Mesh Processing Pipeline

Scan  Reconstruct  Clean  Remesh
Digital Michelangelo Project

1G sample points → 8M triangles

4G sample points → 8M triangles
Local Sightseeing
Surface Reconstruction

physical model

Scanning devices

Registration

acquired point cloud

Implicit reconstruction

reconstructed model
Range Scanning Systems

**Passive:** Stereo matching

Find and match features in both images

Problem: Needs features to match
Range Scanning Systems

**Active**: Structured light

Project special b/w patterns to identify pixels

Problematic for materials / textures having strong color differences.
Range Scanning Systems

**Active:** Structured *infrared* light

Project infrared dots to identify pixels

*video*

Prone to noise

Microsoft Kinect
Range Scanning Systems

**Active**: Laser scanning, time-of-flight

Send laser pulse, time how long it takes to return

\[ r = \frac{1}{2} c \Delta t \]

Accuracy depends on how precisely can measure time

Operates on large distances – good for buildings, large scale objects
Range Scanning Systems

**Active**: Laser scanning triangulation

Sweep laser, record where pixel intensity is max.

Problematic for difficult reflectance properties (highly specular, hair)
Line of sight required to camera and laser

High accuracy
Range Scans

• Scanning generates multiple range images

• Each contain 3D points for different parts of the model in local coordinates of the scanner

• Need registration to one point cloud
Goal

set of raw scans

reconstructed model
Range Processing Pipeline

Steps

1. Initial registration
2. Pairwise registration
3. Global relaxation to spread out error
4. Generate surface

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Range Processing Pipeline

Steps

1. Initial registration
2. Pairwise registration
3. Global relaxation to spread out error
4. Generate surface
Range Processing Pipeline

Steps

1. Initial registration
2. Pairwise registration
3. Global relaxation to spread out error
4. Generate surface

Slides from: http://www.cs.princeton.eu/~bjbrown/iccv05_course/
Range Processing Pipeline

Steps

1. Initial registration
2. Pairwise registration
3. Global relaxation to spread out error
4. Generate surface

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Range Processing Pipeline

Steps

1. Initial registration
2. Pairwise registration
3. Global relaxation to spread out error
4. Generate surface

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Initial Registration

• Options:
  – Scans are calibrated – scan on a turntable
  – Manual feature selection, global coarse alignment
  – Automatic feature selection and matching

Partially Overlapping Scans

Aligned Scans

Slides from: http://www.cs.princeton.edu/~bjbrown ICCV05_course/
Non-Rigid Registration

Video

Data courtesy of C. Stoll, MPI Informatik
Pairwise Rigid Registration

Images from: "Geometry and convergence analysis of algorithms for registration of 3D shapes" by Pottman
Pairwise Rigid Registration Goal

Align two partially-overlapping point clouds given initial guess for relative transform
Aligning 3D Data

If correct correspondences are known, can find correct relative rotation/translation

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Aligning 3D Data

• How to find correspondences: User input? Feature detection? Signatures?
• Alternative: assume closest points correspond
Aligning 3D Data

• ... and iterate to find alignment
  – Iterative Closest Points (ICP) [Besl & McKay 92]

• Converges if starting position "close enough"

Method for registration of 3-D shapes
PJ Besl, ND McKay - Robotics-DL tentative, 1992 - proceedings.spiedigitallibrary.org

abstract This paper describes a general purpose, representation independent method for
the accurate and computationally efficient registration of 3-D shapes including free-form
curves and surfaces. The method handles the full six-degrees of freedom and is based on ...

Cited by 13289  Related articles  All 30 versions  Web of Science: 5350  Cite  Saved  More

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Iterative Closest Point (ICP)

- Given two scans $P$ and $Q$.
- Iterate:
  1. Find some pairs of closest points $(p_i, q_i)$
  2. Find rotation $R$ and translation $t$ to minimize

$$\min_{R,t} \sum_i \| p_i - Rq_i - t \|^2$$
Example
Finding $t$

- Define barycentered point sets
- Optimal translation vector $t$ maps barycenters onto each other

\[
\bar{p} := \frac{1}{m} \sum_{i=1}^{m} p_i \quad \bar{q} := \frac{1}{m} \sum_{i=1}^{m} q_i \\
\hat{p}_i := p_i - \bar{p} \quad \hat{q}_i := q_i - \bar{q}
\]

\[
t = \bar{p} - R\bar{q}
\]

- Proof [Horn 88]
Finding $\mathbf{R}$

- In matrix notation
  \[
  \hat{\mathbf{P}} = \begin{pmatrix} \hat{\mathbf{p}}_1^T \\ \vdots \\ \hat{\mathbf{p}}_m^T \end{pmatrix}, \hat{\mathbf{Q}} = \begin{pmatrix} \hat{\mathbf{q}}_1^T \\ \vdots \\ \hat{\mathbf{q}}_m^T \end{pmatrix}
  \]
  Frobenius norm
  \[
  \|A_{n \times m}\|_F^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij}^2
  \]

- The problem:
  \[
  \min_{\mathbf{R}} \|\hat{\mathbf{P}} \mathbf{R} - \hat{\mathbf{Q}}\|_F^2 \quad s.t. \quad \mathbf{R}^T \mathbf{R} = \mathbf{I}
  \]

- Equivalent to finding rotation matrix $\mathbf{R}$ closest to: $\hat{\mathbf{P}}^T \hat{\mathbf{Q}}$
  Orthogonal Procrustes Problem
Finding \( R \)

- Compute matrix

\[
S = \hat{P}^T \hat{Q}
\]

- Singular value decomposition (SVD) extracts rotation from \( S \)

\[
S = UV^T \quad \Rightarrow \quad R = UV^T
\]

[Schonemann '66 - “A generalized solution of the orthogonal Procrustes problem”]
Converges?

