Computer Vision, Lectures 11, 12, 13
Computational Stereo

Professor Hager
http://www.cs.jhu.edu/~hager
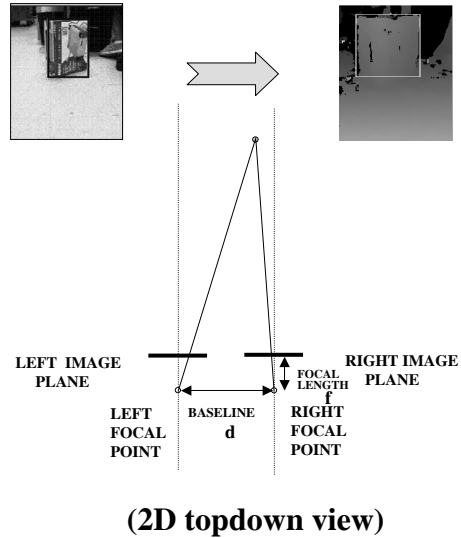
Computational Stereo

- Much of geometric vision is based on information from 2 (or more) camera locations
  - hard to recover 3D information from 2D images without extra knowledge
  - motion and stereo are both common in the world

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

- Stereo involves the following *two problems*
  - matching (correspondence problem)
    - rectification (we'll postpone a bit)
  - reconstruction (reconstruction problem)
    - calibration (already seen)

1

BINOCULAR STEREO SYSTEM: GEOMETRY

- **GOAL:** Passive 2-camera system for triangulating 3D position of points in space to generate a depth map of a world scene.

- **Example of a 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!

LEFT IMAGE PLANE

RIGHT IMAGE PLANE

FOCAL LENGTH
f

LEFT FOCAL POINT

BASELINE d

RIGHT FOCAL POINT

(2D topdown view)

11/5/2003   CS 461, Copyright G.D. Hager

BINOCULAR STEREO SYSTEM

DISPARITY
$(X_L - X_R)$

$Z = (f/X_L) X$
$Z = (f/X_R) (X-d)$
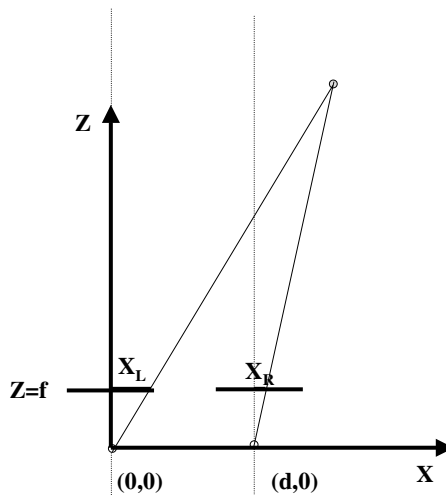
$(f/X_L) X = (f/X_R) (X-d)$
$X = (X_L d) / (X_L - X_R)$

$$X = \frac{d X_L}{(X_L - X_R)}$$

$$Z = \frac{d f}{(X_L - X_R)}$$

Z

X_L

X_R

Z=f

(0,0)

(d,0)

X

Unifocal camera (f=1) $\Rightarrow$ independent of the physical camera
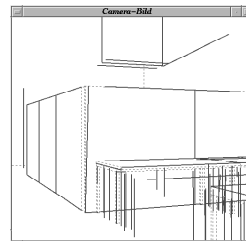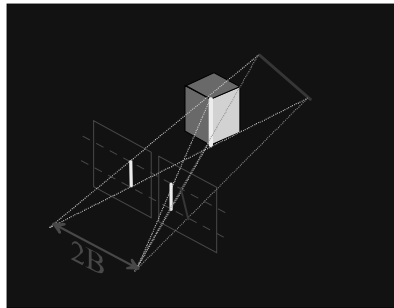
11/5/2003   CS 461, Copyright G.D. Hager

Stereo-Based Reconstruction

Correspondence Problem:

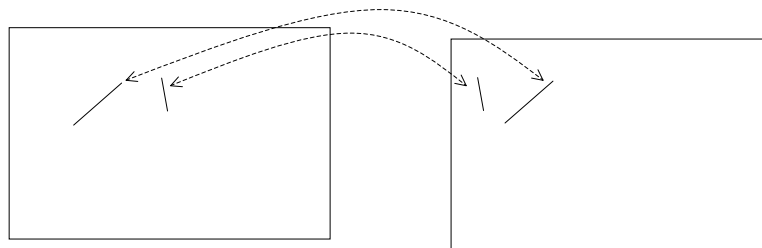How to find corresponding features in two camera images (points, line segments, curves, etc.)

