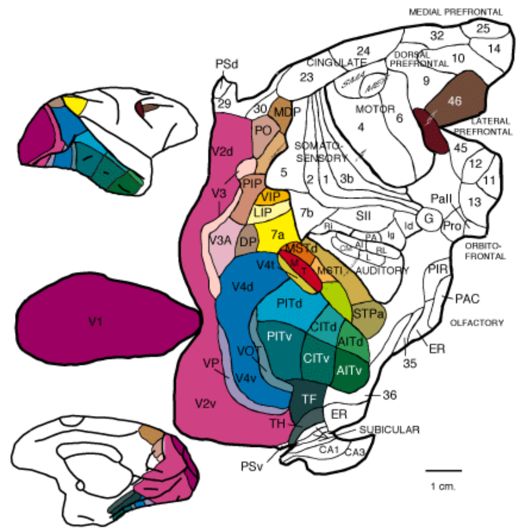
Role of Perception in Robotics

- How can I safely interact with environment (including people!)?
  - sensors: vision, range, haptics (force+tactile)
  - problems: structure/range estimation, modeling, tracking, materials, size, weight, inference
  - integration: navigation, manipulation, control, learning

- How can I solve "new" problems (generalization)?
  - sensors: vision, range, haptics, undefined new sensor
  - problems: categorization by function/shape/context/??
  - integrate: inference, navigation, manipulation, control, learning

Challenges/Issues

- About 60% of our brain is devoted to vision
- We see immediately and can form and understand images instantly



- Detailed representations are often not necessary
- Different approaches in the past Animate Vision (biologically inspired), Purposive Vision (depending on the task/purpose)

Jana Kosecka

Visual Perception Topics

Techniques

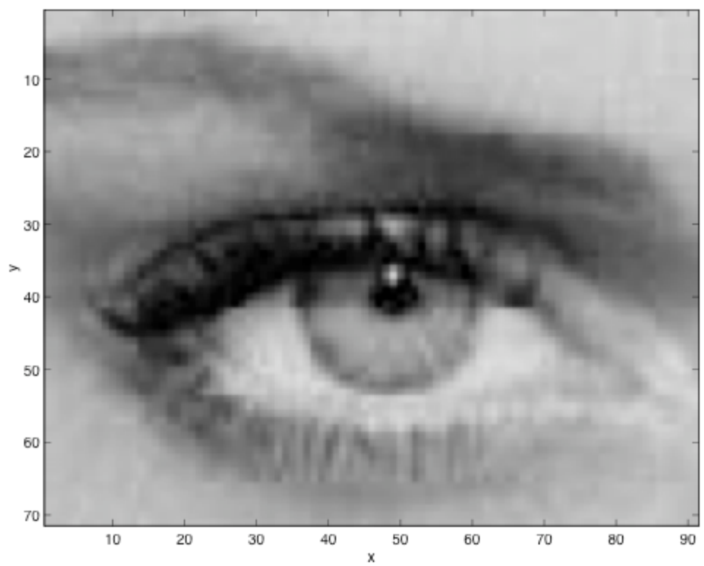
- Computational Stereo
- Feature detection and matching
- Motion tracking and visual feedback

Applications in Robotics:

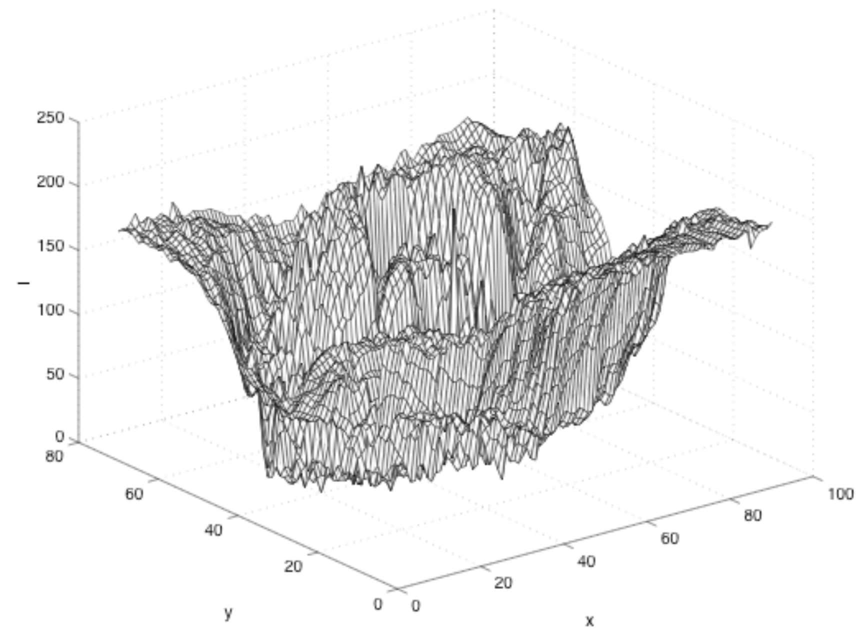
- range sensing, Obstacle detection, environment interaction
- Mapping, registration, localization, recognition
- Manipulation

Image - Apperance

Image



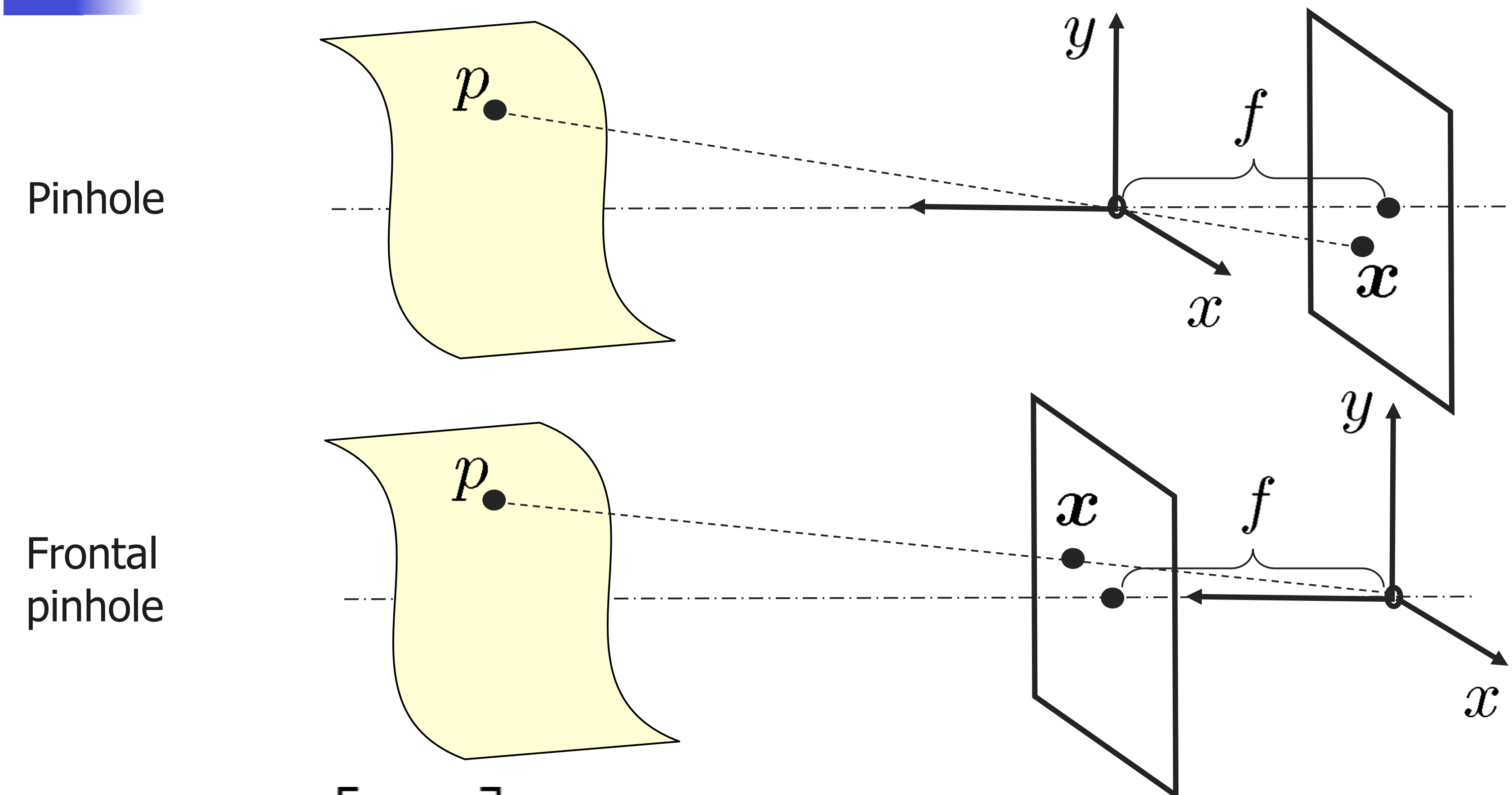
Brightness values



I(x,y)

Jana Kosecka

Image Formation



$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow \boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

J. Kosecka, GMU

Pinhole Camera Model

- Image coordinates are nonlinear function of world coordinates
- Relationship between coordinates in the camera frame and sensor plane

2-D coordinates
$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix}$$

Homogeneous coordinates

$$\boldsymbol{x} \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z}\begin{bmatrix} fX \\ fY \\ Z \end{bmatrix}, \qquad \boldsymbol{X} \rightarrow \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix},$$

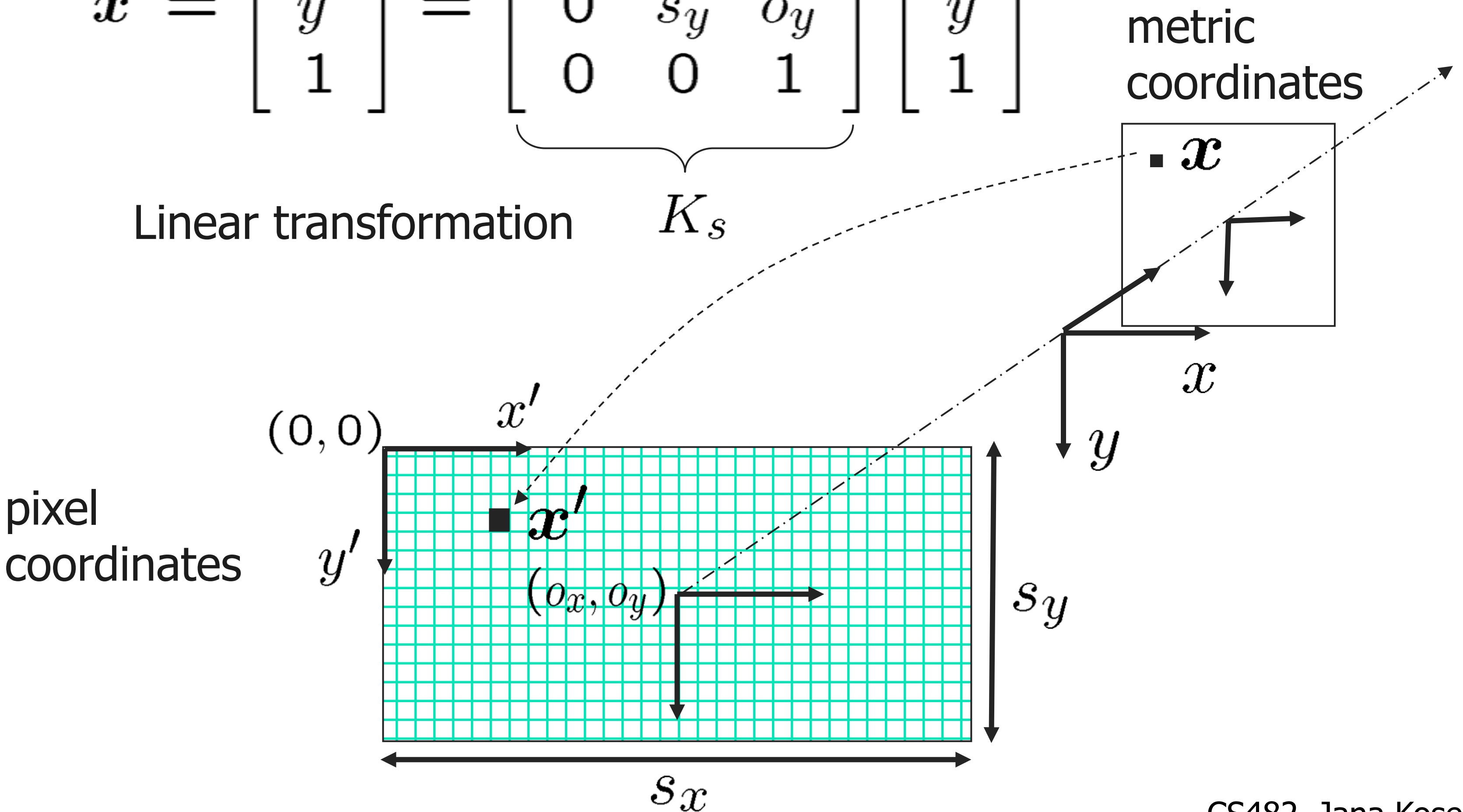
$$Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{K_f} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\Pi_0} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Image Coordinates

