Previously

- Image Primitives (feature points, lines, contours)
- Today:
  - How to match primitives between two (multiple) views
  - Goals: 3D reconstruction, recognition

- Stereo matching and reconstruction (canonical configuration)
- Epipolar Geometry (general two view setting)
Why Stereo Vision?

- 2D images project 3D points into 2D:

  ![Diagram showing projection of 3D points into 2D](image)

  - 3D points on the same viewing line have the same 2D image:
    - 2D imaging results in depth information loss
Canonical Stereo Configuration

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

\[ \frac{Z}{T} = \frac{Z - f}{T - x_l - x_r} \]

\[ Z = \frac{fT}{\text{disparity}} \]
Random Dot Stereo-grams

B. Julesz: showed that the depth can be perceived in the absence of any identifiable objects in correspondence
Autostereograms

- Depth perception from one image

- Viewing trick the brain by focusing at the plane behind - match can be established perception of 3D
Correspondence Problem

- Two classes of algorithms:
  - Correlation-based algorithms
    - Produce a DENSE set of correspondences
  - Feature-based algorithms
    - Produce a SPARSE set of correspondences
Stereo – Photometric Constraint

- Same world point has same intensity in both images.
  - Lambertian fronto-parallel
  - Issues (noise, specularities, foreshortening)

- Difficulties – ambiguities, large changes of appearance, due to change of viewpoint, non-uniqueness
Stereo Matching

For each 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: **improvement match windows**

What if?
Comparing Windows:

For each window, match to closest window on epipolar line in other image.

\[
SSD = \sum_{[i,j] \in R} (f(i, j) - g(i, j))^2
\]

\[
C_{fg} = \sum_{[i,j] \in R} f(i, j)g(i, j)
\]

Most popular

(slides O. Camps)
Comparing Windows:

Minimize \[ \sum_{[i,j] \in R} (f(i, j) - g(i, j))^2 \]  
Sum of Squared Differences

Maximize \[ C_{fg} = \sum_{[i,j] \in R} f(i, j)g(i, j) \]  
Cross correlation

It is closely related to the SSD:

\[ SSD = \sum_{[i,j] \in R} (f - g)^2 = \]
\[ = \sum_{[i,j] \in R} f^2 + \sum_{[i,j] \in R} g^2 - 2 \sum_{[i,j] \in R} f g \]
Window size

Effect of window size

Better results with *adaptive window*


(S. Seitz)
Stereo results

- Data from University of Tsukuba

Scene

Ground truth

(Seitz)
Results with window correlation

Window-based matching (best window size)  Ground truth  (Seitz)
Results with better method

State of the art

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

Ground truth

(Seitz)
More of advanced stereo (later)

- Ordering constraint
- Dynamic programming
- Global optimization
Two view – General Configuration

- Motion between the two views is not known

Given two views of the scene recover the unknown camera displacement and 3D scene structure
Pinhole Camera Imaging Model

- 3D points $X = [X, Y, Z, W]^T \in \mathbb{R}^4, \quad (W = 1)$
- Image points $x = [x, y, z]^T \in \mathbb{R}^3, \quad (z = 1)$
- Perspective Projection $\lambda x = X$
  \[
  \lambda = Z \quad x = \frac{X}{Z} \quad y = \frac{Y}{Z}
  \]
- Rigid Body Motion $\Pi = [R, T] \in \mathbb{R}^{3 \times 4}$
- Rigid Body Motion + Persp. projection $\lambda x = \Pi X = [R, T]X$
  \[
  \lambda x' = K \Pi_0 X = \begin{bmatrix}
  fs_x & fs_\theta & ox \\
  0 & fs_y & oy \\
  0 & 0 & 1
  \end{bmatrix}
  \begin{bmatrix}
  R \\
  T
  \end{bmatrix}
  \begin{bmatrix}
  X \\
  Y \\
  Z \\
  1
  \end{bmatrix}
  \]
Rigid Body Motion – Two Views