MATCHING AND CORRESPONDENCE

- **Two major approaches**
  - feature-based
  - region based

In feature-based matching, the idea is to pick a feature type (e.g. edges), define a matching criteria (e.g. orientation and contrast sign), and then look for matches within a disparity range
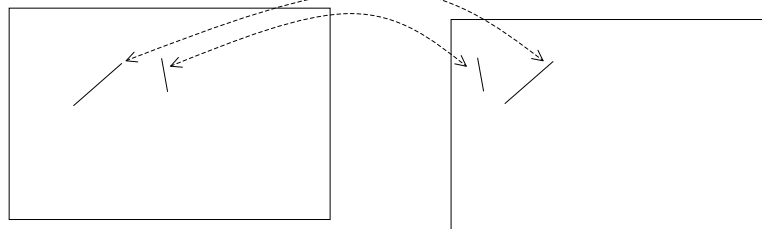
MATCHING AND CORRESPONDENCE

- **Two major approaches**
  - feature-based
  - region based

In feature-based matching, the idea is to pick a feature type (e.g. edges), define a matching criteria (e.g. orientation and contrast sign), and then look for matches within a disparity range
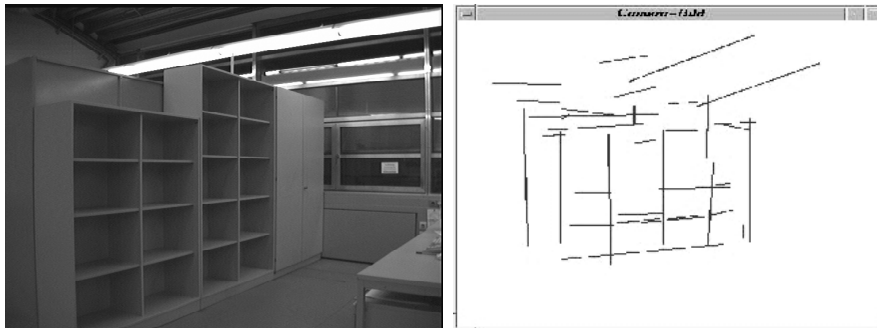
$$S = \frac{1}{w_0(l_l - l_r)^2 + w_1(m_l - m_r)^2 + w_2(o_l - o_r)^2 + w_3(c_l - c_r)^2}$$

11/5/2003 CS 461, Copyright G.D. Hager

Results - Reconstruction

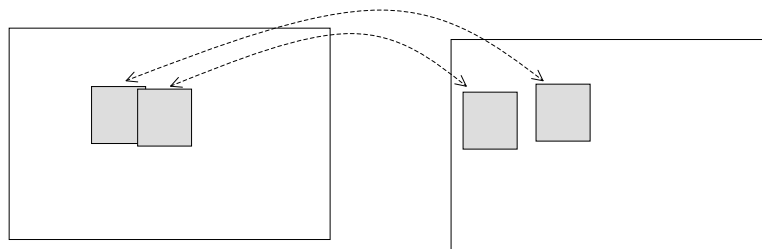11/5/2003 CS 461, Copyright G.D. Hager

MATCHING AND CORRESPONDENCE

- **Two major approaches**
  - feature-based
  - region based

In region-based matching, the idea is to pick a region in the image and attempt to find the matching region in the second image by maximizing the some measure:
  1. normalized SSD
  2. SAD
  3. normalized cross-correlation

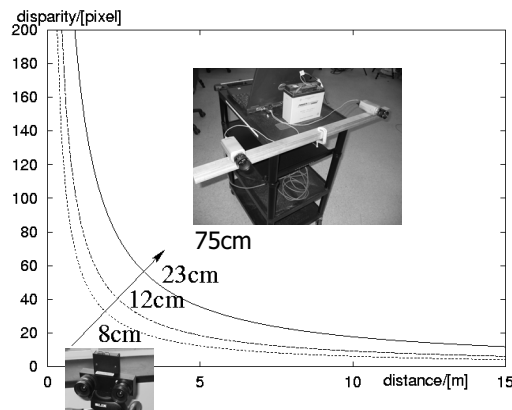11/5/2003　　　　　　　CS 461, Copyright G.D. Hager

---

Computing the Disparity Range

- The first step in correspondence search is to compute the range of disparities to search
  - The *horopter* is the set of distances which have disparity zero (or very close to zero) for a verged system. Human stereo only takes place within the horopter.