- Relationship between coordinates in the sensor plane and image

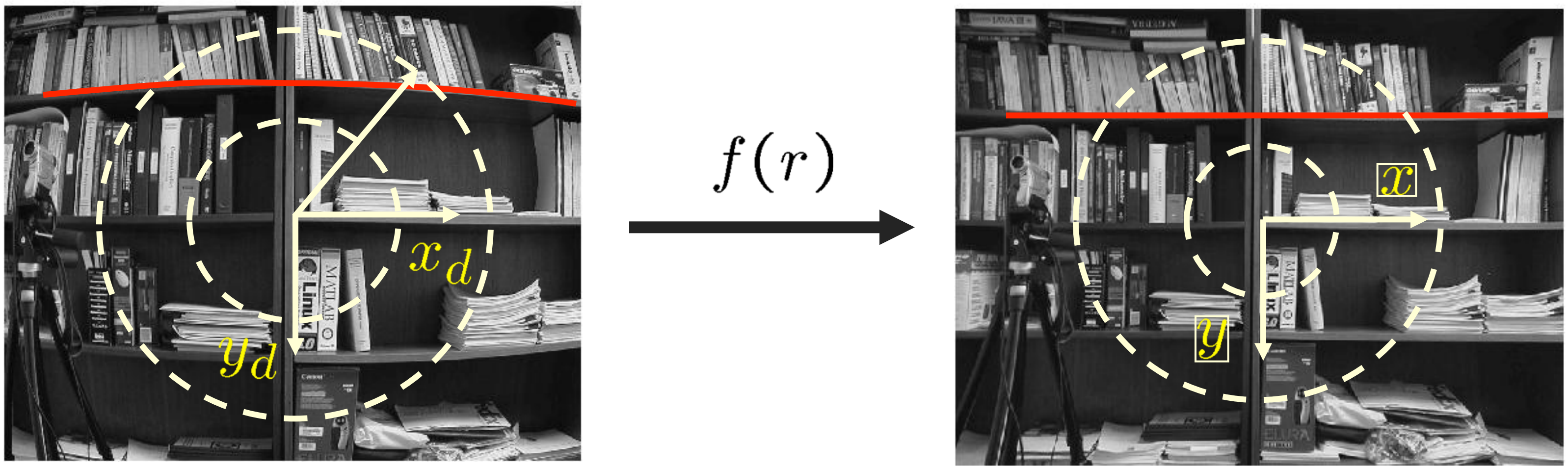
$$\boldsymbol{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{K_s} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Linear transformation



Camera parameters – Radial Distortion

Nonlinear transformation along the radial direction



$$
\begin{aligned}
\boldsymbol{x} &= c + f(r)(\boldsymbol{x}_d - c), \qquad r = \|\boldsymbol{x}_d - c\| \\
f(r) &= 1 + a_1 r + a_2 r^2 + a_3 r^3 + a_4 r^4 + \cdots
\end{aligned}
$$

Distortion correction: make lines straight

Calibration Matrix and Camera Model

- Relationship between coordinates in the world frame and image
- Intrinsic parameters

Pinhole camera

$$\lambda \boldsymbol{x} = K_f \Pi_0 \boldsymbol{X}$$

Pixel coordinates

$$\boldsymbol{x}' = K_s \boldsymbol{x}$$

- Adding transformation between camera coordinate systems and world coordinate system
- Extrinsic Parameters

$$\lambda \boldsymbol{x}' = \begin{bmatrix} f s_x & f s_\theta & o_x \\ 0 & f s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

$$\lambda \boldsymbol{x} = K_f \Pi_0 g \boldsymbol{X} = \Pi \boldsymbol{X}$$

Image of a Point

Homogeneous coordinates of a 3-D point p

$$\boldsymbol{X} = [X, Y, Z, W]^T \in \mathbb{R}^4, \quad (W = 1)$$

Homogeneous coordinates of its 2-D image

$$\boldsymbol{x} = [x, y, z]^T \in \mathbb{R}^3, \quad (z = 1)$$

Projection of a 3-D point to an image plane

$$\boxed{\lambda \boldsymbol{x} = \Pi \boldsymbol{X}}$$

$$\lambda \in \mathbb{R}, \ \Pi = [R, T] \in \mathbb{R}^{3\times 4}$$

$$\boxed{\lambda \boldsymbol{x}' = \Pi \boldsymbol{X}}$$

$$\lambda \in \mathbb{R}, \ \Pi = [KR, KT] \in \mathbb{R}^{3\times 4}$$

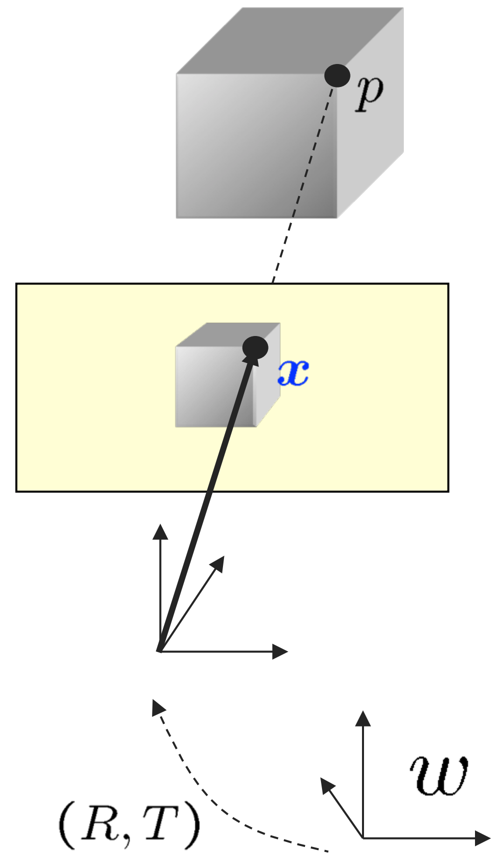
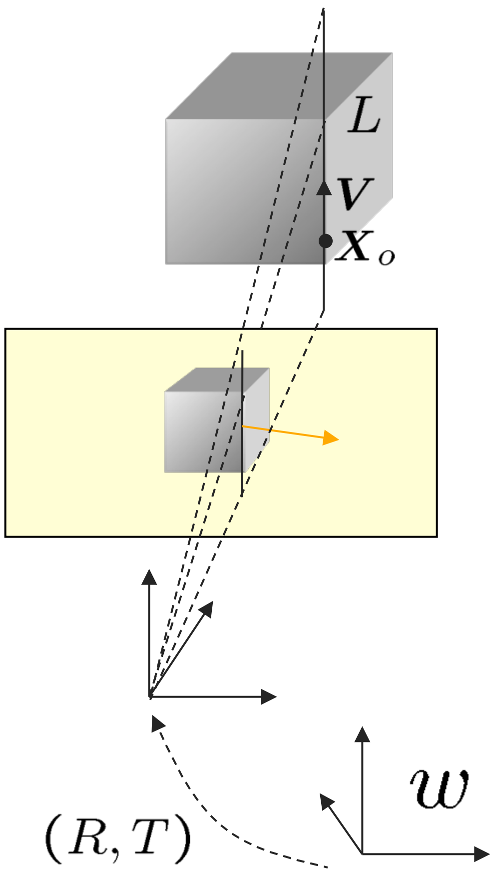


Image of a Line

Homogeneous representation of a 3-D line L

$$\boldsymbol{X} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} X_o \\ Y_o \\ Z_o \\ 1 \end{bmatrix} + \mu \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ 0 \end{bmatrix}, \quad \mu \in \mathbb{R}$$



Homogeneous representation of its 2-D image

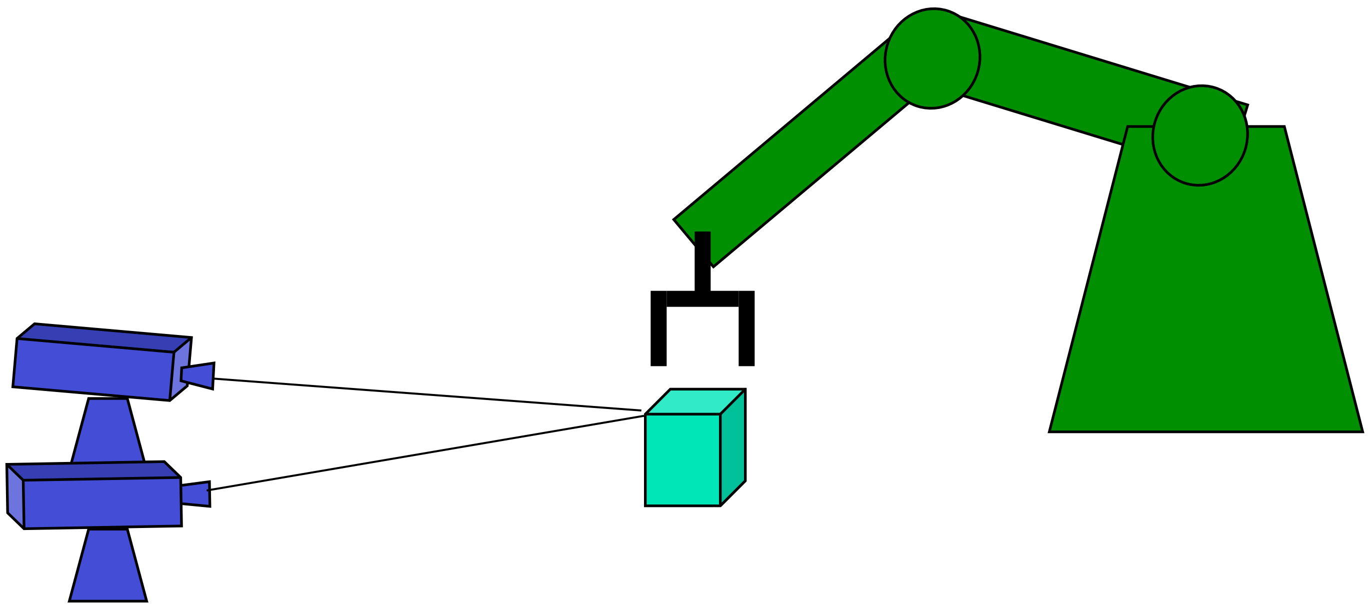
$$\boldsymbol{l} = [a, b, c]^T \in \mathbb{R}^3$$

Projection of a 3-D line to an image plane

$$\boxed{\boldsymbol{l}^T \boldsymbol{x} = \boldsymbol{l}^T \Pi \boldsymbol{X} = 0}$$

$$\Pi = [KR, KT] \in \mathbb{R}^{3 \times 4}$$

What is Computational Stereo?



