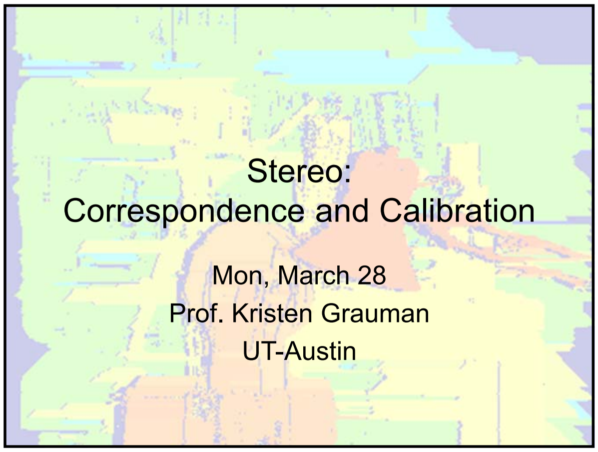


Stereo: Correspondence and Calibration

Mon, March 28
Prof. Kristen Grauman
UT-Austin

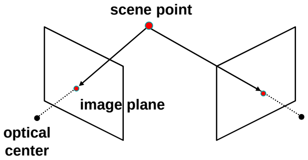
Today

- Recap: epipolar constraint
- Stereo image rectification
- Stereo solutions
  - Computing correspondences
  - Non-geometric stereo constraints
- Calibration

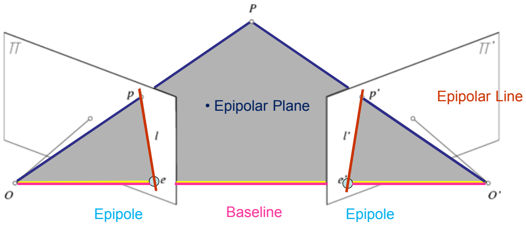
- Example stereo applications

Last time: Estimating depth with stereo

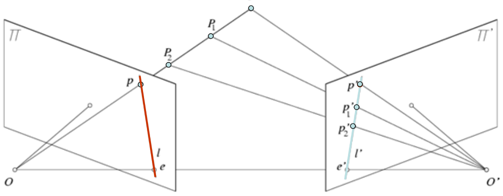
- **Stereo**: shape from "motion" between two views
- We need to consider:
  - Info on camera pose ("calibration")
  - Image point correspondences



Last time: Epipolar geometry



Last time: Epipolar constraint



- Potential matches for *p* have to lie on the corresponding epipolar line *l'*.
- Potential matches for *p'* have to lie on the corresponding epipolar line *l*.

Slide credit: M. Pollefeys

Example: converging cameras

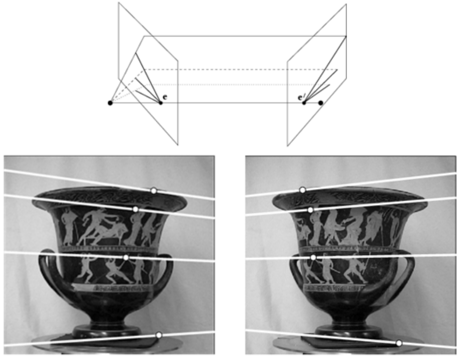


Figure from Hartley & Zisserman

Example: parallel cameras

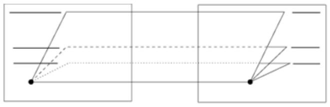
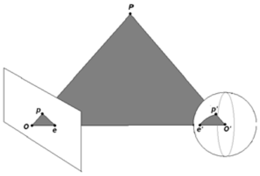


Figure from Hartley & Zisserman

An audio camera & epipolar geometry



Spherical microphone array

Adam O' Donovan, Ramani Duraiswami and Jan Neumann
Microphone Arrays as Generalized Cameras for Integrated Audio Visual Processing, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, 2007

An audio camera & epipolar geometry

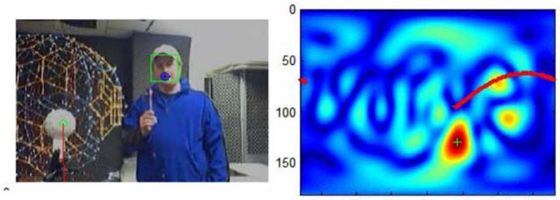


Figure 4. An example of the use of the system in speaker tracking with noise suppression. The bright red spot on the sound image (marked with a +) corresponds to the dominant source. The less dominant source however lies on the epipolar line in the sound image induced by the location of the mouth in the camera image, and this source is beamformed.

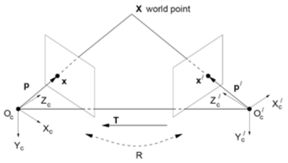
Last time: Essential matrix

$$\mathbf{X}'\cdot\left(\mathbf{T}\times\mathbf{R}\mathbf{X}\right)=0$$

$$\mathbf{X}'\cdot\left([\mathrm{T}_x]\mathbf{R}\mathbf{X}\right)=0$$

Let $\mathbf{E}=[\mathrm{T}_x]\mathbf{R}$

$$\mathbf{X}'^T\mathbf{E}\mathbf{X}=0$$

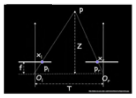


E is called the **essential matrix**, and it relates corresponding points between both cameras, given the rotation and translation.