- We assume a non-verged system. Therefore, we have
  - $d = f\, b/z$
  - given a range z_{min} to z_{max}, we calculate
    - $d_{min} = f\, b / z_{max}$
    - $d_{max} = f\, b/ z_{min}$

- Thus, for each point u_l in the left image, we will search points $u_l + d_{min}$ to $u_l + d_{max}$ in the right.

- Note we can turn this around and start at a point u_r and search from $u_r - d_{max}$ to $u_r - d_{min}$

11/5/2003　　　　　　　CS 461, Copyright G.D. Hager

Design of the Stereo-System

disparity/[pixel]

75cm
23cm
12cm
8cm

distance/[m]

$$d_p = \boxed{\frac{B \cdot f}{p_x}} \cdot \frac{1}{z} [pixel]$$

Possible measures:

- Increase of the baseline (B) → size of the system

- Increase of the focal length (f) → field of view

- Decrease of the pixel-size (p$_x$) → resolution of the camera

11/5/2003   CS 461, Copyright G.D. Hager

MATCHING AND CORRESPONDENCE

- **Feature-based vs. region-based**
  - feature-based leads to sparse disparity maps
    - interpolation to fill in gaps
    - scale-space approaches to fill in gaps
  - region-based matching only works where there is texture
    - compute a confidence measure for regions
    - apply continuity or match ordering constraints
  - region matching can be sensitive to changes in surface orientation
  - feature-based can be sensitive to feature "drop-outs"

11/5/2003   CS 461, Copyright G.D. Hager

Region-Based Matching Metrics

- An obvious solution: minimize the sum of squares
  - think of R and R' as a region and a candidate region in vector form
  - SSD = $|| R - R' ||^2$ = $|| R ||^2 - 2 R . R' + || R' ||^{\wedge}2$
  - Note that we can change the SSD by making the image brighter or dimmer, or changing contrast
  - As a result, it is common to
    - subtract the mean of both images (removes brightness)
    - normalize by variance (removes contrast)
    - Note taking two derivatives (e.g. a Laplacian) has roughly the same effect!
  - In this case, minimizing SSD is equivalent to maximizing R. R'
    - this is the normalized cross correlation!

- Both SSD and NCC are sensitive to outliers
  - SAD = 1 . $|R - R'|$ is less sensitive to outliers and thus more robust
  - it is also easier to compute.

Other Matching Metrics

- rank transformation:
  - The value of a window is the # of values less than center pixel
  - Compare values using SAD or SSD on transformed image
  - Invariant over rotations, reflection, and any monotone transformation of gray values
- census transformation
  - A window becomes a bit string based on comparison to center pixel
  - Compare using Hamming distance (# of bits that differ)

  - Note both of these are easy to implement in hardware

Region Matching

- For each pixel (i,j) of the left image and offset $\nabla i, \nabla j$ in disparity range

  – compute $d(\nabla i, \nabla j) = \sum_{k,l} \psi(I_l(i+k,j+l), I_r(i+k+\nabla i, j+l+\nabla j))$

  – the disparity is the value $(\nabla i, \nabla j)$ that minimizes d

- The result of performing this search over every pixel is the *disparity map.*

- Often, this map is computed at different scales, by performing reduction using a Gaussian pyramid

11/5/2003   CS 461, Copyright G.D. Hager

Match Metric Summary

MATCH METRIC	DEFINITION
Normalized Cross-Correlation (NCC)	$\dfrac{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)\cdot\left(I_2(u+d,v)-\bar{I}_2\right)}{\sqrt{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)^2 \cdot \sum_{u,v}\left(I_2(u+d,v)-\bar{I}_2\right)^2}}$
Sum of Squared Differences (SSD)	$\sum_{u,v}\left(I_1(u,v)-I_2(u+d,v)\right)^2$
Normalized SSD	$\sum_{u,v}\left(\dfrac{\left(I_1(u,v)-\bar{I}_1\right)}{\sqrt{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)^2}} - \dfrac{\left(I_2(u+d,v)-\bar{I}_2\right)}{\sqrt{\sum_{u,v}\left(I_2(u+d,v)-\bar{I}_2\right)^2}}\right)^2$
Sum of Absolute Differences (SAD)	$\sum_{u,v}\left|I_1(u,v)-I_2(u+d,v)\right|$
Zero Mean SAD	$\sum_{u,v}\left|(I_1(u,v)-\bar{I}_1)-(I_2(u+d,v)-\bar{I}_2)\right|$
Rank	$I'_k(u,v)=\sum_{m,n} I_k(m,n) < I_k(u,v)$ $\sum_{u,v}\left(I'_1(u,v)-I'_2(u+d,v)\right)$
Census	$I'_k(u,v)=BITSTRING_{m,n}\left(I_k(m,n)<I_k(u,v)\right)$ $\sum_{u,v} HAMMING\left(I'_1(u,v),I'_2(u+d,v)\right)$