Viewing the same physical point from two different viewpoints allows depth from triangulation

Computational Stereo

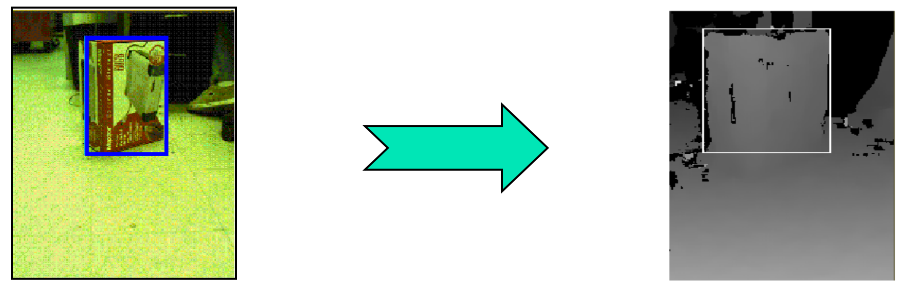
- Much of geometric vision is based on information from 2 (or more) camera locations
- Hard to recover 3D information from a single 2D image without extra knowledge
- Motion and stereo (multiple cameras) are both common in the world

- Stereo vision is ubiquitous in nature
  (oddly, nearly 10% of people are stereo blind)

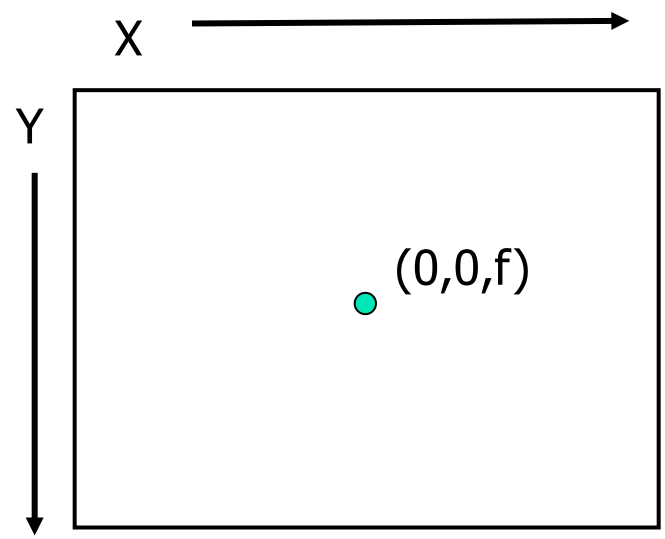
- Stereo involves the following *three problems*:
  1. calibration
  2. matching (correspondence problem)
  3. reconstruction (reconstruction problem)

Binocular Stereo System: Geometry

- **GOAL:** Passive 2-camera system using triangulation to generate a depth map of a world scene.



- **Depth map:** z=f(x,y) where x,y are coordinates one of the image planes and z is the height above the respective image plane.
    - Note that for stereo systems which differ only by an offset in x, the v coordinates (projection of y) is the same in both images!
    - Note we must convert from image (pixel) coordinates to external coordinates -- **requires calibration**

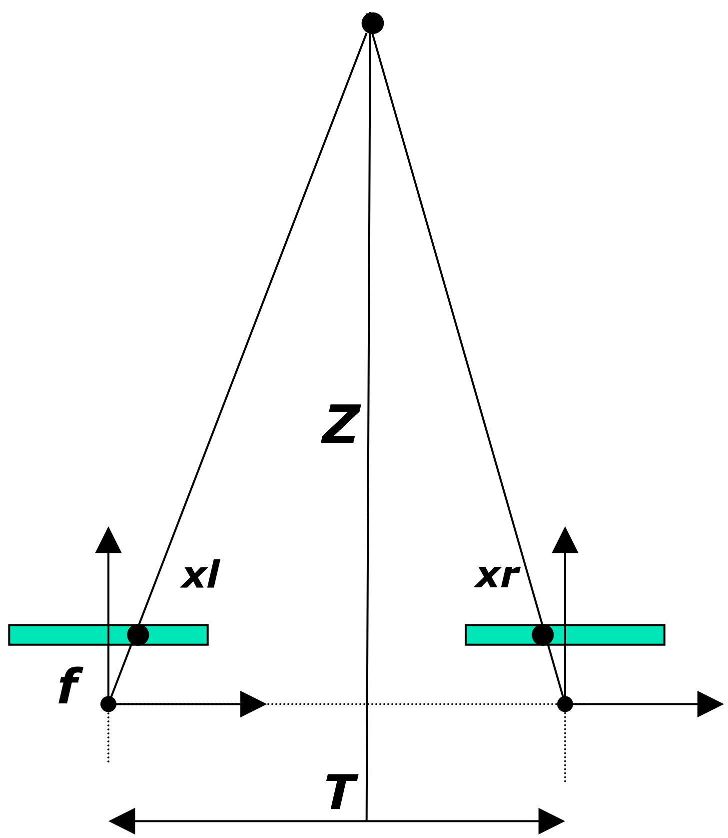


4 intrinsic parameters convert from pixel to metric values

s_x s_y c_x c_y

Stereo Configuration

- Images are scan-aligned
- Disparity between two images – inversely proportional to depth
- Disparity – difference between x-coordinates of a feature
- Triangle similarity



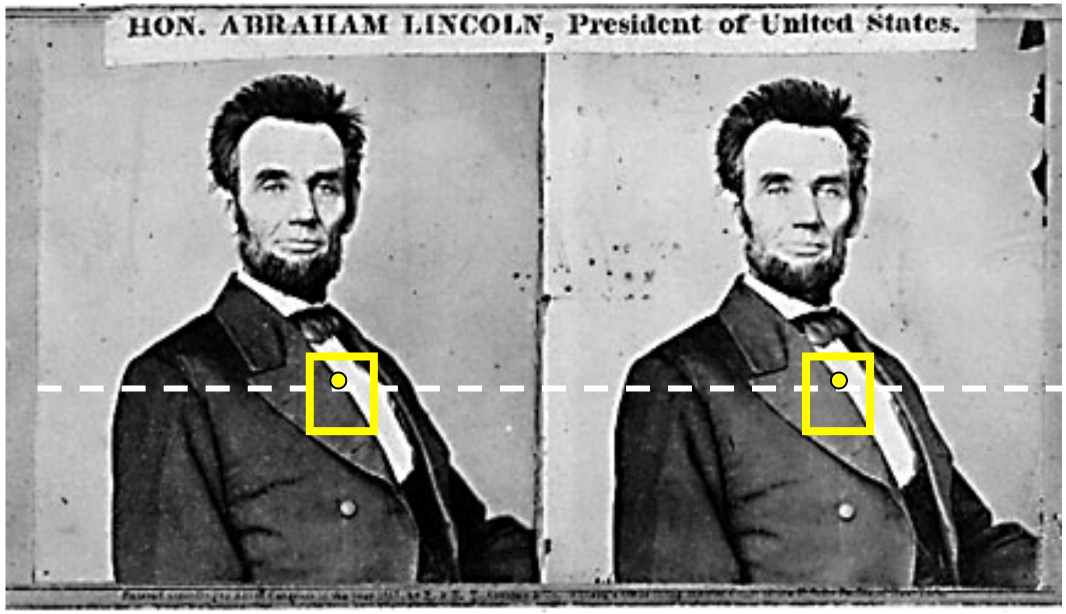
$$\frac{Z}{T} = \frac{Z-f}{T-\mathbf{x}_l-\mathbf{x}_r}$$

$$Z = \frac{fT}{\text{disparity}}$$

Stereo Vision

- Distance is inversely proportional to disparity
  - closer objects can be measured more accurately
- Disparity is proportional to baseline
  - For a given disparity error, the accuracy of the depth estimate increases with increasing baseline baseline
  - However, as baseline is increased, some objects may appear in one camera, but not in the other.
  - Image resolution is also a factor

Stereo Matching – Stereo Correspondence



For each epipolar line (scanline)

For each pixel in the left image

- compare with every pixel on same epipolar line in right image
- pick pixel with minimum match cost
- This will never work, so:
- Match Windows

Region based Similarity Metric

- Sum of squared differences

$$SSD(h) = \sum_{\tilde{\mathbf{x}} \in W(\mathbf{x})} \| I_1(\tilde{\mathbf{x}}) - I_2(h(\tilde{\mathbf{x}}) \|^2$$

- Normalize cross-correlation

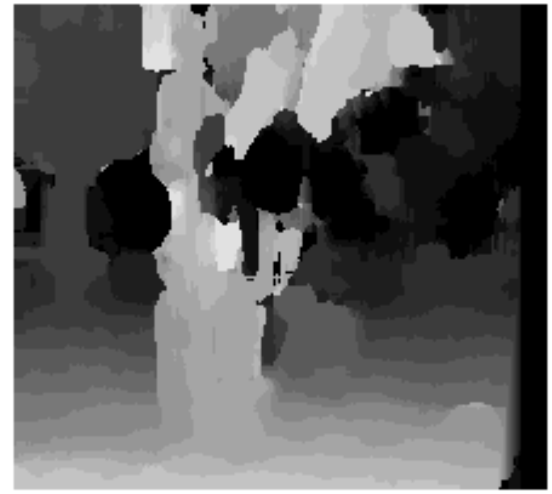
$$NCC(h) = \frac{\sum_{W(\mathbf{x})} (I_1(\tilde{\mathbf{x}}) - \bar{I}_1)(I_2(h(\tilde{\mathbf{x}})) - \bar{I}_2))}{\sqrt{\sum_{W(\mathbf{x})} (I_1(\tilde{\mathbf{x}}) - \bar{I}_1)^2 \sum_{W(\mathbf{x})} (I_2(h(\tilde{\mathbf{x}})) - \bar{I}_2)^2)}}$$

- Sum of absolute differences

$$SAD(h) = \sum_{\tilde{\mathbf{x}} \in W(\mathbf{x})} |I_1(\tilde{\mathbf{x}}) - I_2(h(\tilde{\mathbf{x}}))|$$

Window size

W = 3



W = 20

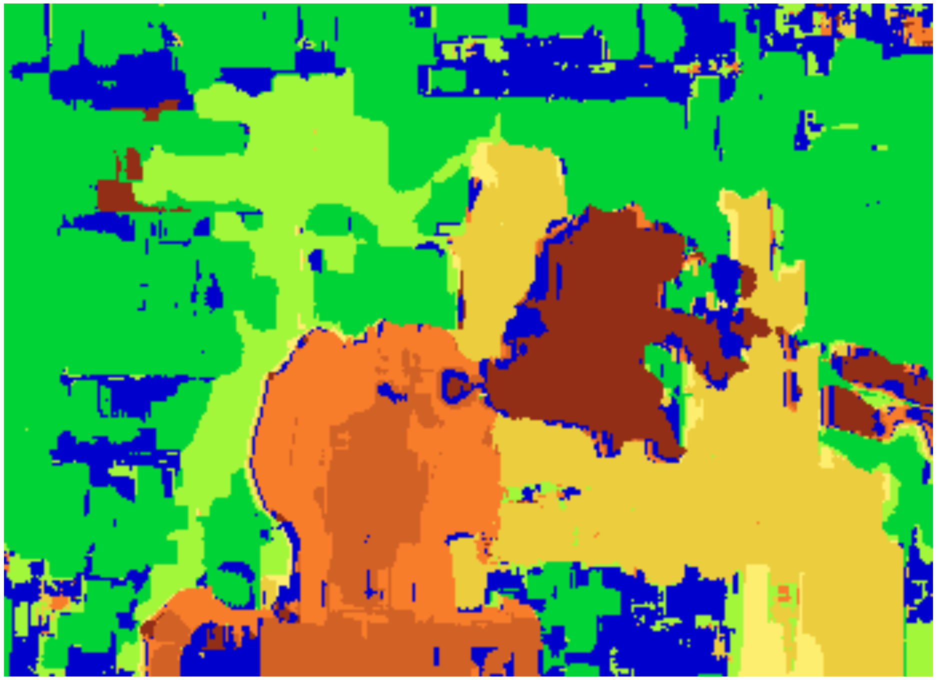
- Effect of window size

With adaptive window

- T. Kanade and M. Okutomi, *A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment*,, Proc. International Conference on Robotics and Automation, 1991.
- D. Scharstein and R. Szeliski. Stereo matching with nonlinear diffusion. International Journal of Computer Vision, 28(2): 155-174, July 1998

(S. Seitz) Jana Kosecka

Results with window correlation



Window-based matching
(best window size)

Ground truth

(slide courtesy S. Seitz)

Results with better method

State of the art method

Boykov et al., Fast Approximate Energy Minimization via Graph Cuts, International Conference on Computer Vision, September 1999.