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

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

\[ \lambda_1 \mathbf{x}_1 = \mathbf{X} \]

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

\[ \lambda \mathbf{x} = \mathbf{\Pi} \mathbf{X} = [R, T] \mathbf{X} \]

\[ \mathbf{\Pi} = [R, T] \in \mathbb{R}^{3 \times 4} \]
3D Structure and Motion Recovery

Euclidean transformation

\[ \lambda_2 x_2 = R \lambda_1 x_1 + T \]

Measurements
Corresponding points
unknowns

\[ \sum_{j=1}^{n} \|x^j_1 - \pi(R_1, T_1, X)\|^2 + \|x^j_2 - \pi(R_2, T_2, X)\|^2 \]

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
Cross product between two vectors in

\[ c = a \times b \]

where

\[
\begin{bmatrix}
-a_3b_2 + a_2b_3 \\
a_3b_1 - a_1b_3 \\
-a_2b_1 + a_1b_2
\end{bmatrix}
\]

\[
\hat{a} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}
\]
Epipolar Geometry

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

\[ \lambda_2 x_2 = R \lambda_1 x_1 + T / \hat{x}_2 T \]

- **Algebraic Elimination of Depth** [Longuet-Higgins ’81]:
  \[ x_2^T \hat{T} R x_1 = 0 \]

- **Essential matrix**
  \[ E = \hat{T} R \]
**Epipolar Geometry**

- Epipolar lines \( l_1, l_2 \)
- Epipoles \( e_1, e_2 \)

\[
x_2^T E x_1 = 0
\]

\[
E = \hat{T} R
\]

- Additional constraints

\[
l_1 \sim E^T x_2 \quad l_i^T x_i = 0 \quad l_2 \sim E x_1
\]

\[
E e_1 = 0 \quad l_i^T e_i = 0 \quad e_2 E^T = 0
\]
Characterization of Essential Matrix

$$x^T_2 \hat{T} Rx_1 = 0$$

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

$$x^T_2 \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} x_1 = 0$$

(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
  \[ \min_E \sum_{j=1}^{n} (x_2^j E x_1^j)^2 \]

- Space of all Essential Matrices is 5 dimensional
- 3 DOF Rotation, 2 DOF – Translation (up to scale !)

- Denote \( a = x_1 \otimes x_2 \)

\[ a = [x_1 x_2, x_1 y_2, x_1 z_2, y_1 x_2, y_1 y_2, y_1 z_2, z_1 x_2, z_1 y_2, z_1 z_2]^T \]

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

- Rewrite
  \[ a^T E^s = 0 \]

- Collect constraints from all points
  \[ \chi E^s = 0 \]

\[ \min_E \sum_{j=1}^{n} x_2^j E x_1^j \quad \Rightarrow \quad \min_{E^s} \| \chi E^s \|^2 \]
Estimating Essential Matrix

\[ \min_E \sum_{j=1}^{n} x_2^T E x_1^j \rightarrow \min_{E_s} \| \chi E^s \|^2 \]

Solution is
- Eigenvector associated with the smallest eigenvalue of \( \chi^T \chi \)
- If \( \text{rank}(\chi^T \chi) < 8 \) degenerate configuration

\( E_s \) estimated using linear least squares
unstack \( E_s \rightarrow F \)

Projection on to Essential Space

(Project onto a space of Essential Matrices)
If the SVD of a matrix \( F \in \mathbb{R}^{3\times3} \) is given by \( F = U \text{diag}(\sigma_1, \sigma_2, \sigma_3)V^T \) then the essential matrix which minimizes the Frobenius distance \( \|E - F\|_F^2 \) is given by \( E = U \text{diag}(\sigma, \sigma, 0)V^T \) with \( \sigma = \frac{\sigma_1 + \sigma_2}{2} \)
Pose Recovery from Essential Matrix

Essential matrix

\[ E = \hat{T} R \]

(Pose Recovery)