Remember, these two are actually the same

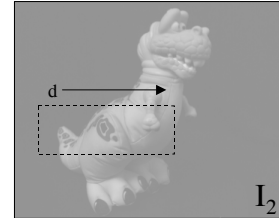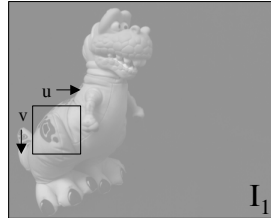11/5/2003   CS 461, Copyright G.D. Hager

Correspondence Search Algorithm
(simple version for CC)

I_1 I_2

```
For i = 1:nrows
  for j=1:ncols
        best(i,j) = -1
      for k = mindisparity:maxdisparity
        c = CC(I1(i,j),I2(i,j+k),winsize)
        if (c > best(i,j))
              best(i,j) = c
              disparities(i,j) = k
         end
        end
  end
end
```

O(nrows * ncols * disparities * winx * winy)

Correspondence Search Algorithm
(efficient version for CC)

```
best = -ones(size(im))
disp = zeros(size(im))
for k = mindisparity:maxdisparity
      prod = I1(:,overlap) .* I2(:,k+overlap)
      CC = box(prod,winsize)
      better = CC > best;
      disp = better .* k + (1-better).*disp;
end
```

O(disparities * nrows * ncols)

An Additional Twist

- Note that searching from left to right *is not the same* as searching from right to left.

- As a result, we can obtain a somewhat independent disparity map by flipping the images around.

- The results should be the same map up to sign.

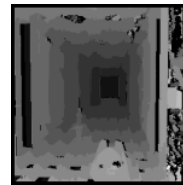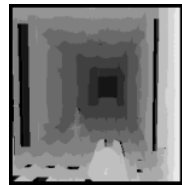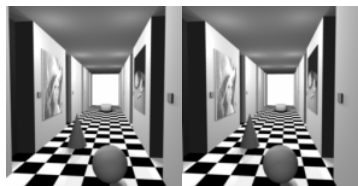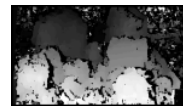- LRCheck: $\text{disp}_{lr}(i,j) = - \text{disp}_{rl}(i,j+\text{disp}_{lr}(i,j))$

Example Disparity Maps

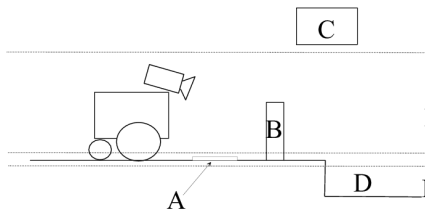SSD   ZNNC
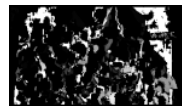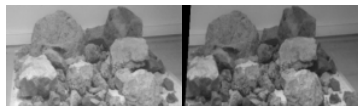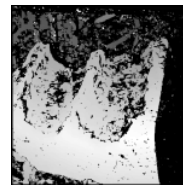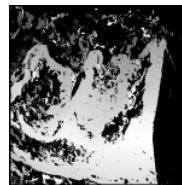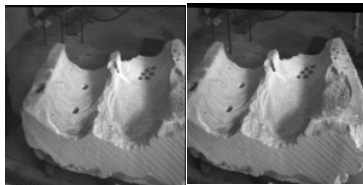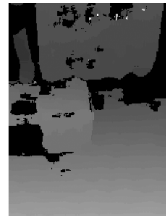
Obstacle Detection

Problem to solve:

Distinguish between relevant obstacles (B,D) and irrelevant (A,C) obstacles
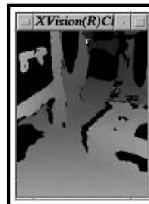
Obstacle Detection (cont'd)

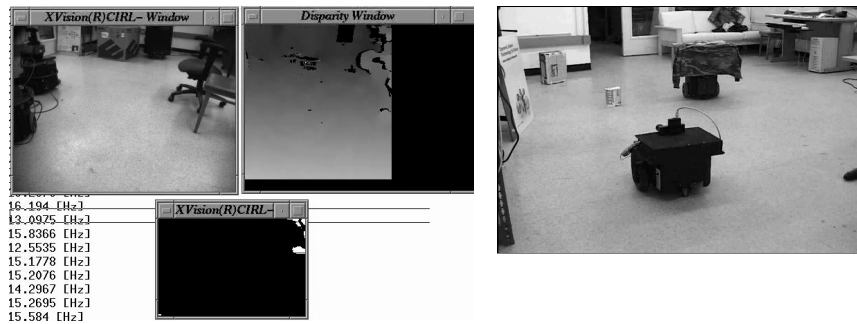# Obstacle Detection&Avoidance

---

BINOCULAR STEREO SYSTEM
(2D VIEW)