Ground truth

(slide courtesy S. Seitz)

Jana Kosecka

Applications of Real-Time Stereo

- Mobile robotics
  - Detect the structure of ground; detect obstacles; convoying

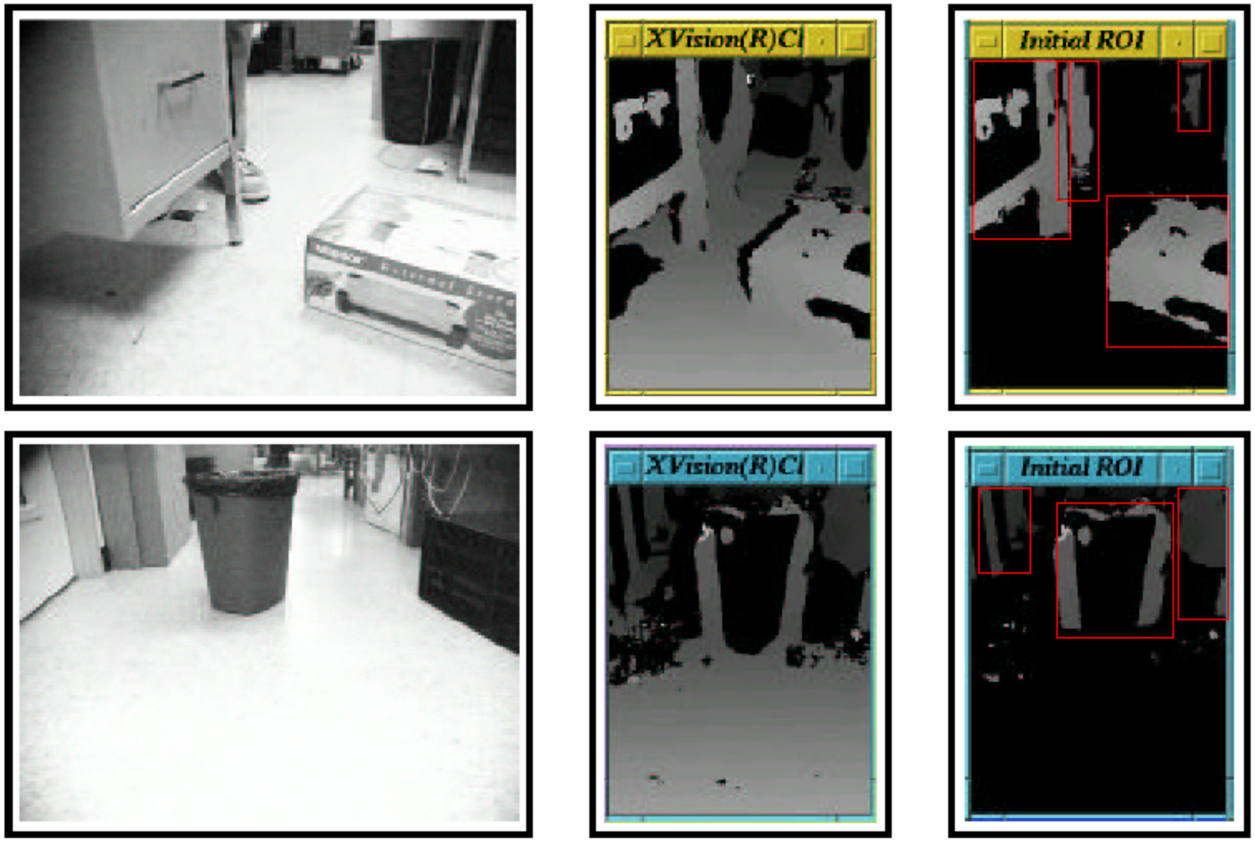
- Graphics/video
  - Detect foreground objects and matte in other objects (super-matrix effect)

- Surveillance
  - Detect and classify vehicles on a street or in a parking garage

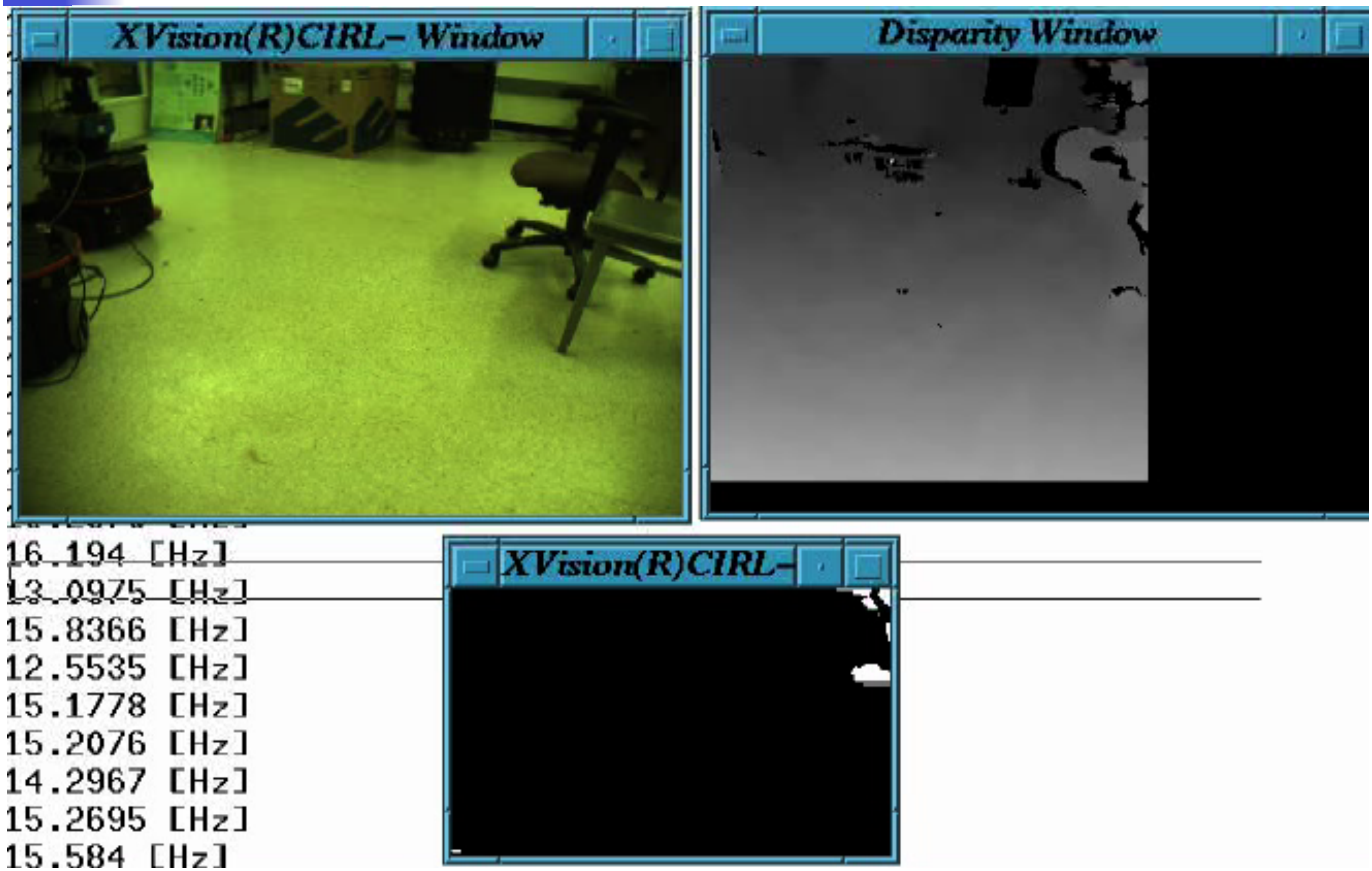
- Medical
  - Measurement (e.g. sizing tumors)
  - Visualization (e.g. register with pre-operative CT)

Obstacle Detection (cont'd)

Observation: Removing the ground plane immediately exposes obstacles



Applications of Real-Time Stereo

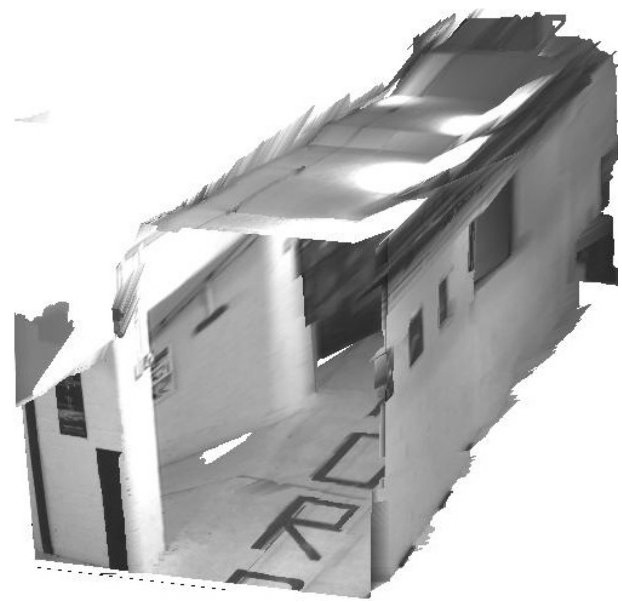


GMU building




Oxford corridor

using 6 images



3D model

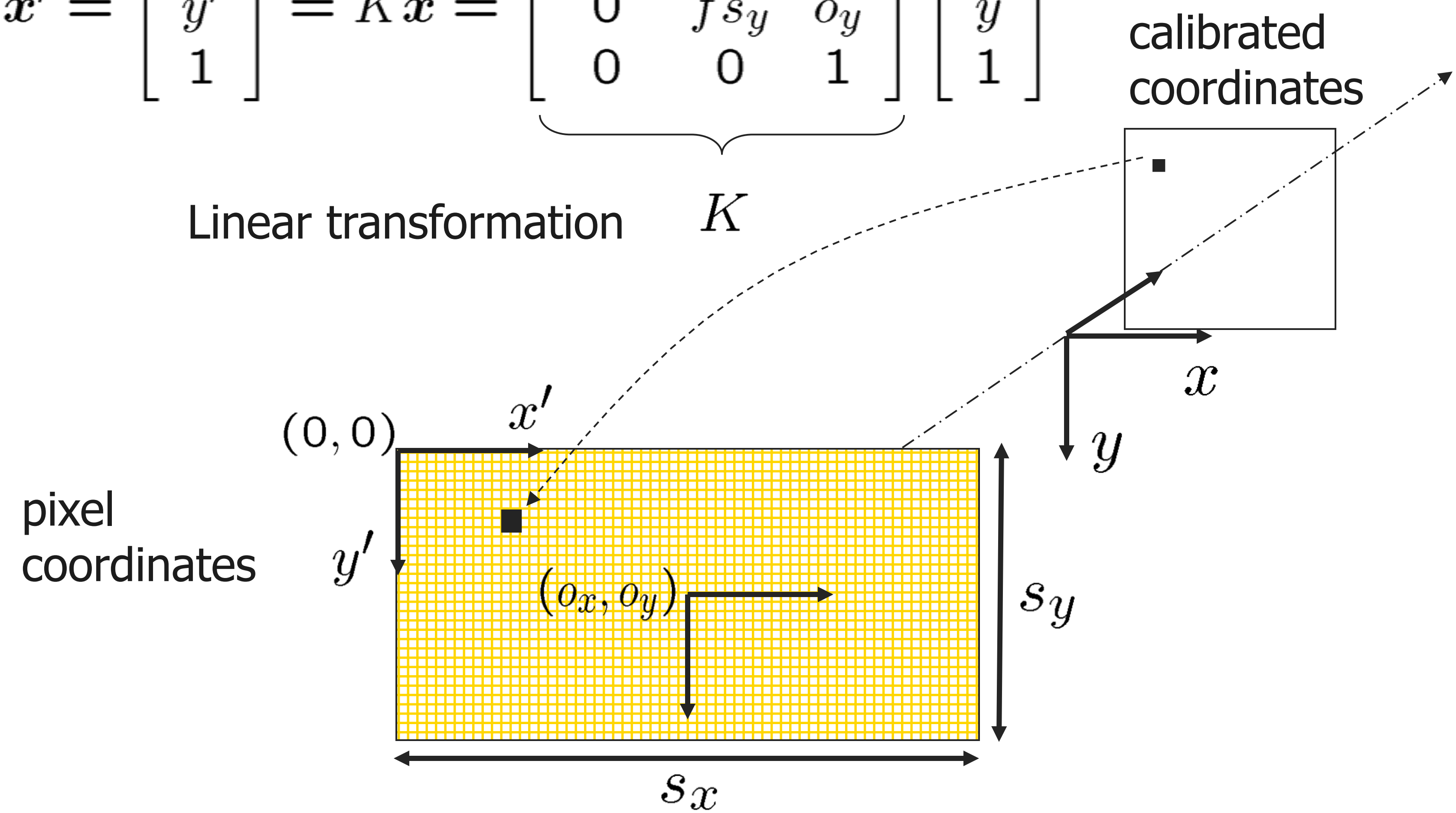
Feature based stereo

- Instead of matching each pixel
- Match features in the image
- What are good features ? – next lecture
- Examples of features – line matching, point matching region matching