If we observe a point in one image, its position in other image is constrained to lie on line defined by above.

Note: these points are in **camera coordinate systems**.

Essential matrix example: parallel cameras



$\mathbf{R}=$

$\mathbf{T}=$

$\mathbf{E}=[\mathrm{T}_x]\mathbf{R}=$

$\mathbf{p}=[x,y,f]$

$\mathbf{p}'=[x',y',f]$

$$\mathbf{p}'^{\mathrm{T}}\mathbf{E}\mathbf{p}=0$$

For the parallel cameras, image of any point must lie on same horizontal line in each image plane.

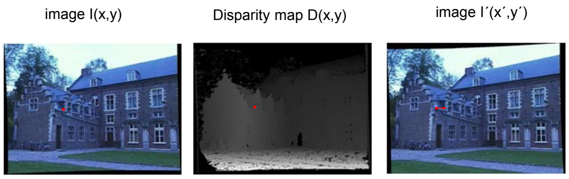


image I(x,y)

Disparity map D(x,y)

image I´(x´,y´)

(x´,y´)=(x+D(x,y),y)

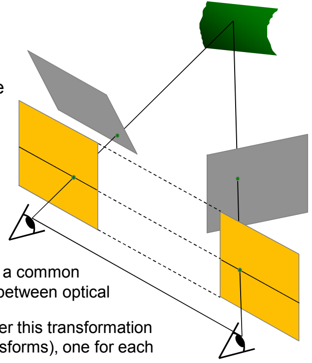
What about when cameras' optical axes are not parallel?

Today

- Recap: epipolar constraint
- Stereo image rectification
- Stereo solutions
  - Computing correspondences
  - Non-geometric stereo constraints
- Calibration
- Example stereo applications

Stereo image rectification

In practice, it is convenient if image scanlines (rows) are the epipolar lines.



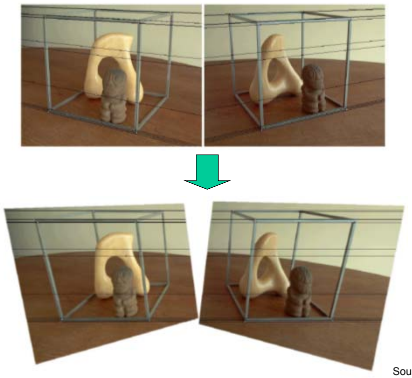
reproject image planes onto a common plane parallel to the line between optical centers

pixel motion is horizontal after this transformation

two homographies (3x3 transforms), one for each input image reprojection

Slide credit: Li Zhang

Stereo image rectification: example

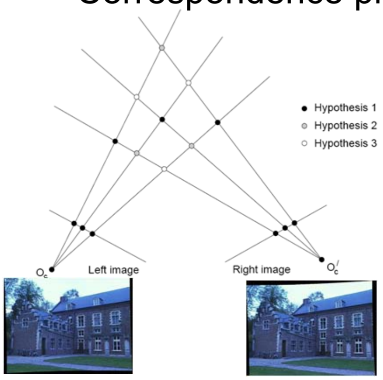


Source: Alyosha Efros

Today

- Recap: epipolar constraint
- Stereo image rectification
- Stereo solutions
  - Computing correspondences
  - Non-geometric stereo constraints
- Calibration
- Example stereo applications

Correspondence problem

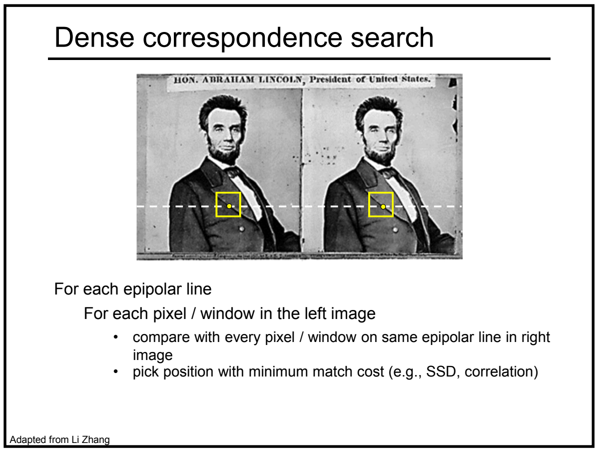


Multiple match hypotheses satisfy epipolar constraint, but which is correct?