It is not hard to
show that when
we rotate the cameras
inward, corresponding
points no longer lie on a
scan line

**Nonverged
Stereo System**

**Verged
Stereo System**
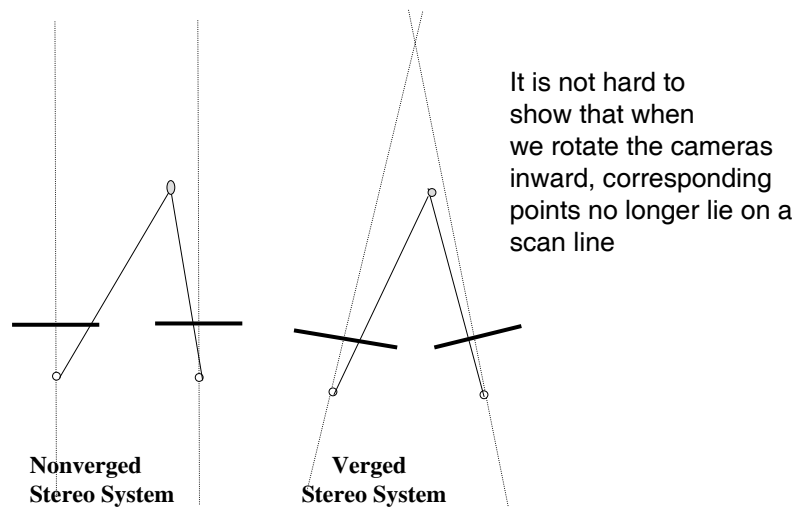
More Generally

If we think about it, something is fishy:
      a point has 3 coordinates
      two camera observations have four coordinates
therefore:
      if we choose a point in one image (2 coords),
      then there is only one degree of freedom left

There is in fact a geometric constraint that we can exploit to improve the performance of stereo (and in fact do even more)

EPIPOLAR GEOMETRY

- For an image point C in the left image plane consider the plane determined by the left image focal point A the right image focal point B and the point C. Call this the epipolar plane for image point C with respect to a particular binocular stereo configuration.

EPIPOLAR GEOMETRY

- In order to guarantee intersection of projective rays produced from the left and right image planes, the point in the right image plane corresponding to C must lie on the intersection of the epipolar plane with the right image plane.

C

A

B

EPIPOLAR PLANE

EPIPOLAR LINE

THE EPIPOLAR CONSTRAINT

line along which
the physical point
must lie (projection line)

x

x

x

the image of the
projection line

an observed point

EPIPOLAR GEOMETRY
('SCANLINE COHERENT' STEREO SYSTEM)

EPIPOLAR GEOMETRY
(VERGED IN)

Trick to find the epipolar line:

2 known points on the line are

1. the projection of the focal point e of the other camera, since this is the start of the line where the corresponding point can be

2. the position of any projection of the point along the projection line in the reference camera

P

p   e   e'   p'

O_L ◄——— T ———► O_R

EPIPOLAR GEOMETRY: DERIVATION

$$(P_l-T)^T \cdot (T \times P_l) = 0$$
$$P_r^T R (T \times P_l) = 0$$
$$P_r^T R \, sk(T) \, P_l = 0$$
$$P_r^T E \, P_l = 0$$

where $E = R \, sk(T)$

$$sk(T) = \begin{pmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{pmatrix}$$

$$P_r = R(P_l - T)$$

The matrix E is called the *essential* matrix and completely describes the epipolar geometry of the stereo pair

a natural link between the epipolar constraint and extrinsic parameters

11/5/2003   CS 461, Copyright G.D. Hager

EPIPOLAR GEOMETRY: DERIVATION

Note that E is invariant to the scale of the points, therefore we also have

$$p_r^t E \, p_l = 0$$

where p denotes the (metric) image projection of P
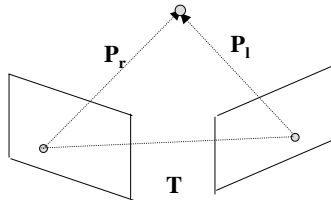
Now if H denotes the internal calibration, converting from metric to pixel coordinates, we have further that

$$r_r^t H^{-t} E \, H^{-1} r_l = r_r^t F \, r_l = 0$$

where r denotes the *pixel* coordinates of p. F is called the *fundamental matrix*