Uncalibrated Camera

$$\boldsymbol{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K\boldsymbol{x} = \underbrace{\begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}}_{K} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Linear transformation

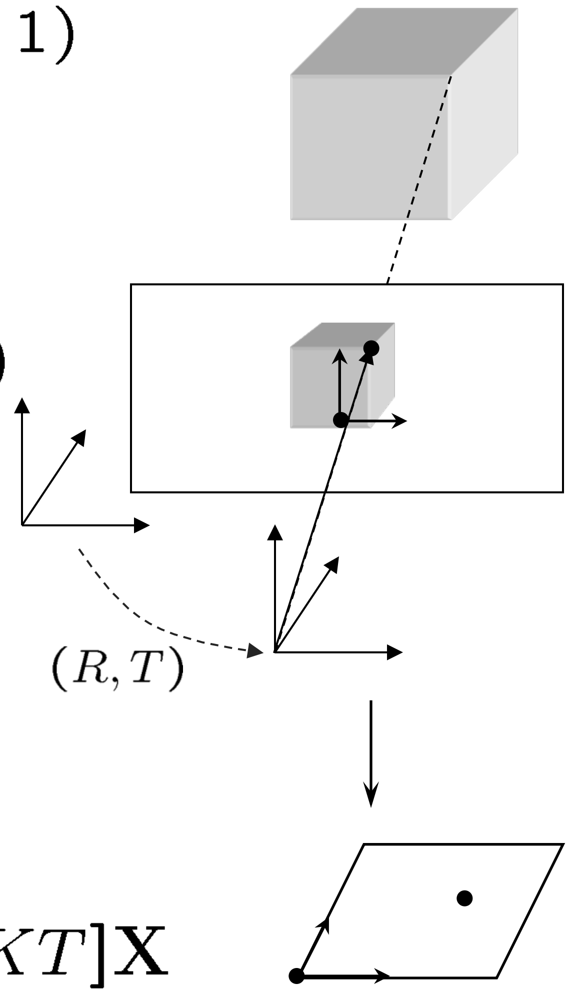


Uncalibrated Camera

$$\mathbf{X} = [X, Y, Z, W]^T \in \mathbb{R}^4, \quad (W = 1)$$

Calibrated camera

- Image plane coordinates $\mathbf{x} = [x, y, 1]^T$
- Camera extrinsic parameters $g = (R, T)$
- Perspective projection $\lambda \mathbf{x} = [R, T]\mathbf{X}$



Uncalibrated camera

- Pixel coordinates $\mathbf{x}' = K\mathbf{x}$
- Projection matrix $\lambda \mathbf{x}' = \Pi \mathbf{X} = [KR, KT]\mathbf{X}$

Calibration with a Rig

Use the fact that both 3-D and 2-D coordinates of feature points on a pre-fabricated object (e.g., a cube) are known.

Calibration with a Rig

- Given 3-D coordinates on known object

$$\lambda \mathbf{x}' = [KR, KT]\mathbf{X} \Longrightarrow \lambda \mathbf{x}' = \Pi \mathbf{X}$$

$$\lambda \begin{bmatrix} x^i \\ y^i \\ 1 \end{bmatrix} = \begin{bmatrix} \pi_1^T \\ \pi_2^T \\ \pi_3^T \end{bmatrix} \begin{bmatrix} X^i \\ Y^i \\ Z^i \\ 1 \end{bmatrix}$$

- Eliminate unknown scales

$$\begin{aligned} x^i(\pi_3^T \mathbf{X}) &= \pi_1^T \mathbf{X}, \\ y^i(\pi_3^T \mathbf{X}) &= \pi_2^T \mathbf{X} \end{aligned}$$

- Recover projection matrix $\Pi = [KR, KT] = [R', T']$

$$\min \|M\Pi^s\|^2 \quad \text{subject to} \quad \|\Pi^s\|^2 = 1$$

$$\Pi^s = [\pi_{11}, \pi_{21}, \pi_{31}, \pi_{12}, \pi_{22}, \pi_{32}, \pi_{13}, \pi_{23}, \pi_{33}, \pi_{14}, \pi_{24}, \pi_{34}]^T$$

- Factor the into $R \in SO(3)$ and using QR decomposition
- Solve for translation $T = K^{-1}T'$

More details

- Direct calibration by recovering and decomposing the projection matrix

$$\lambda \begin{bmatrix} x^i \\ y^i \\ 1 \end{bmatrix} = \begin{bmatrix} \pi_1^T \\ \pi_2^T \\ \pi_3^T \end{bmatrix} \begin{bmatrix} X^i \\ Y^i \\ Z^i \\ 1 \end{bmatrix} \rightarrow Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \pi_{11} & \pi_{12} & \pi_{13} & \pi_{14} \\ \pi_{21} & \pi_{22} & \pi_{23} & \pi_{24} \\ \pi_{31} & \pi_{32} & \pi_{33} & \pi_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$x_i = \frac{\pi_{11}X_i + \pi_{12}Y_i + \pi_{13}Z_i + \pi_{14}}{\pi_{31}X_i + \pi_{32}Y_i + \pi_{33}Z_i + \pi_{34}} \quad y_i = \frac{\pi_{21}X_i + \pi_{22}Y_i + \pi_{23}Z_i + \pi_{24}}{\pi_{31}X_i + \pi_{32}Y_i + \pi_{33}Z_i + \pi_{34}}$$

$$x_i(\pi_{31}X_i + \pi_{32}Y_i + \pi_{33}Z_i + \pi_{34}) = \pi_{11}X_i + \pi_{12}Y_i + \pi_{13}Z_i + \pi_{14}$$
$$y_i(\pi_{31}X_i + \pi_{32}Y_i + \pi_{33}Z_i + \pi_{34}) = \pi_{21}X_i + \pi_{22}Y_i + \pi_{23}Z_i + \pi_{24}$$

$$\begin{aligned} x^i(\pi_3^T \mathbf{X}) &= \pi_1^T \mathbf{X}, \\ y^i(\pi_3^T \mathbf{X}) &= \pi_2^T \mathbf{X} \end{aligned}$$

2 constraints per point

$$[X_i, Y_i, Z_i, 1, 0, 0, 0, 0, -x_iX_i, -x_iY_i, -x_iZ_i, -x_i]\Pi_s = 0$$
$$[0, 0, 0, 0, X_i, Y_i, Z_i, 1, -y_iX_i, -y_iY_i, -y_iZ_i, -y_i]\Pi_s = 0$$

$$\Pi_s = [\pi_{11}, \pi_{12}, \pi_{13}, \pi_{14}, \pi_{21}, \pi_{22}, \pi_{23}, \pi_{24}, \pi_{31}, \pi_{32}, \pi_{33}, \pi_{34}]^T$$

More details

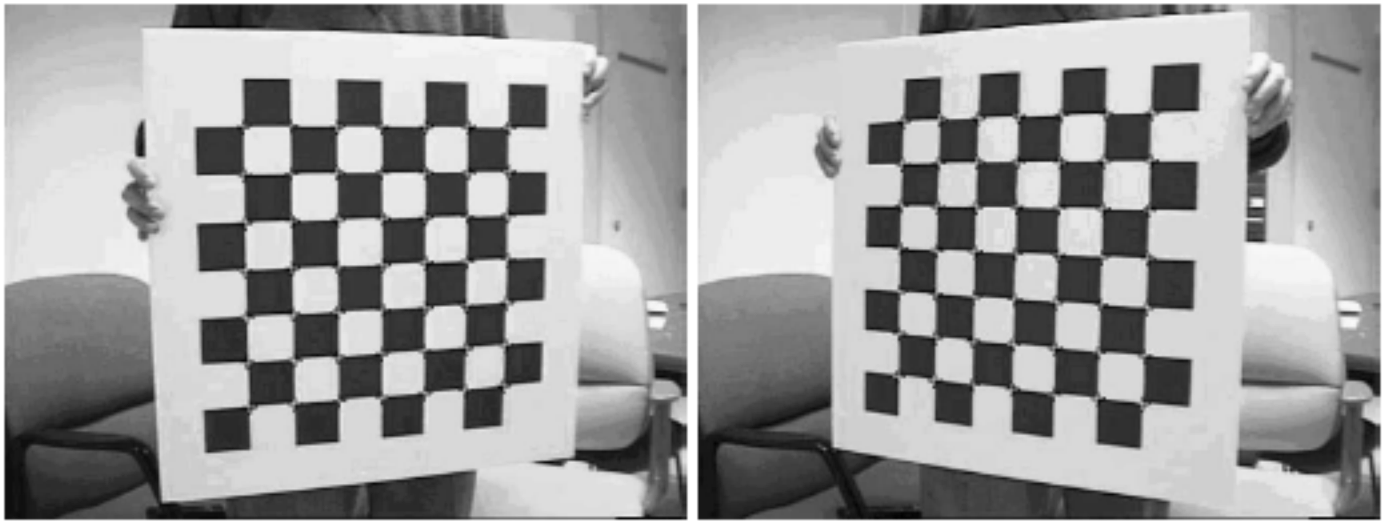
- Recover projection matrix $\Pi = [KR, KT] = [R', T']$

$$\min \|M\Pi^s\|^2 \quad \text{subject to} \quad \|\Pi^s\|^2 = 1$$

$$\Pi^s = [\pi_{11}, \pi_{21}, \pi_{31}, \pi_{12}, \pi_{22}, \pi_{32}, \pi_{13}, \pi_{23}, \pi_{33}, \pi_{14}, \pi_{24}, \pi_{34}]^T$$

- Collect the constraints from all N points into matrix M (2N x 12)
- Solution eigenvector associated with the smallest eigenvalue $M^T M$
- Unstack the solution and decompose into rotation and translation
- Factor the R' into $R \in SO(3)$ and K using QR decomposition
- Solve for translation $T = K^{-1}T'$

Calibration with a planar pattern



$$H \doteq K[r_1, r_2, T] \quad \in \mathbb{R}^{3\times 3} \qquad\qquad \lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K[r_1, r_2, T] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix},$$

To eliminate unknown depth, multiply both sides by $\widehat{x'}$

$$\widehat{x'} H [X, Y, 1]^T = 0.$$

Calibration with a planar pattern

$$[h_1, h_2] \sim K[r_1, r_2]$$

$$K^{-1}[h_1, h_2] \sim [r_1, r_2]$$

Because r_1, r_2 are orthogonal and unit norm vectors of rotation matrix
We get the following two constraints

$$h_1^T K^{-T} K^{-1} h_2 = 0, \quad h_1^T K^{-T} K^{-1} h_1 = h_2^T K^{-T} K^{-1} h_2.$$