Figure from Gee & Cipolla 1999

Correspondence problem

- Beyond the hard constraint of epipolar geometry, there are “soft” constraints to help identify corresponding points
  - Similarity
  - Uniqueness
  - Ordering
  - Disparity gradient
- To find matches in the image pair, we will assume
  - Most scene points visible from both views
  - Image regions for the matches are similar in appearance



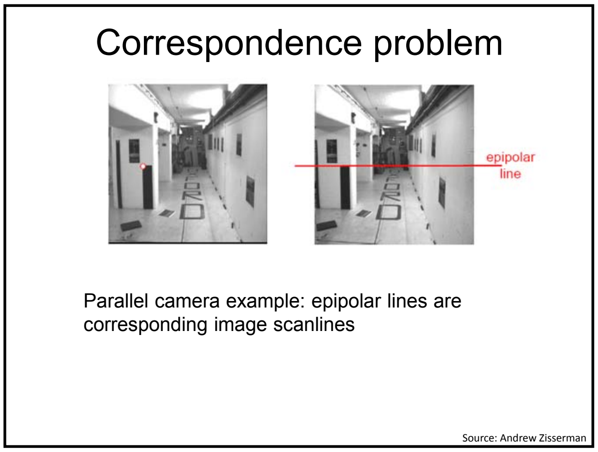
Dense correspondence search

For each epipolar line

For each pixel / window in the left image

- compare with every pixel / window on same epipolar line in right image
- pick position with minimum match cost (e.g., SSD, correlation)

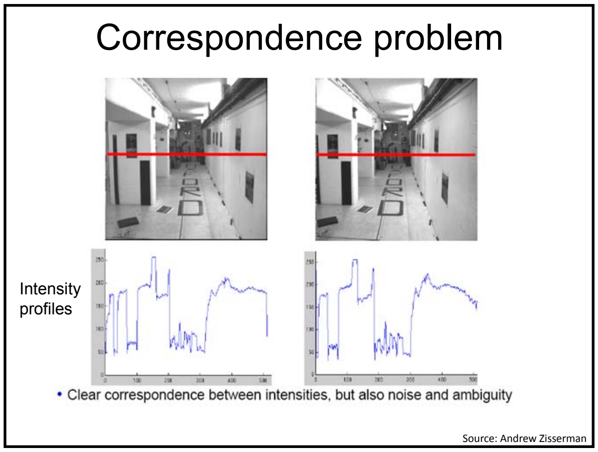
Adapted from Li Zhang



Correspondence problem

Parallel camera example: epipolar lines are corresponding image scanlines

Source: Andrew Zisserman

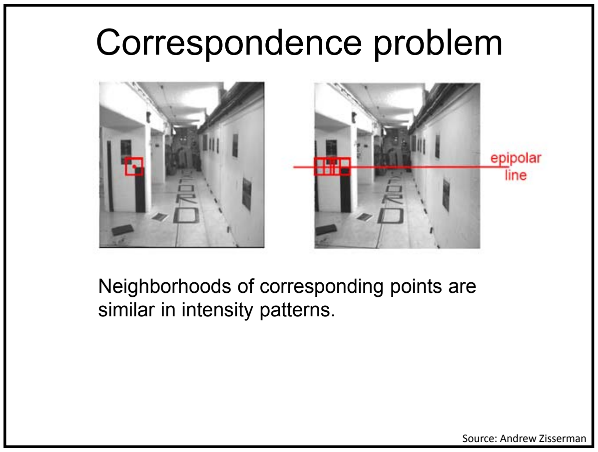


Correspondence problem

Intensity profiles

- Clear correspondence between intensities, but also noise and ambiguity

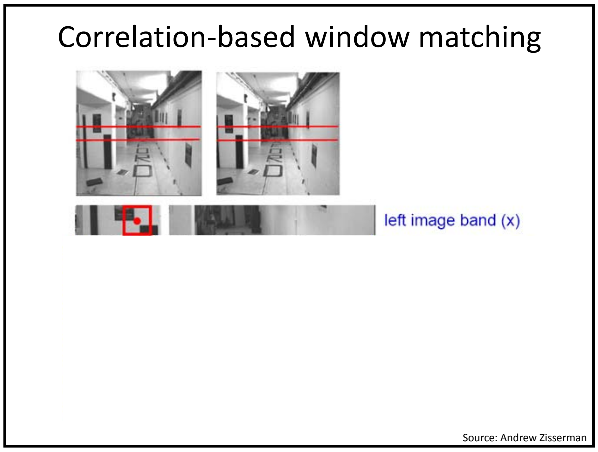
Source: Andrew Zisserman



Correspondence problem

Neighborhoods of corresponding points are similar in intensity patterns.