$$P_r = R(P_l - T)$$

11/5/2003   CS 461, Copyright G.D. Hager

16

EPIPOLAR GEOMETRY: COMPUTATION

$$p_r{}^t E\; p_l = 0 \quad \text{or} \quad r_r{}^t F\; r_l = 0$$

Note that, given a correspondence, we can form a linear
constraint on E (or F). Both E and F are only unique up
to scale, therefore we need 9-1 = 8 matches, then we can
form a system of the form

$$C\; e = 0 \quad \text{where e is the vector of 9 values in E}$$

Using SVD, we can write $C = U\; D\; V^t$

E (or F) is the column of V corresponding to the least singular
value of C.

WHY?

E (or F) is supposed to be rank deficient; to enforce this, we
can compute the SVD of E (or F), set the smallest singular
value to 0, then multiply the components to get the corrected F

EPIPOLAR GEOMETRY: STEREO CORRESPONDENCE

$$p_r{}^t E\; p_l = 0 \quad \text{or} \quad r_r{}^t F\; r_l = 0$$

One of the important uses of epipolar geometry is that
it greatly reduces the complexity of stereo. Given a match
in the left image, the appropriate place to look for a match
in the right is along the corresponding epipolar line.

Alternatively, it is possible to use epipolar structure to *warp*
the image to have parallel epipolar geometry, making stereo
search a trivial scan-line search.

EPIPOLAR GEOMETRY: RECONSTRUCTION

$$p_r{}^t E\, p_l = 0 \quad \text{or} \quad r_r{}^t F\, r_l = 0$$

One additional useful fact is that we can use epipolar geometry for stereo calibration of a sort.

First, note that $E^t E$ involves only translation and that $tr(E^t E) = 2\,\|T\|^2$

So, if we normalize by sqrt(tr($E^t E$)/2), we compute a new matrix E' which has unit norm translation T' up to sign.

We can solve for T' from E' (or T from E for that matter)

Now define $w_i = E'_i \times T'$ and $R_i = w_i + w_j \times w_k$

The three values of R_i for all combinations of 1,2,3 are the rows of the rotation matrix.

How to Change Epipolar Geometry

Image rectification is the computation of an image as seen by a rotated camera

Original image plane

New image plane

P

p   e   e'   p'
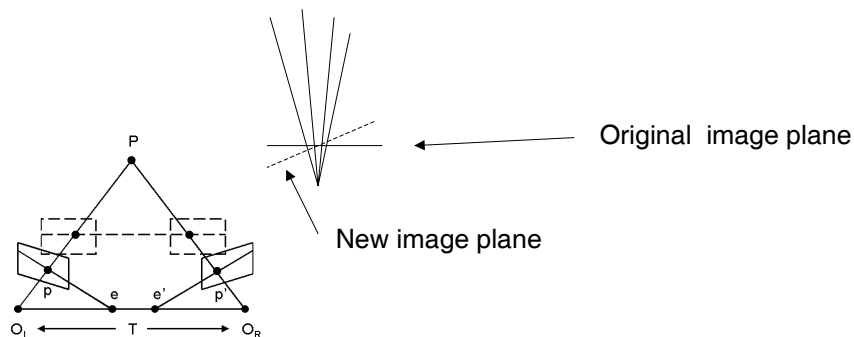
O_L ←— T —→ O_R

Using E to get Nonverged Stereo

- From E we get R and T such that $^l p = {^l R_r}\, ^r p + {^l T}\, k$

- Note that T is really the direction we'd like the camera baseline to point in (note we don't care about its norm)

- Let $R_x = T$
- Let $R_y = (0,0,1) \times T\, /\, |T \times (0,0,1)|$
- Let $R_z = R_x \times R_y$

- Now, $R = [R_x, R_y, R_z]'$ takes point from the left camera to a nonverged camera system, so we have

- $^{newl}R = R$, $^{newr}R = R\, ^l R_r$
    - (note the book uses the transpose of this, i.e. the rotation of the frame rather than the points)

Rectification: Basic Algorithm

1. Create a mesh of pixel coordinates for the rectified image
2. Turn the mesh into a list of homogeneous points
3. Project *backwards* through the intrinsic parameters to get unit focal length values
4. Rotate these values back to the current camera coordinate system.
5. Project them *forward* through the intrinsic parameters to get pixel coordinates again.
6. Sample at these points to populate the rectified image.