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

\[ E = U \Sigma V^T \]

\[
\begin{align*}
(\hat{T}_1, R_1) &= (UR_Z(\frac{\pi}{2})\Sigma U^T, UR_Z^T(\frac{\pi}{2})V^T) \\
(\hat{T}_2, R_2) &= (UR_Z(-\frac{\pi}{2})\Sigma U^T, UR_Z^T(-\frac{\pi}{2})V^T)
\end{align*}
\]

\[ \Sigma = diag([1, 1, 0]) \quad R_z(\frac{\pi}{2}) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

• Twisted pair ambiguity \((R_2, T_2) = (e^{\hat{u}\pi} R_1, -T_1)\)
Two view linear algorithm - summary

\[ E = \{ \hat{T}R | R \in SO(2), T \in S^2 \} \]

- Solve the **LLSE** problem:
  \[ \min_E \sum_{j=1}^{n} \left( x_2^j \hat{E} x_1^j \right)^2 \rightarrow \chi E^s = 0 \]

- Solution eigenvector associated with smallest eigenvalue of \( \chi^T \chi \)

- Compute SVD of \( F \) recovered from data
  \[ E^s \rightarrow F \quad F = U \Sigma V^T \]

- Project onto the essential manifold:
  \[ \Sigma' = \text{diag}(1, 1, 0) \quad E = U \Sigma' V^T \]

- Recover the unknown pose:
  \[ (\hat{T}, R) = (UR_Z(\pm \frac{\pi}{2}) \Sigma U^T, UR_Z^T(\pm \frac{\pi}{2}) V^T) \]
Pose Recovery

• There are two pairs \((R, T)\) corresponding to essential matrix.

• There are two pairs \((R, T)\) corresponding to essential matrix.

• Positive depth constraint disambiguates the impossible solutions.

• Translation has to be non-zero.

• 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

\[ \lambda_2 x_2 = R \lambda_1 x_1 + \gamma T \]

- Eliminate one of the scale’s

\[ \lambda_1 \hat{x}_2^j R \hat{x}_1^j + \gamma \hat{x}_2^j T = 0, \quad j = 1, 2, \ldots, n \]

- Solve LLSE problem

\[ M^j \bar{\lambda}^j \hat{\lambda}^j \hat{\lambda}^j = \begin{bmatrix} \hat{x}_2^j R \hat{x}_1^j, \hat{x}_2^j T \end{bmatrix} \begin{bmatrix} \lambda_1^j \\ \gamma \end{bmatrix} = 0 \]

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
Example

Two views

Point Feature Matching
Example

Epipolar Geometry

Camera Pose
and
Sparse Structure Recovery
Epipolar Geometry - Planar case

- Plane in first camera coordinate frame

\[ aX + bY + cZ + d = 0 \]

\[ \frac{1}{d}N^T X = 1 \]

\[ \lambda_2 x_2 = R\lambda_1 x_1 + T \]

\[ \lambda_2 x_2 = (R + \frac{1}{d}TN^T)\lambda_1 x_1 \]

\[ x_2 \sim Hx_1 \]

\[ H = (R + \frac{1}{d}TN^T) \]

Linear mapping relating two corresponding planar points in two views
Decomposition of H (into motion and plane normal)

- Algebraic elimination of depth \( \hat{x}_2 H x_1 = 0 \)
- can be estimated linearly \( H_L = \lambda H \)
- Normalization of \( H = H_L / \sigma_3 \)
- Decomposition of H into 4 solutions \( H = (R + \frac{1}{d}TN^T) \)

| \( R_1 = W_1 U_1^T \) | \( R_3 = R_1 \) | \( R_2 = W_2 U_2^T \) | \( R_4 = R_2 \) |
| \( N_1 = \hat{w}_2 u_1 \) | \( N_3 = -N_1 \) | \( N_2 = \hat{w}_2 u_2 \) | \( N_4 = -N_2 \) |
| \( \frac{1}{d} T_1 = (H - R_1) N_1 \) | \( \frac{1}{d} T_3 = -\frac{1}{d} T_1 \) | \( \frac{1}{d} T_2 = (H - R_2) N_2 \) | \( \frac{1}{d} T_4 = -\frac{1}{d} T_2 \) |