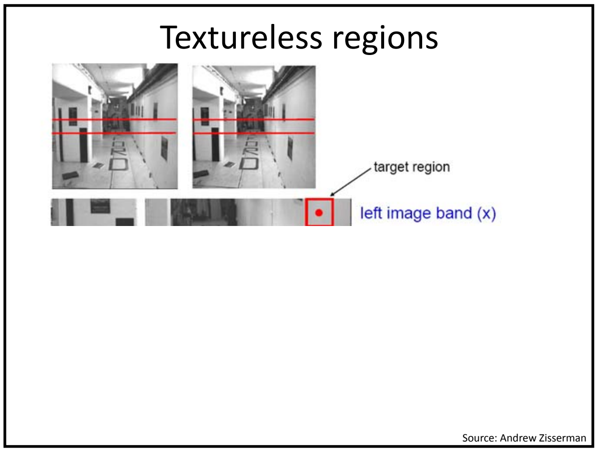
Source: Andrew Zisserman



Correlation-based window matching

left image band (x)

Source: Andrew Zisserman



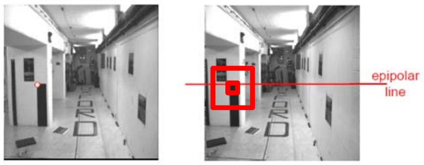
Textureless regions

target region

left image band (x)

Source: Andrew Zisserman

Effect of window size



Source: Andrew Zisserman

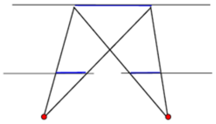
Effect of window size

W = 3 W = 20

Want window large enough to have sufficient intensity variation, yet small enough to contain only pixels with about the same disparity.

Figures from Li Zhang

Foreshortening effects

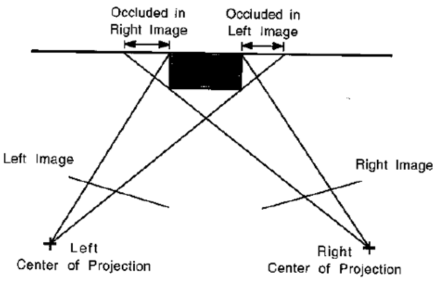


fronto-parallel surface

imaged length the same

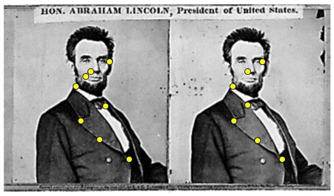
Source: Andrew Zisserman

Occlusion



Slide credit: David Kriegman

Sparse correspondence search



- Restrict search to sparse set of **detected features** (e.g., corners)
- Rather than pixel values (or lists of pixel values) use *feature descriptor* and an associated *feature distance*
- Still narrow search further by epipolar geometry

Tradeoffs between dense and sparse search?

Correspondence problem

- Beyond the hard constraint of epipolar geometry, there are “soft” constraints to help identify corresponding points
  - Similarity
  - Uniqueness
  - Disparity gradient
  - Ordering

Uniqueness constraint

- Up to one match in right image for every point in left image

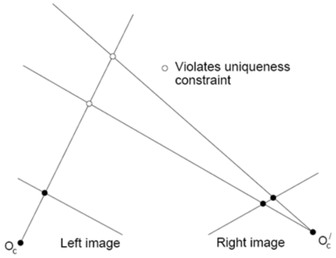
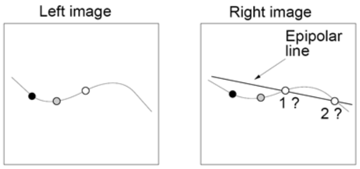


Figure from Gee & Cipolla 1999

Disparity gradient constraint

- Assume piecewise continuous surface, so want disparity estimates to be locally smooth



Given matches ● and ◎, point ○ in the left image must match point 1 in the right image. Point 2 would exceed the disparity gradient limit.

Figure from Gee & Cipolla 1999

Ordering constraint

- Points on **same surface** (opaque object) will be in same order in both views

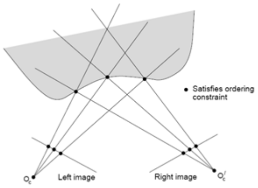
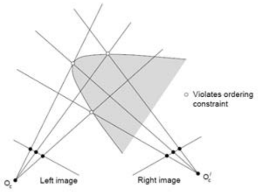
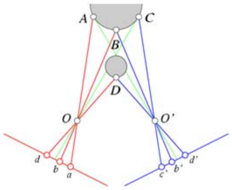


Figure from Gee & Cipolla 1999

Ordering constraint

- Won't always hold, e.g. consider transparent object, or an occluding surface



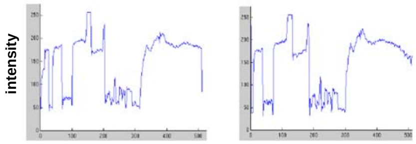


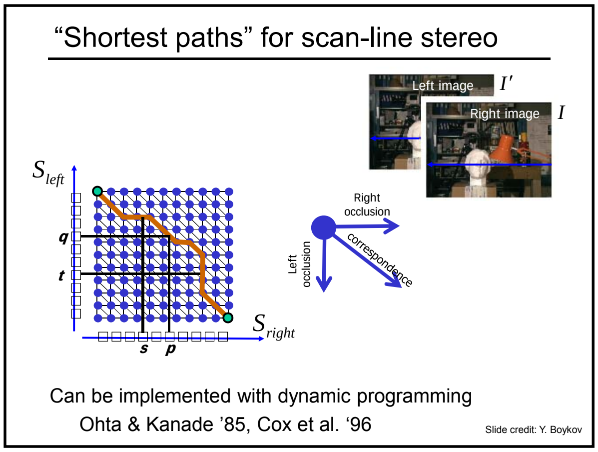
Figures from Forsyth & Ponce

- Beyond individual correspondences to estimate disparities:
- Optimize correspondence assignments jointly
  - Scanline at a time (DP)
  - Full 2D grid (graph cuts)