Rectification Results

.2 rad

.4 rad

.6 rad
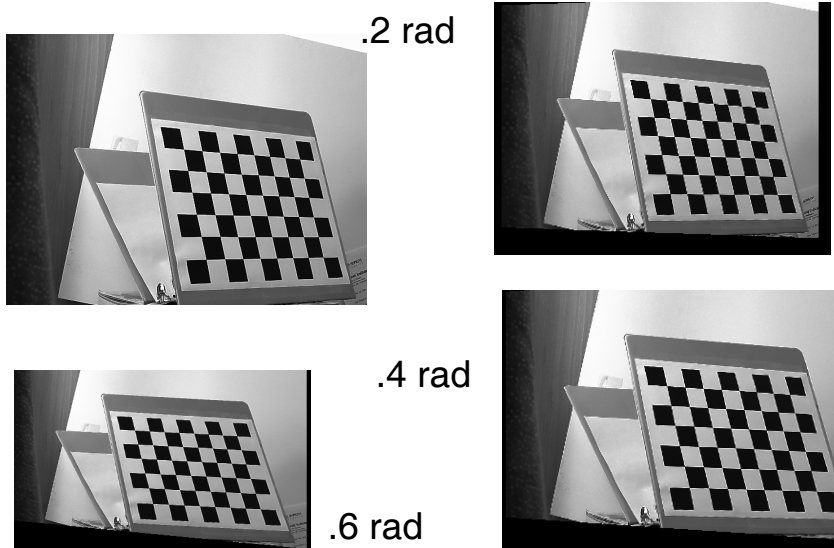
EPIPOLAR GEOMETRY: RECONSTRUCTION

$$p_r^t\, E\, p_l = 0 \quad \text{or} \quad r_r^t\, F\, r_l = 0$$

Putting all this together, we get the following algorithm:

1. **Find 8 or more correspondences and compute E (note we need internal parameters to do this).**
2. **Given E, compute T' and R.**
3. **Rectify the image using T' and R.**
4. **Now, do standard nonverged stereo.**
5. **Check the sign of the depth in the left and right images; if not both positive, adjust signs in E or T' (see T&V pg. 166).**

Result: We can reconstruct up to scale using only camera images, provided we know internal parameters

THE FUNDAMENTAL MATRIX AND RECONSTRUCTION

$$p_r^t E p_l = 0 \quad \text{or} \quad r_r^t F r_l = 0$$

If we do not know the internal parameters, then the 8 point algorithm can only be used to compute F.

Unfortunately, F has less structure; what we can show is that we can only reconstruct up to a projective transformation.

SUMMARY: SIMPLE STEREO

Given two cameras with *known* relative positions in space and known internal parameters:

1. **Rectify the two images using epipolar geometry.**
2. **Compute image correspondences using either feature-based or correlation-based matching**
3. **Convert resulting pixel coordinates to metric coordinates using internal calibration**
4. **Use triangulation equation to compute distance**
   1. **If unknown baseline, simply invert disparity (reconstruction up to a scale factor)**
5. **Post-process**
   1. **remove outliers (e.g. median filter)**
   2. **interpolate surface**

Real-Time Stereo

REAL-TIME STEREO SYSTEM	IMAGE SIZE	FRAME RATE	RANGE BINS	METHOD	PROCESSOR	CAMERAS
INRIA 1993	256x256	3.6 fps	32	Normalized Correlation	PeRLe-1	3
CMU iWarp 1993	256x240	15 fps	16	SSAD	64 Processor iWarp Computer	3
Teleos 1995	320x240	0.5 fps	32	Sign Correlation	Pentium 166 MHz	2
JPL 1995	256x240	1.7 fps	32	SSD	Datacube & 68040	2
CMU Stereo Machine 1995	256x240	30 fps	30	SSAD	Custom HW & C40 DSP Array	6
Point Grey Triclops 1997	320x240	6 fps	32	SAD	Pentium II 450 MHz	3
SRI SVS 1997	320x240	12 fps	32	SAD	Pentium II 233 MHz	2
SRI SVM II 1997	320x240	30+ fps	32	SAD	TMS320C60x 200MHz DSP	2
Interval PARTS Engine 1997	320x240	42 fps	24	Census Matching	Custom FPGA	2
CSIRO 1997	256x256	30 fps	32	Census Matching	Custom FPGA	2
SAZAN 1999	320x240	20 fps	25	SSAD	FPGA & Convolvers	9
Point Grey Triclops 2001	320x240	20 fps 13 fps	32	SAD	Pentium IV 1.4 GHz	2 3
SRI SVS 2001	320x240	30 fps	32	SAD	Pentium III 700 MHZ	2