- We want to recover S $\quad S = K^{-T} K^{-1} \quad\quad e_1^T S e_2 = 0$
- Unknowns in K (S) $\quad f s_x, f s_y, f s_\theta, o_x, o_y$

Skew s_θ is often close 0 -> 4 unknowns

- S is symmetric matrix (6 unknowns) in general we need at least 3 views
- To recover S (2 constraints per view) - S can be recovered linearly
- Get K by Cholesky decomposition of directly from entries of S

Alternative camera models/projections

Orthographic projection

$$\mathbf{x}' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

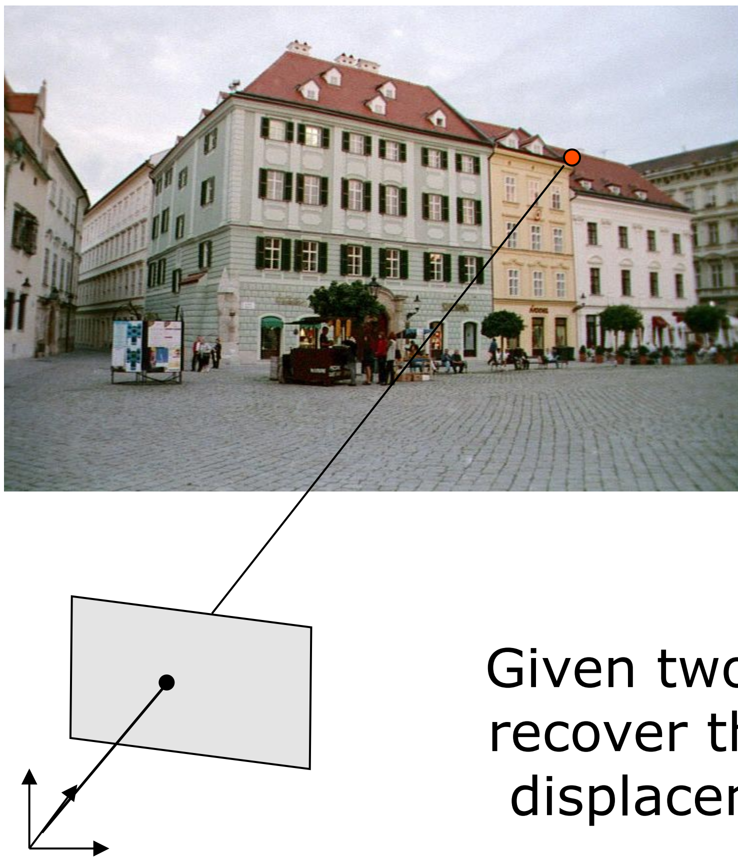
Scaled orthographic projection

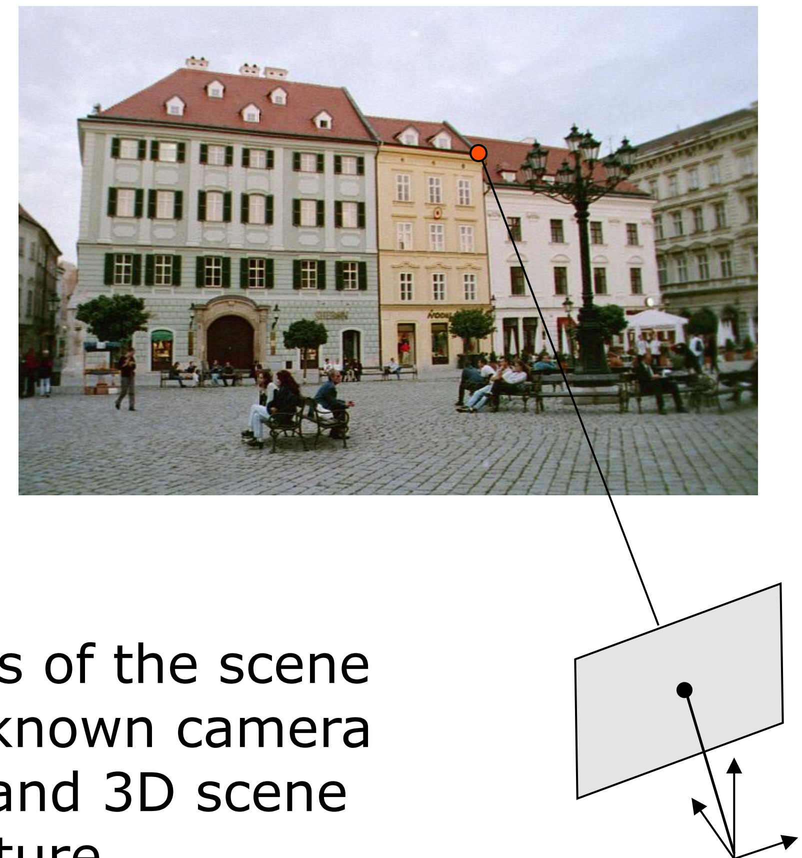
$$\mathbf{x}' = s \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Affine camera model

$$\mathbf{x}' = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{33} & a_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

General Formulation





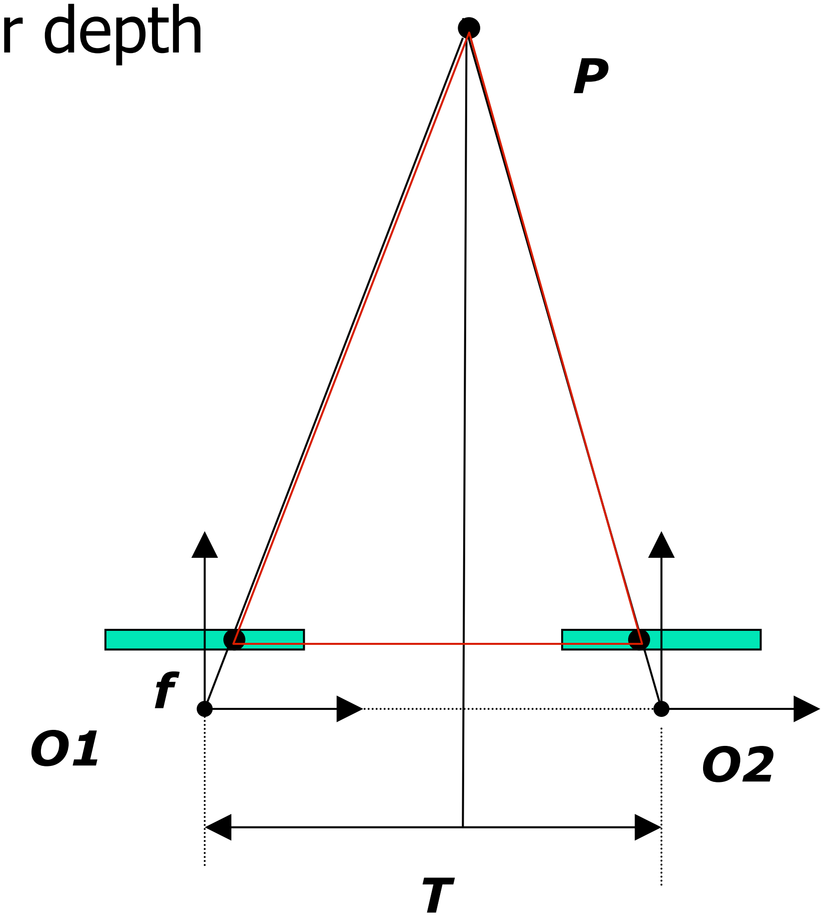
Given two views of the scene recover the unknown camera displacement and 3D scene structure

Stereo

- What if the motion between cameras is not known ?

Canonical Stereo Configuration

- Assumes (two) cameras
- Known positions and focal lengths
- Recover depth



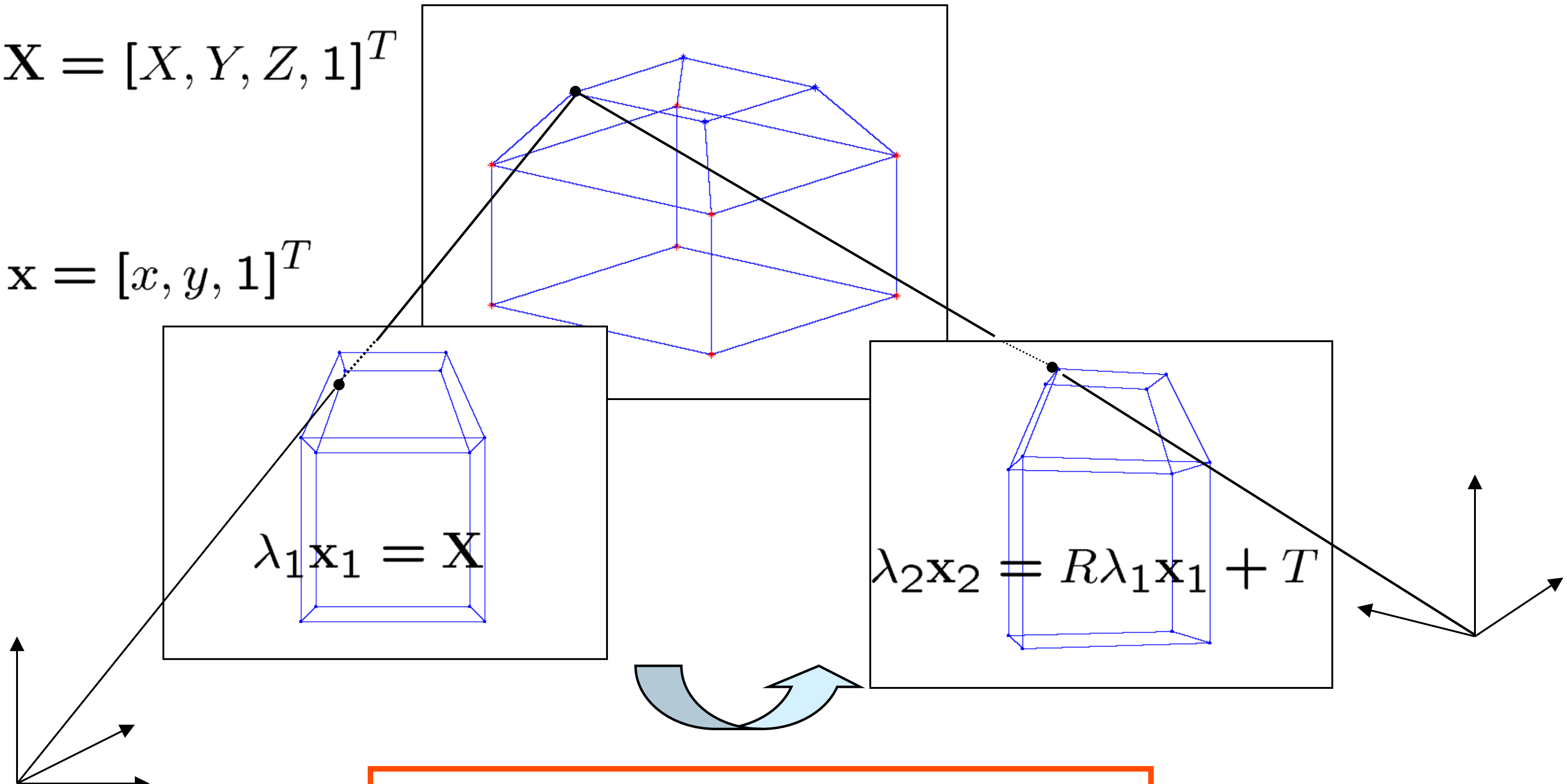
$$\frac{Z}{T} = \frac{Z-f}{T-\mathbf{x}_l-\mathbf{x}_r}$$

$$Z = \frac{fT}{\text{disparity}}$$

Rigid Body Motion – Two Views

$$\mathbf{X} = [X, Y, Z, 1]^T$$

$$\mathbf{x} = [x, y, 1]^T$$



$$\boxed{\lambda_2 \mathbf{x}_2 = R\lambda_1 \mathbf{x}_1 + T}$$

3D Structure and Motion Recovery

Euclidean transformation

$$\lambda_2 \mathbf{x}_2 = R \lambda_1 \mathbf{x}_1 + T$$

measurements unknowns

$$\sum_{j=1}^{n} \|\mathbf{x}_1^j - \pi(R_1, T_1, \mathbf{X})\|^2 + \|\mathbf{x}_2^j - \pi(R_2, T_2, \mathbf{X})\|^2$$