Scanline stereo

- Try to coherently match pixels on the entire scanline
- Different scanlines are still optimized independently



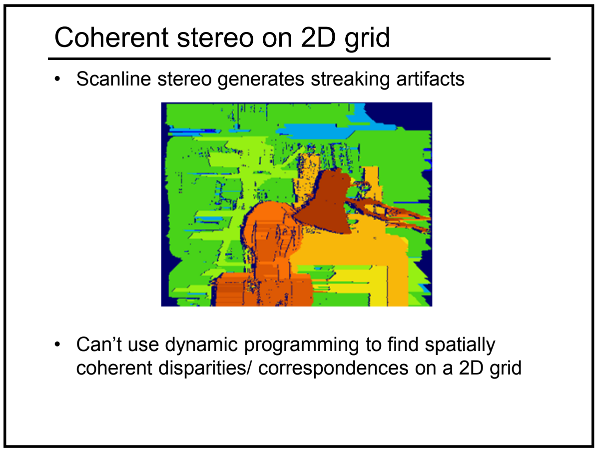


"Shortest paths" for scan-line stereo

Can be implemented with dynamic programming

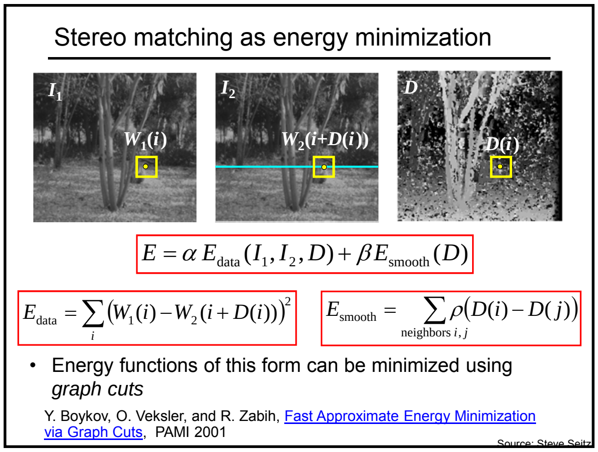
Ohta & Kanade '85, Cox et al. '96

Slide credit: Y. Boykov



Coherent stereo on 2D grid

- Scanline stereo generates streaking artifacts
- Can't use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid



Stereo matching as energy minimization

$$E = \alpha E_{\text{data}}(I_1, I_2, D) + \beta E_{\text{smooth}}(D)$$

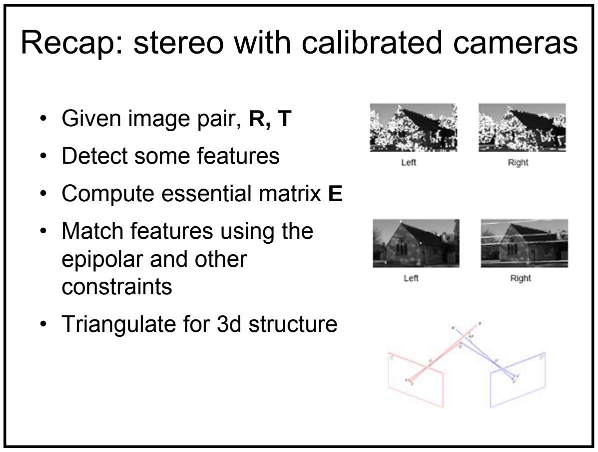
$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2$$

$$E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \rho\left(D(i) - D(j)\right)$$

- Energy functions of this form can be minimized using *graph cuts*

Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Source: Steve Seitz



Recap: stereo with calibrated cameras

- Given image pair, **R, T**
- Detect some features
- Compute essential matrix **E**
- Match features using the epipolar and other constraints
- Triangulate for 3d structure

Error sources

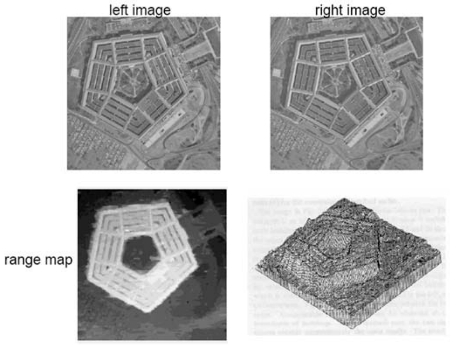
- Low-contrast ; textureless image regions
- Occlusions
- Camera calibration errors
- Violations of *brightness constancy* (e.g., specular reflections)
- Large motions