\[
H^T H = V \Sigma V^T \quad V = [v_1, v_2, v_3] \quad \Sigma = \text{diag}(\sigma_1^2, \sigma_2^2, \sigma_3^2)
\]

\[
u_1 = \frac{\sqrt{1-\sigma_3^2}v_1 + \sqrt{\sigma_1^2-1}v_3}{\sqrt{\sigma_1^2-\sigma_3^2}} \quad u_2 = \frac{\sqrt{1-\sigma_3^2}v_1 - \sqrt{\sigma_1^2-1}v_3}{\sqrt{\sigma_1^2-\sigma_3^2}}
\]

\[
U_1 = [v_2, u_1, \hat{w}_2 u_1], \quad W_1 = [Hv_2, Hu_1, \hat{H}v_2 Hu_1]; \quad U_2 = [v_2, u_2, \hat{w}_2 u_2], \quad W_2 = [Hv_2, Hu_2, \hat{H}v_2 Hu_2].
\]
Uncalibrated Camera

\[ x' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K x = \begin{bmatrix} f s_x & f s_\theta & o_x \\ 0 & f s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \]

Linear transformation \( K \)

Pixel coordinates

(0, 0)

Calibrated coordinates

(0_\text{x}, 0_\text{y})

\( f s_x \) \( f s_y \) \( o_x \) \( o_y \)
Uncalibrated Epipolar Geometry

- Epipolar constraint
  \[ x'^T_2 K^{-T} \hat{T} RK^{-1} x'_1 = 0 \]

- Fundamental matrix
  \[ F = K^{-T} \hat{T} RK^{-1} \]
Properties of the Fundamental Matrix

\[ x'^T_2 F x'_1 = 0 \]

- Epipolar lines $l_1, l_2$
- Epipoles $e_1, e_2$

\[
\begin{align*}
l_1 & \sim F^T x'_2 \\
F e_1 & = 0 \\
l'_i x'_i & = 0 \\
l'_i e_i & = 0 \\
l_2 & \sim F x'_1 \\
e'_2 F & = 0
\end{align*}
\]
Epipolar Geometry for Parallel Cameras

Epipoles are at infinite
Epipolar lines are parallel to the baseline
Properties of the Fundamental Matrix

A nonzero matrix $F \in \mathbb{R}^{3 \times 3}$ is a fundamental matrix if it has a singular value decomposition (SVD) $F = U \Sigma V^T$ with

$$\Sigma = \text{diag}\{\sigma_1, \sigma_2, 0\}$$

for some $\sigma_1, \sigma_2 \in \mathbb{R}_+$. There is little structure in the matrix $F$ except that

$$\det(F) = 0$$
Estimating Fundamental Matrix

• Find such $F$ that the epipolar error is minimized

$$
\min_F \sum_{j=1}^{n} (x_j^T F x_j)^2
$$

- Pixel coordinates

• Fundamental matrix can be estimated up to scale

• Denote $a = x'_1 \otimes x'_2$

$$
a = [x_1 x_2, x_1 y_2, x_1 z_2, y_1 x_2, y_1 y_2, y_1 z_2, z_1 x_2, z_1 y_2, z_1 z_2]^T
$$

$$
F^s = [f_1, f_4, f_7, f_2, f_5, f_8, f_3, f_6, f_9]^T
$$

• Rewrite

$$
a^T F^s = 0
$$

• Collect constraints from all points

$$
\chi F^s = 0
$$

$$
\min_F \sum_{j=1}^{n} (x_2^j T F x_1^j)^2 \rightarrow \min_{F^s} \| \chi F^s \|^2
$$
Two view linear algorithm – 8-point algorithm