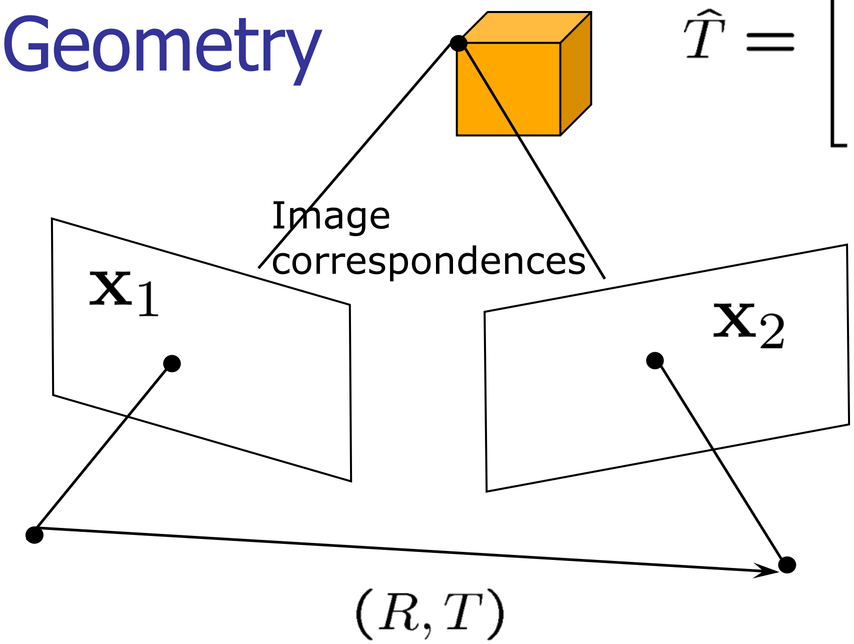
Find such Rotation and Translation and Depth that the reprojection error is minimized

Two views ~ 200 points
6 unknowns – Motion 3 Rotation, 3 Translation
- Structure 200x3 coordinates
- (-) universal scale

Difficult optimization problem

Epipolar Geometry

$$\hat{T} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$$



$$\lambda_2 \mathbf{x}_2 = R\lambda_1 \mathbf{x}_1 + T \quad / \widehat{\mathbf{x}_2} T$$

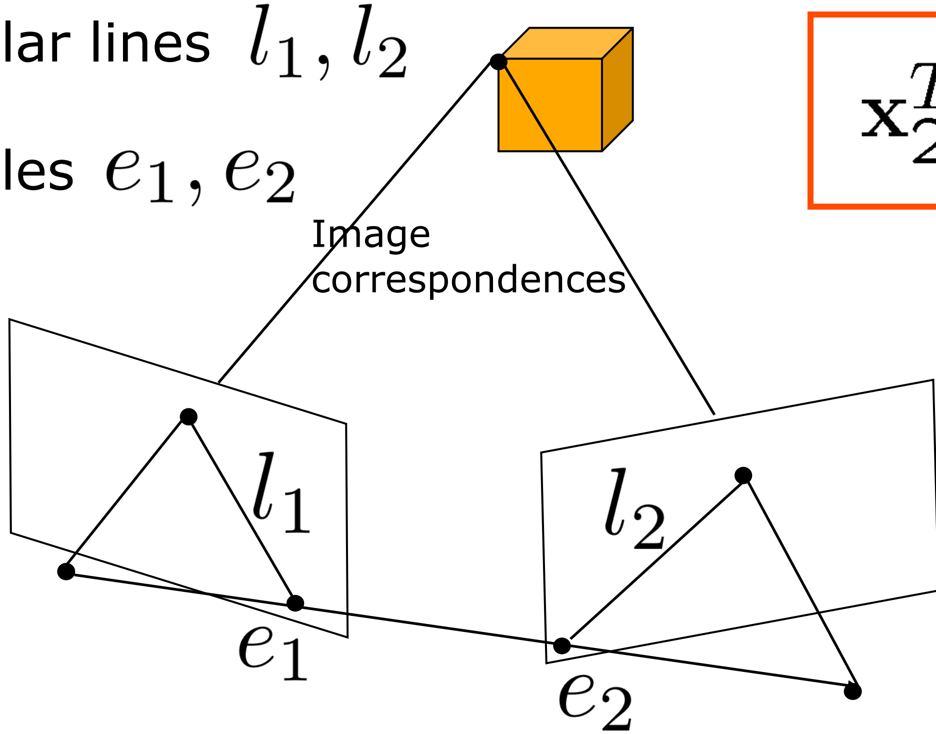
- Algebraic Elimination of Depth [Longuet-Higgins ’81]:

$$\mathbf{x}_2^T \underbrace{\hat{T} R}_{E} \mathbf{x}_1 = 0$$

- Essential matrix $E = \hat{T} R$

Epipolar Geometry

- Epipolar lines l_1, l_2
- Epipoles e_1, e_2



$$\boxed{\mathbf{x}_2^T E \mathbf{x}_1 = 0}$$

$$E = \hat{T} R$$

- Additional constraints

Epipolar transfer

$$l_1 \sim E^T \mathbf{x}_2 \qquad l_i^T \mathbf{x}_i = 0 \qquad l_2 \sim E \mathbf{x}_1$$

$$E \mathbf{e}_1 = 0 \qquad l_i^T \mathbf{e}_i = 0 \qquad \mathbf{e}_2 E^T = 0$$

Characterization of Essential Matrix

$$\mathbf{x}_2^T \widehat{T} R \mathbf{x}_1 = 0$$

Essential matrix $E = \widehat{T}R$ special 3x3 matrix

$$\mathbf{x}_2^T \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \mathbf{x}_1 = 0$$

(Essential Matrix Characterization)
A non-zero matrix E is an essential matrix iff its SVD: $E = U\Sigma V^T$
satisfies: $\Sigma = diag([\sigma_1, \sigma_2, \sigma_3])$ with $\sigma_1 = \sigma_2 \neq 0$ and $\sigma_3 = 0$
and $U, V \in SO(3)$

Estimating Essential Matrix

- Find such Rotation and Translation that the epipolar error is minimized

$$min_E \sum_{j=1}^{n} (\mathbf{x}_2^{jT} E \mathbf{x}_1^j)^2$$

- Space of all Essential Matrices is 5 dimensional
- 3 DOF Rotation, 2 DOF – Translation (up to scale !)
- Denote $\mathbf{a} = \mathbf{x}_1 \otimes \mathbf{x}_2$

$$\mathbf{a} = [x_1x_2, x_1y_2, x_1z_2, y_1x_2, y_1y_2, y_1z_2, z_1x_2, z_1y_2, z_1z_2]^T$$

$$E^s = [e_1, e_4, e_7, e_2, e_5, e_8, e_3, e_6, e_9]^T$$

- Rewrite

$$\mathbf{a}^T E^s = 0$$

- Collect constraints from all points

$$\chi E^s = 0$$

$$\min_E \sum_{i=1}^{n} (x_2^{j^T} E x_1^j)^2 \Rightarrow min_{E^s} \|\chi E^s\|^2$$

Estimating Essential Matrix

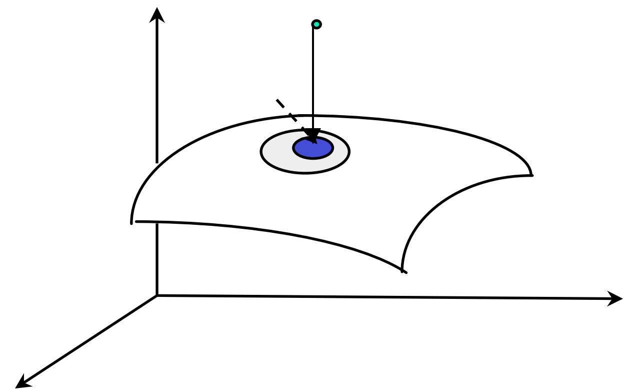
$$min_E \sum_{j=1}^{n} \mathbf{x}_2^{jT} E \mathbf{x}_1^j \Rightarrow min_{E^s} \|\chi E^s\|^2$$

Solution is

- Eigenvector associated with the smallest eigenvalue of $\chi^T \chi$
- If $rank(\chi^T \chi) < 8$ degenerate configuration

E_s estimated using linear least squares

unstack $\vec{E}_s$ F



Projection on to Essential Space

(Project onto a space of Essential Matrices)

If the SVD of a matrix $F \in \mathcal{R}^{3\times 3}$ is given by $F = Udiag(\sigma_1, \sigma_2, \sigma_3)V^T$ then the essential matrix which minimizes the Frobenius distance $\|E - F\|_f^2$ is given by $E = Udiag(\sigma, \sigma, 0)V^T$ with $\sigma = \frac{\sigma_1 + \sigma_2}{2}$

Pose Recovery from Essential Matrix

Essential matrix $E = \widehat{T}R$

(Pose Recovery)
There are two relative poses (R, T) with $T \in \mathcal{R}^3$ and $R \in SO(3)$ corresponding to a non-zero matrix essential matrix.

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

$$\begin{aligned}(\widehat{T}_1, R_1) &= (UR_Z(+\tfrac{\pi}{2})\Sigma U^T, UR_Z^T(+\tfrac{\pi}{2})V^T)\\ (\widehat{T}_2, R_2) &= (UR_Z(-\tfrac{\pi}{2})\Sigma U^T, UR_Z^T(-\tfrac{\pi}{2})V^T)\end{aligned}$$

$$\Sigma = diag([1,1,0]) \qquad R_z(+\tfrac{\pi}{2}) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Twisted pair ambiguity $(R_2, T_2) = (e^{\widehat{u}\pi} R_1, -T_1)$

Pose Recovery

- There are two pairs (R,T) corresponding to essential matrix E .
- There are two pairs (R,T) corresponding to essential matrix $-E$.
- Positive depth constraint disambiguates the impossible solutions
- Translation has to be non-zero, can be recovered up to scale
- Points have to be in general position
  - degenerate configurations – planar points
  - quadratic surface
- Linear 8-point algorithm
- Nonlinear 5-point algorithms yields up to 10 solutions

3D Structure Recovery

$$\underline{\lambda_2}\mathbf{x}_2 = R\underline{\lambda_1}\mathbf{x}_1 + \underline{\gamma}T$$

unknowns

- Eliminate one of the scale's

$$\lambda_1^j \widehat{\mathbf{x}_2^j} R \mathbf{x}_1^j + \gamma \widehat{\mathbf{x}_2^j} T = 0, \quad j = 1, 2, \ldots, n$$

- Solve LLSE problem

$$M^j \bar{\lambda}^j \doteq \left[\widehat{\mathbf{x}_2^j} R \mathbf{x}_1^j, \ \widehat{\mathbf{x}_2^j} T \right] \begin{bmatrix} \lambda_1^j \\ \gamma \end{bmatrix} = 0$$

If the configuration is non-critical, the Euclidean structure of the points and motion of the camera can be reconstructed up to a universal scale.