Today

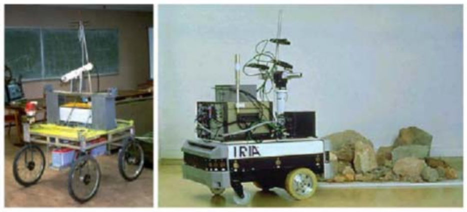
- – Recap: epipolar constraint
- – Stereo image rectification
- – Stereo solutions
  - Computing correspondences
  - Non-geometric stereo constraints
- – Calibration

- – Example stereo applications

-



Stereo in machine vision systems

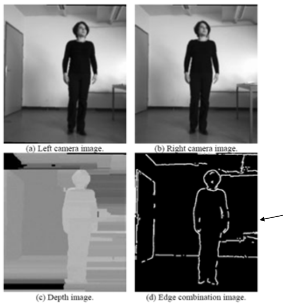


Left : The Stanford cart sports a single camera moving in discrete increments along a straight line and providing multiple snapshots of outdoor scenes

Right : The INRIA mobile robot uses three cameras to map its environment

Forsyth & Ponce

Depth for segmentation



(a) Left camera image. (b) Right camera image.

(c) Depth image. (d) Edge combination image.

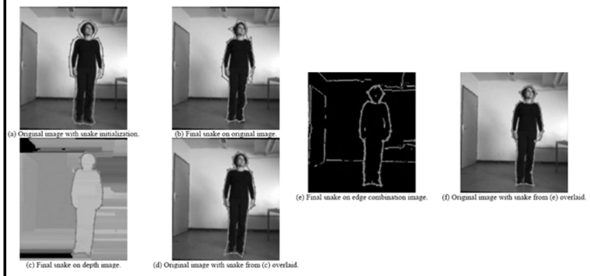
Figure 3 Stereo video frames with computed depth map and edge combination result.

Edges in disparity in conjunction with image edges enhances contours found

Danijela Markovic and Margrit Gelautz, Interactive Media Systems Group, Vienna University of Technology

---

Depth for segmentation



(a) Original image with snake initialization. (b) Final snake on original image.

(c) Final snake on depth image. (d) Original image with snake from (c) overlaid.

(e) Final snake on edge combination image. (f) Original image with snake from (e) overlaid.

Danijela Markovic and Margrit Gelautz, Interactive Media Systems Group, Vienna University of Technology

---

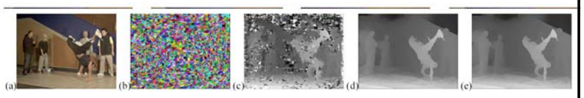
Model-based body tracking, stereo input

David Demirdjian, MIT Vision Interface Group

http://people.csail.mit.edu/demirdji/movie/artic-tracker/turn-around.m1v

---

Virtual viewpoint video



(a) (b) (c) (d) (e)

Figure 6: *Sample results from stereo reconstruction stage: (a) input color image; (b) color-based segmentation; (c) initial disparity estimates $\hat{d}_{ij}$; (d) refined disparity estimates; (e) smoothed disparity estimates $d_i(x)$.*

d) A depth-matted object from earlier in the sequence is inserted into the video.

C. Zitnick et al, High-quality video view interpolation using a layered representation, SIGGRAPH 2004.

Virtual viewpoint video

http://research.microsoft.com/IVM/VVV/

Uncalibrated case

- What if we don't know the camera parameters?

Two possibilities:

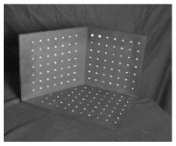
1. Calibrate with a calibration object
2. Weak calibration

Calibrating a camera

- Compute intrinsic and extrinsic parameters using observed camera data

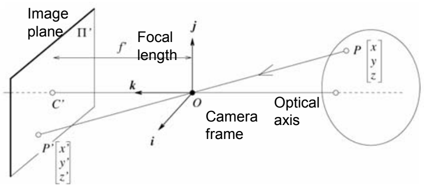
Main idea

- Place "calibration object" with known geometry in the scene
- Get correspondences
- Solve for mapping from scene to image



The Opti-CAL Calibration Target Image

Perspective projection



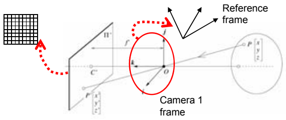
$$(x,y,z) \rightarrow (f'\frac{x}{z}, f'\frac{y}{z})$$

Scene point → Image coordinates

Thus far, in **camera's** reference frame only.