MATCHING AND CORRESPONDENCE

There is no "Best" solution for correspondence

new frame-rate stereo systems use cross-correlation with left-right and right-left validation

There has been recent work on computing a "globally" optimal disparity map taking into account

occlusion

C^0 and C^1 discontinuities

ordering constraints based on continuous surfaces

SOME OTHER MAJOR PROBLEMS WITH CORRESPONDENCE (2D VIEW)
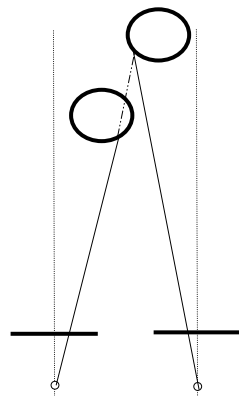
OCCLUSION
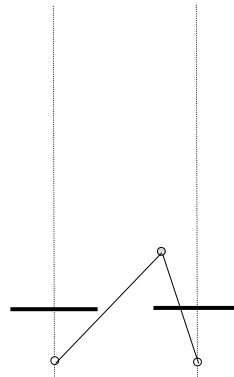
LIMITED FIELD OF VIEW

Other Problems:

- Photometric issues:
  - specularities
  - strongly non-Lambertian BRDF's

- Surface structure
  - lack of texture
  - repeating texture within horopter bracket

- Geometric ambiguities
  - as surfaces turn away, difficult to get accurate reconstruction (affine approximate can help)
  - at the occluding contour, likelihood of good match but incorrect reconstruction

Stereo Constraints

CONSTRAINT	BRIEF DESCRIPTION
1-D Epipolar Search	Arbitrary images of the same scene may be rectified based on epipolar geometry such that stereo matches lie along one-dimensional scanlines. This reduces the computational complexity and also reduces the likelihood of false matches.
Monotonic Ordering	Points along an epipolar scanline appear in the same order in both stereo images, assuming that all objects in the scene are approximately the same distance from the cameras.
Image Brightness Constancy	Assuming Lambertian surfaces, the brightness of corresponding points in stereo images are the same.
Match Uniqueness	For every point in one stereo image, there is at most one corresponding point in the other image.
Disparity Continuity	Disparities vary smoothly (i.e. disparity gradient is small) over most of the image. This assumption is violated at object boundaries.
Disparity Limit	The search space may be reduced significantly by limiting the disparity range, reducing both computational complexity and the likelihood of false matches.
Fronto-Parallel Surfaces	The implicit assumption made by area-based matching is that objects have fronto-parallel surfaces (i.e. depth is constant within the region of local support). This assumption is violated by sloping and creased surfaces.
Feature Similarity	Corresponding features must be similar (e.g. edges must have roughly the same length and orientation).
Structural Grouping	Corresponding feature groupings and their connectivity must be consistent.

Methods for Occlusion

APPROACH	RELEVANT PAPERS	BRIEF DESCRIPTION
METHODS THAT DETECT OCCLUSION		
Depth Map Discontinuities		Discontinuities in the depth map are assumed to be occlusion regions.
Left-Right Matching	[15], [29]	Matches that are not unique when estimated from left-to-right and right-to-left are assumed to be in occlusion regions.
Ordering Constraint	[50], [74], [88]	Oppositely ordered adjacent matches indicate occlusion.
Intensity Edges	[11], [17], [60]	Intensity edges are assumed to correspond to occlusion boundaries.
METHODS THAT REDUCE SENSITIVITY TO OCCLUSION		
Robust Similarity Criterion	[7], [66], [70], [74], [89]	Robust methods are employed in the match metric to reduce sensitivity to occlusion.
Adaptive Regions of Support	[31], [39], [52], [68], [71], [90]	Regions of support are adaptively resized, reshaped or diffused to obtain the best match and minimize the effects of occlusion.
METHODS THAT MODEL OCCLUSION GEOMETRY		
Global Occlusion Modeling	[5], [10], [36], [65]	Occlusion is modeled and included in the match procedure, usually using dynamic programming.
Multiple Cameras	[58], [67]	Multiple cameras ensure that every point in the scene is visible by at least two cameras.
Active Vision	[16], [49], [61], [64]	The camera or stereo rig is moved in order to detect occlusion and to determine occlusion width.

Local vs. Global Matching

Comparative results on images from the University of Tsukuba, provided by Scharstein and Szeliski [69]. Left to right: left stereo image, ground truth, Muhlmann et al.'s area correlation algorithm [57], dynamic programming (similar to Intille and Bobick [36]), Roy and Cox's maximum flow [65] and Komolgorov and Zabih's graph cuts [45].

CS 461, Copyright G.D. Hager