- Solve the **LLSE** problem:
  \[
  \min_F \sum_{j=1}^{n} (x_2^j F x_1^j)^2 \rightarrow \min_{F_S} \| \chi F \| \| F \|^2 = 0
  \]

- Solution eigenvector associated with smallest eigenvalue of \( \chi^T \chi \)

- Compute SVD of \( F \) recovered from data
  \[
  F = U \Sigma V^T \quad \Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3)
  \]

- **Project** onto the essential manifold:
  \[
  \Sigma' = \text{diag}(\sigma_1, \sigma_2, 0) \quad F = U \Sigma' V^T
  \]

- cannot be unambiguously decomposed into pose and calibration
  \[
  F = K^{-T} \hat{R} K^{-1}
  \]
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 \( \rho \)
- Then number of samples needed to get an outlier free sample with probability \( \rho \)

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

<table>
<thead>
<tr>
<th>Outlier ratio</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
</tr>
</thead>
<tbody>
<tr>
<td>seven-point algorithm</td>
<td>13</td>
<td>35</td>
<td>106</td>
<td>382</td>
<td>1827</td>
<td>13696</td>
</tr>
<tr>
<td>eight-point algorithm</td>
<td>17</td>
<td>51</td>
<td>177</td>
<td>766</td>
<td>4570</td>
<td>45658</td>
</tr>
</tbody>
</table>

- 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

- $s = \infty$, sample_count = 0;
- While $s >$ sample_count repeat
  - choose a sample and count the number of inliers
  - set $\epsilon = 1 - \frac{\text{number_of_inliers}}{\text{total_number_of_points}}$
  - set $s$ from $\epsilon$ and $\rho = 0.99$
  - increment sample_count by 1
- terminate
Robust technique

(a) correspondences.  (b) identified inliers.  (c) identified outliers.
More correspondences and Robust matching

- Select set of putative correspondences $x_1^j, x_2^j$

$$x_2^T F x_1 = 0$$

- 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

$$d_j^2 = \frac{(x_2^j)^T F_k x_1^j)^2}{\| \hat{e}_3^T F x_1^j \|^2 + \| x_2^j^T \hat{F} \hat{e}_3 \|^2}$$

$$d_j \leq \tau_d$$

4. Count the number of points in the consensus set
RANSAC in action

\[ d_j \leq \tau_d \]

Inliers

\[ d_j > \tau_d \]

Outliers
Epipolar Geometry

- Epipolar geometry in two views
- **Refined epipolar geometry using nonlinear estimation of F**
- The techniques mentioned so far simple linear least-squares estimation methods. The obtained estimates are used as initialization for non-linear optimization methods
Special Motions – Pure Rotation

- Calibrated Two views related by rotation only \( \hat{x}_2 R x_1 = 0 \)
  \[ \lambda_2 x_2 = R \lambda_1 x_1 \]

- Uncalibrated Case \( x' = K x \quad x = K^{-1} x' \)
  \[ \hat{x}'_2 H x'_1 = \hat{x}'_2 K R K^{-1} x'_1 = 0 \]

- Mapping to a reference view – H can be estimated

- Mapping to a cylindrical surface - applications – image mosaics
Projective Reconstruction

- Euclidean Motion Cannot be obtained in uncalibrated setting (F cannot be uniquely decomposed into R, T and K matrix)
- Can we still say something about 3D?
- Notion of the projective 3D structure
  (study of projective geometry)

Euclidean reconstruction  Projective reconstruction
Euclidean vs Projective reconstruction

- **Euclidean reconstruction** – true metric properties of objects; lengths (distances), angles, parallelism are preserved.
- Unchanged under rigid body transformations.
- => Euclidean Geometry – properties of rigid bodies under rigid body transformations, similarity transformation.

- **Projective reconstruction** – lengths, angles, parallelism are **NOT** preserved – we get distorted images of objects – their distorted 3D counterparts --> 3D projective reconstruction.
- => Projective Geometry.