Camera parameters

- **Extrinsic: location and orientation of camera frame with respect to reference frame**
- Intrinsic: how to map pixel coordinates to image plane coordinates



Extrinsic camera parameters

$$\mathbf{P}_c = \mathbf{R}(\mathbf{P}_w - \mathbf{T})$$

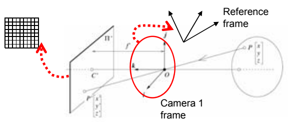
Camera reference frame

World reference frame

$$\mathbf{P}_c = (X, Y, Z)^T$$

Camera parameters

- Extrinsic: location and orientation of camera frame with respect to reference frame
- **Intrinsic: how to map pixel coordinates to image plane coordinates**



Intrinsic camera parameters

- Ignoring any geometric distortions from optics, we can describe them by:

$$x = -(x_{im} - o_x)s_x$$

$$y = -(y_{im} - o_y)s_y$$

Coordinates of projected point in camera reference frame

Coordinates of image point in pixel units

Coordinates of image center in pixel units

Effective size of a pixel (mm)

Camera parameters

- We know that in terms of camera reference frame:

$$x = f\frac{X}{Z} \qquad y = f\frac{Y}{Z} \qquad \text{and} \qquad \mathbf{P}_c = \mathbf{R}(\mathbf{P}_w - \mathbf{T}) \quad \mathbf{P}_c = (X, Y, Z)^T$$

- Substituting previous eqns describing intrinsic and extrinsic parameters, can relate *pixels coordinates* to *world points:*

$$-(x_{im} - o_x)s_x = f\frac{\mathbf{R}_1 \cdot (\mathbf{P}_w - \mathbf{T})}{\mathbf{R}_3 \cdot (\mathbf{P}_w - \mathbf{T})}$$

$$-(y_{im} - o_y)s_y = f\frac{\mathbf{R}_2 \cdot (\mathbf{P}_w - \mathbf{T})}{\mathbf{R}_3 \cdot (\mathbf{P}_w - \mathbf{T})}$$

$\mathbf{R}_i$ = Row i of rotation matrix

Projection matrix

- This can be rewritten as a matrix product using homogeneous coordinates:

$$\begin{bmatrix} wx_{im} \\ wy_{im} \\ w \end{bmatrix} = \mathbf{M}P_w$$

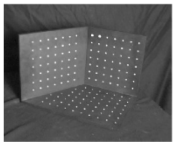
where:

$$\mathbf{M}_{\text{int}} = \begin{bmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & -\mathbf{R}_1{}^T\mathbf{T} \\ r_{21} & r_{22} & r_{23} & -\mathbf{R}_2{}^T\mathbf{T} \\ r_{31} & r_{32} & r_{33} & -\mathbf{R}_3{}^T\mathbf{T} \end{bmatrix}$$

Calibrating a camera

- Compute intrinsic and extrinsic parameters using observed camera data



Main idea

- Place “calibration object” with known geometry in the scene
- Get correspondences
- Solve for mapping from scene to image: estimate $\mathbf{M}=\mathbf{M}_{\text{int}}\mathbf{M}_{\text{ext}}$

The Opti-CAL Calibration Target Image

When would we calibrate this way?

- Makes sense when geometry of system is not going to change over time

…when would it change?

Weak calibration

- Want to estimate world geometry without requiring calibrated cameras
  - Archival videos
  - Photos from multiple unrelated users
  - Dynamic camera system
- **Main idea**:
  - Estimate epipolar geometry from a (redundant) set of point correspondences between two uncalibrated cameras

From before: Projection matrix

- This can be rewritten as a matrix product using homogeneous coordinates:

$$\begin{bmatrix} wx_{im} \\ wy_{im} \\ w \end{bmatrix} = \mathbf{M}_{\text{int}} \mathbf{M}_{ext} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

where:

$$\mathbf{M}_{\text{int}} = \begin{bmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & -\mathbf{R}_1{}^{\mathrm{T}}\mathbf{T} \\ r_{21} & r_{22} & r_{23} & -\mathbf{R}_2{}^{\mathrm{T}}\mathbf{T} \\ r_{31} & r_{32} & r_{33} & -\mathbf{R}_3{}^{\mathrm{T}}\mathbf{T} \end{bmatrix}$$

From before: Projection matrix

- This can be rewritten as a matrix product using homogeneous coordinates:

$$\begin{bmatrix} wx_{im} \\ wy_{im} \\ w \end{bmatrix} = \mathbf{M}_{int} \mathbf{M}_{ext} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

$$\mathbf{p}_{im} = \mathbf{M}_{int} \underbrace{\mathbf{M}_{ext} \mathbf{P}_w}_{\mathbf{p}_c}$$

$$\mathbf{p}_{im} = \mathbf{M}_{int} \mathbf{p}_c$$

Uncalibrated case

For a given camera: $\mathbf{p}_{im} = \mathbf{M}_{int} \mathbf{p}_c$

So, for **two** cameras (left and right):

$$\mathbf{p}_{c,left} = \mathbf{M}^{-1}_{int,left} \mathbf{p}_{im,left}$$

$$\mathbf{p}_{c,right} = \underbrace{\mathbf{M}^{-1}_{int,right}}_{\text{Internal calibration matrices, one per camera}} \mathbf{p}_{im,right}$$