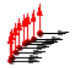
- Alternatively recover each point depth separately

Two views

Point Feature Matching

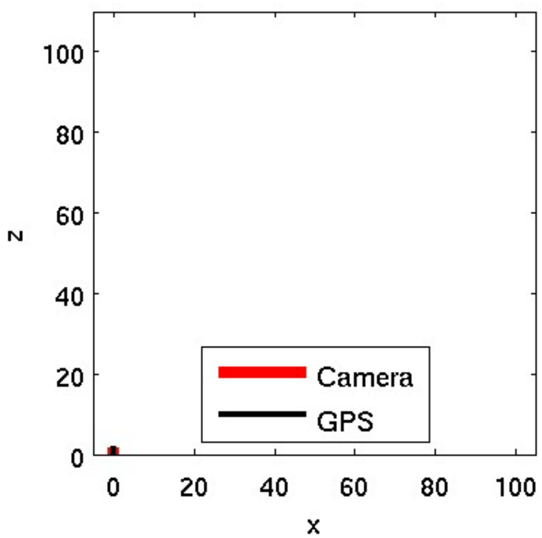
Epipolar Geometry

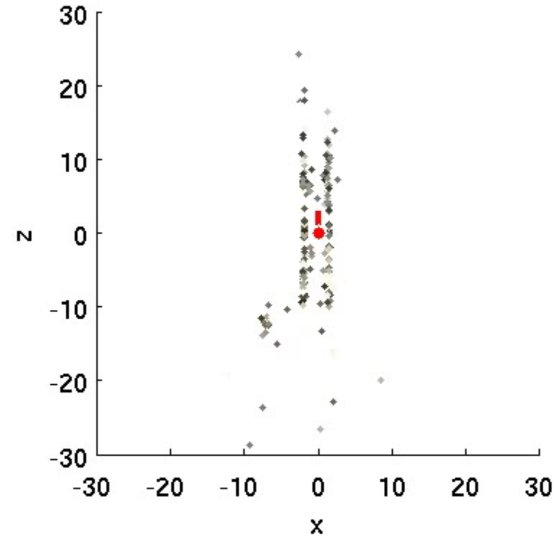
Camera Pose
and
 Sparse Structure Recovery



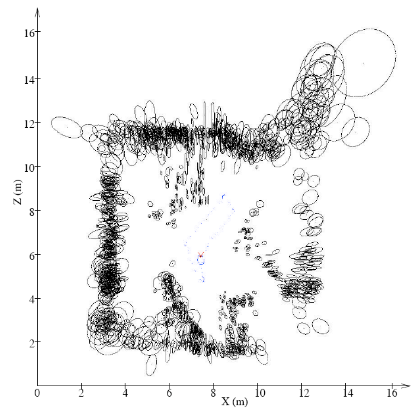
Visual Odometry

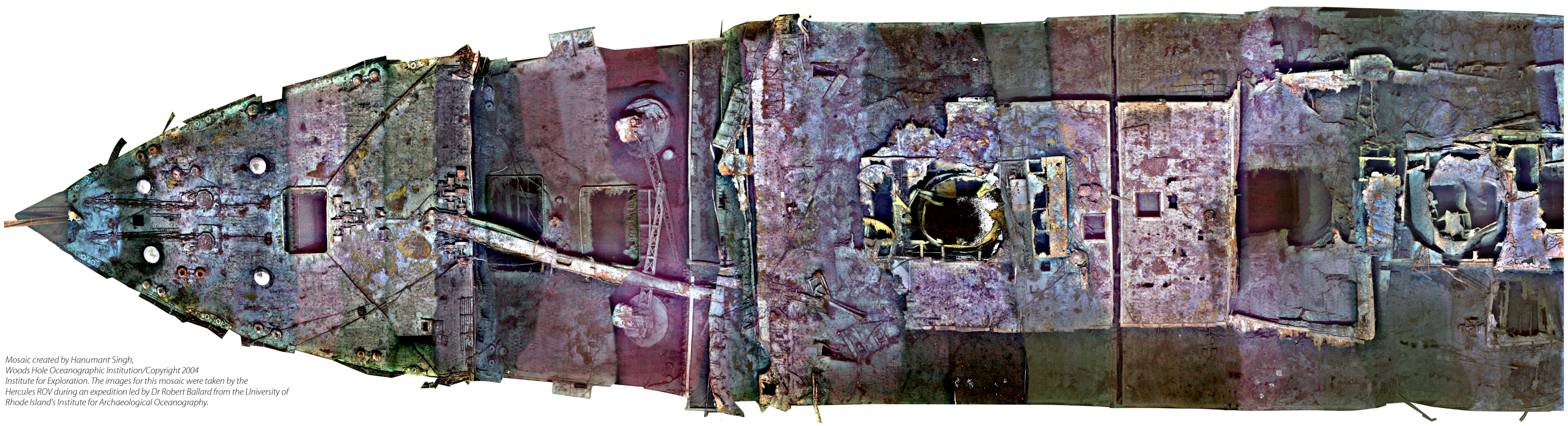
estimate motion from image correspondences





Mapping, Localization, Recognition





Mosaic created by Hanumant Singh,
Woods Hole Oceanographic Institution/Copyright 2004
Institute for Exploration. The images for this mosaic were taken by the
Hercules ROV during an expedition led by Dr Robert Ballard from the University of
Rhode Island's Institute for Archaeological Oceanography.

Two view motion estimation

- Key component of visual odometry
- When carried our over multiple frames – need for global adjustment
- Later in the class when we talk about mapping and localization
- Alternatives – motion estimation using moving stereo rig

Dealing with correspondences

- Previous methods assumed that we have exact correspondences
- Followed by linear least squares estimation
- Correspondences established either by tracking (using affine or translational flow models)
- Or wide-baseline matching (using scale/rotation invariant features and their descriptors)
- In many cases we get incorrect matches/tracks

Robust estimators for dealing with outliers

- Use robust objective function
- The M-estimator and Least Median of Squares (LMedS) Estimator (neither of them can tolerate more than 50% outliers)

- The RANSAC (RANdom SAmple Consensus) algorithm
- Proposed by Fischler and Bolles
- Popular technique used in Computer Vision community (and else where for robust estimation problems)

- It can tolerate more than 50% outliers

The RANSAC algorithm

- Generate *M* (a predetermined number) model hypotheses, each of them is computed using a minimal subset of points

- Evaluate each hypothesis

- Compute its residuals with respect to all data points.
- Points with residuals less than some threshold are classified as its inliers

- The hypothesis with the maximal number of inliers is chosen. Then re-estimate the model parameter using its identified inliers.

RANSAC – Practice

- The theoretical number of samples needed to ensure 95% confidence that at least one outlier free sample could be obtained.

$$\rho = 1 - (1 - (1 - \epsilon)^k)^s$$

- Probability that a point is an outlier $1 - \epsilon$
- Number of points per sample k
- Probability of at least one outlier free sample ρ

- Then number of samples needed to get an outlier free sample with probability ρ

$$s = \frac{\log(1 - \rho)}{\log(1 - (1 - \epsilon)^k)}$$

RANSAC – Practice

- The theoretical number of samples needed to ensure 95% confidence that at least one outlier free sample could be obtained.
- Example for estimation of essential/fundamental matrix
- Need at least 7 or 8 points in one sample i.e. k = 7, probability is
- 0.95 then the number if samples for different outlier ratio ϵ

Outlier ratio	20%	30%	40%	50%	60%	70%
seven-point algorithm	13	35	106	382	1827	13696
eight-point algorithm	17	51	177	766	4570	45658

- In practice we do not now the outlier ratio
- Solution adaptively adjust number of samples as you go along
- While estimating the outlier ratio

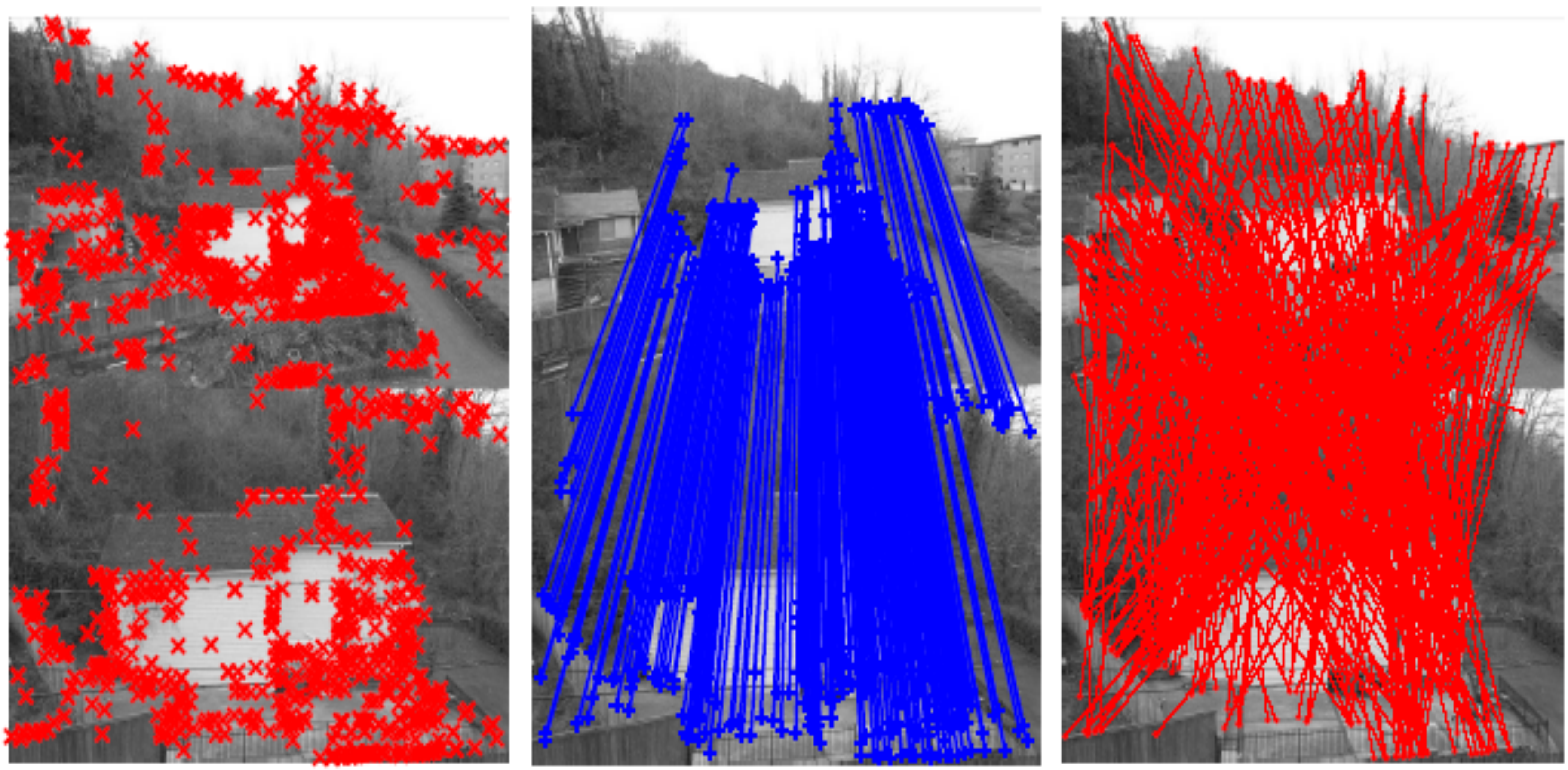
The difficulty in applying RANSAC

- Drawbacks of the standard RANSAC algorithm
  - Requires a large number of samples for data with many outliers (exactly the data that we are dealing with)
  - Needs to know the outlier ratio to estimate the number of samples
  - Requires a threshold for determining whether points are inliers
- Various improvements to standard approaches
  [Torr'99, Murray'02, Nister'04, Matas'05, Sutter'05 and many others

Adaptive RANSAC

- s = infinity, sample_count = 0;
- While s > sample_count repeat
  - choose a sample and count the number of inliers
  - set $\epsilon = 1-$ (number_of_inliers/total_number_of_points)
  - set s from ϵ and $\rho = 0.99$
  - increment sample_count by 1
- terminate

Robust technique



(a) correspondences. (b) identified inliers. (c) identified outliers.

Robust matching

- Select set of putative correspondences $\mathbf{x}_1^j, \mathbf{x}_2^j$

$$\mathbf{x}_2^T F \mathbf{x}_1 = 0$$

- Repeat

  1. Select at random a set of 8 successful matches

  2. Compute fundamental matrix

  3. Determine the subset of inliers, compute distance to epipolar line

$$d^{2}_{j} \doteq \frac{(\mathbf{x}_2^{jT} F_k \mathbf{x}_1^j)^2}{\|\widehat{e}_3 F \mathbf{x}_1^j\|^2 + \|\mathbf{x}_2^{jT} F \widehat{e}_3\|^2} \qquad d_j \leq \tau_d$$

  4. Count the number of points in the consensus set