- Errors decrease monotonically [Besl & McKay 92]
- Converges to local minimum
- Good initial guess $\rightarrow$ Converges to global minimum
ICP Variants

Variants on the following stages of ICP have been proposed:

1. Selecting source points (from one or both meshes)
2. Matching to points in the other mesh
3. Weighting the correspondences
4. Rejecting certain (outlier) point pairs
5. Assigning an error metric to the current transform
6. Minimizing the error metric w.r.t. transformation
ICP Variants

Variants on the following stages of ICP have been proposed:

1. **Selecting** source points (from one or both meshes)
2. **Matching** to points in the other mesh
3. **Weighting** the correspondences
4. **Rejecting** certain (outlier) point pairs
5. **Assigning an error metric** to the current transform
6. **Minimizing** the error metric w.r.t. transformation
Point-to-Plane Error Metric

Using point-to-plane distance instead of point-to-point lets flat regions slide along each other

[Chen & Medioni 91]
Point-to-Plane Error Metric

- Error function:
  \[ E = \sum \left( (Rp_i + t - q_i) \cdot n_i \right)^2 \]

- Linearize (assume rotation is small)
  \[ R = e^H = 1 + H + \frac{1}{2} H^2 + \ldots \]

Skew-Symmetric Matrix

\[ H = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}, \quad r = \begin{pmatrix} r_x \\ r_y \\ r_z \end{pmatrix} \]

\[ Hp_i = r \times p_i \]
Point-to-Plane Error Metric

\[ E = \sum ((R\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i) \cdot \mathbf{n}_i)^2 \]

\[ R\mathbf{p}_i \approx (I + H)\mathbf{p}_i = \mathbf{p}_i + \mathbf{r} \times \mathbf{p}_i \]

Result: over-constrained linear system

\[ E \approx \sum ((\mathbf{p}_i - \mathbf{q}_i) \cdot \mathbf{n}_i + \mathbf{r} \cdot (\mathbf{p}_i \times \mathbf{n}_i) + \mathbf{t} \cdot \mathbf{n}_i)^2 \]

\[ \mathbf{r} = \begin{pmatrix} r_x \\ r_y \\ r_z \end{pmatrix} \]
Point-to-Plane Error Metric

- Over-constrained linear system

\[ \mathbf{A} \mathbf{x} = \mathbf{b}, \]

\[ \mathbf{A} = \begin{pmatrix} \mathbf{p}_1 \times \mathbf{n}_1 & \mathbf{n}_1 & \vdots \\ \mathbf{p}_2 \times \mathbf{n}_2 & \mathbf{n}_2 & \vdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} t_x \\ r_y \\ r_z \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} -(\mathbf{p}_1 - \mathbf{q}_1) \cdot \mathbf{n}_1 \\ -(\mathbf{p}_2 - \mathbf{q}_2) \cdot \mathbf{n}_2 \\ \vdots \end{pmatrix} \]

- Solve using least squares

\[ \mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{b} \]

\[ \mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \]
Real-time Example
Photograph
Global Registration Goal

• Given: \( n \) scans around an object
• Goal: align them all
• First attempt: ICP each scan to one other

Slides from: http://www.cs.princeton.edu/~bjbrown/iccv05_course/
Global Registration Goal

• Want method for distributing accumulated error among all scans
Approach #1: Avoid the Problem

• In some cases, have 1 (possibly low-resolution) scan that covers most of surface
• Align all other scans to this “anchor”  [Turk 94]
• Disadvantage: not always practical to obtain anchor scan
Approach #2: The Greedy Solution

- Align each new scan to all previous scans [Masuda 96]

- Disadvantages:
  - Order dependent
  - Doesn’t spread out error
Approach #3: The Brute-Force Solution

• While not converged:
  – For each scan:
    • For each point:
      – For every other scan
        » Find closest point
  – Minimize error w.r.t. transforms of all scans

• Disadvantage:
  – Solve \((6n) \times (6n)\) matrix equation, where \(n\) is number of scans
Approach #3a: Slightly Less Brute-Force

• While not converged:
  – For each scan:
    • For each point:
      – For every other scan
        » Find closest point
    • Minimize error w.r.t. transform of this scan

• Faster than previous method (matrices are $6 \times 6$) [Bergevin 96, Benjemaa 97]
Graph Methods

- Speedup previous by creating a graph of pairwise alignments between overlapping scans

- Find transformations consistent as possible with all pairwise ICP
Surface Reconstruction

Next time:

physical model → Scanning devices → Registration → Implicit reconstruction → acquired point cloud → reconstructed model
References

• “The 3D Model Acquisition Pipeline”, Bernardini et al, ‘02
• “A method for registration of 3-D shapes”, Besl et al., ’92 ← ICP paper
• “Surface reconstruction from unorganized points”, Hoppe et al., ’92
• “Closed-form solution of absolute orientation using orthonormal matrices”, Horn et al., ‘88
• “A general solution of the orthogonal Procrustes problem”, Schönemann ’66