Uncalibrated case

$$\mathbf{p}_{c,left} = \mathbf{M}^{-1}_{int,left} \mathbf{p}_{im,left}$$

$$\mathbf{p}_{c,right} = \mathbf{M}^{-1}_{int,right} \mathbf{p}_{im,right}$$

$$\boxed{\mathbf{p}_{c,right}{}^{\mathrm{T}} \mathbf{E} \mathbf{p}_{c,left} = 0}$$

From before, the **essential** matrix **E.**

$$\left(\mathbf{M}^{-1}_{int,right} \mathbf{p}_{im,right}\right)^{\mathrm{T}} \mathbf{E} \left(\mathbf{M}^{-1}_{int,left} \mathbf{p}_{im,left}\right) = 0$$

$$\mathbf{p}^{\mathrm{T}}_{im,right} \underbrace{\left(\mathbf{M}^{-\mathrm{T}}_{int,right} \mathbf{E} \mathbf{M}^{-1}_{int,left}\right)}_{\mathbf{F}} \mathbf{p}_{im,left} = 0$$

“Fundamental matrix”

$$\boxed{\mathbf{p}^{\mathrm{T}}_{im,right} \mathbf{F} \mathbf{p}_{im,left} = 0}$$

Computing F from correspondences

Each point correspondence generates one constraint on F

$$\mathbf{p}^{\mathrm{T}}_{im,right} \mathbf{F} \mathbf{p}_{im,left} = 0$$

$$\begin{bmatrix} u' & v' & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0$$

Collect n of these constraints

$$\begin{bmatrix} u'_1u_1 & u'_1v_1 & u'_1 & v'_1u_1 & v'_1v_1 & v'_1 & u_1 & v_1 & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = \mathbf{0}$$

Solve for f , vector of parameters.

Fundamental matrix

- Relates **pixel coordinates** in the two views
- More general form than essential matrix: we remove need to know intrinsic parameters
- If we estimate fundamental matrix from correspondences in *pixel coordinates*, can reconstruct epipolar geometry without intrinsic or extrinsic parameters.

Stereo pipeline with weak calibration

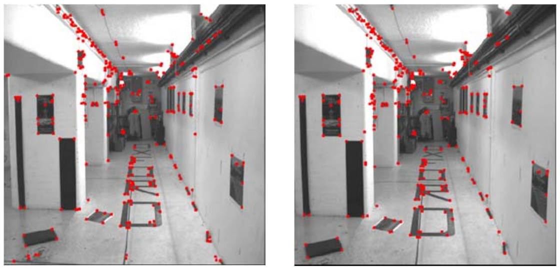
- So, where to start with uncalibrated cameras?
  - Need to find fundamental matrix F **and** the correspondences (pairs of points (u',v') ↔ (u,v)).

- 1) Find interest points in image
- 2) Compute correspondences
- 3) Compute epipolar geometry
- 4) Refine

Example from Andrew Zisserman

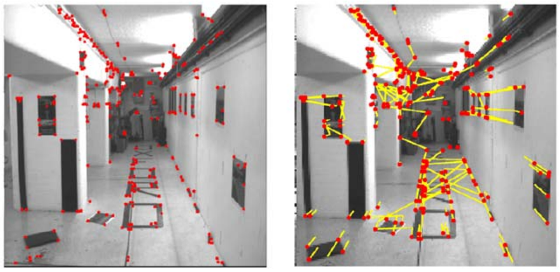
Stereo pipeline with weak calibration

1) Find interest points



Stereo pipeline with weak calibration

2) Match points within proximity to get putative matches



Stereo pipeline with weak calibration

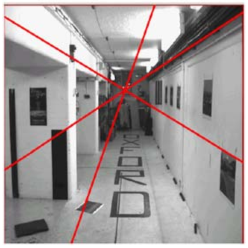
3) Compute epipolar geometry -- robustly with RANSAC

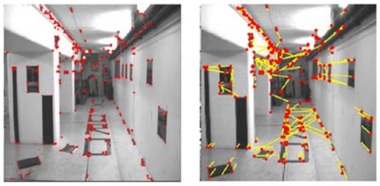
Select random sample of putative correspondences

Compute **F** using them
- determines epipolar constraint

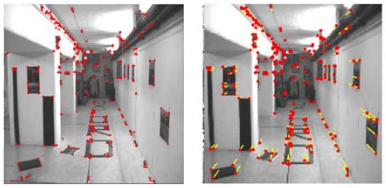
Evaluate amount of support
- inliers within threshold distance of epipolar line

Choose F with most support (inliers)





Using correlation search to get putative matches: noisy, but enough to compute F using RANSAC



Pruned matches: those consistent with epipolar geometry

Summary

- **Rectification**: make epipolar lines align with scanlines
- Stereo solutions:
  - **Correspondence**: dense, or at interest points
  - **Non-geometric stereo constraints** (e.g., similarity, order, smoothness)
- Calibration
  - **With calibration object** in scene: relate world coordinates to image coordinates
  - **Weak calibration**: solve for fundamental matrix, relate image coordinates to image coordinates