Lecture 25: Structure from motion
Announcements

• HW 2 out.
  – New version from yesterday with some clarifications on coordinate systems

• PA 4 tonight
Structure from motion

- Given many images, how can we
  a) figure out where they were all taken from?
  b) build a 3D model of the scene?

This is (roughly) the **structure from motion** problem
Structure from motion

- **Input**: images with points in correspondence $p_{i,j} = (u_{i,j}, v_{i,j})$

- **Output**
  - structure: 3D location $\mathbf{x}_i$ for each point $p_i$
  - motion: camera parameters $\mathbf{R}_j$, $\mathbf{t}_j$ possibly $\mathbf{K}_j$

- **Objective function**: minimize *reprojection error*
What we’ve seen so far...

• 2D transformations between images
  – Translations, affine transformations, homographies...

• Fundamental matrices
  – Still represent relationships between 2D images

• What’s new: Explicitly representing 3D geometry of cameras and points
Camera calibration and triangulation

• Suppose we know 3D points
  – And have matches between these points and an image
  – How can we compute the camera parameters?

• Suppose we have know camera parameters, each of which observes a point
  – How can we compute the 3D location of that point?
Structure from motion

• SfM solves both of these problems *at once*
• A kind of chicken-and-egg problem
  – (but solvable)
Also doable from video
\[ p_{1,1}, t_{1} \]

\[ p_{1,2}, t_{2} \]

\[ p_{1,3}, t_{3} \]
\[ \Pi_1 \mathbf{X}_1 \sim p_{11} \]

Structure from motion

minimize \( g(R, T, X) \)

non-linear least squares
Photo Tourism

Photo Tourism
Exploring photo collections in 3D

Noah Snavely  Steven M. Seitz  Richard Szeliski
University of Washington  Microsoft Research

SIGGRAPH 2006
Input

Feature detection

Feature matching
First step: how to get correspondence?

• Feature detection and matching
Feature detection

Detect features using SIFT [Lowe, IJCV 2004]
Feature detection

Detect features using SIFT [Lowe, IJCV 2004]
Feature matching

Match features between each pair of images
Feature matching

Refine matching using RANSAC to estimate fundamental matrix between each pair.
Correspondence estimation

• Link up pairwise matches to form connected components of matches across several images: tracks

Image 1  Image 2  Image 3  Image 4
Geometric inference based on tracks

Structure from motion

- Minimize sum of squared reprojection errors:
  \[
g(X, R, T) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} \cdot \left\| P(x_i, R_j, t_j) - \begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix} \right\|^2
  \]
  is point \( i \) visible in image \( j \) ?

- Minimizing this function is called \textit{bundle adjustment}
  - Optimized using non-linear least squares, e.g. Levenberg-Marquardt
Problem size

Trevi Fountain collection

466 input photos
+ > 100,000 3D points
= very large optimization problem
Image connectivity graph

**Fig. 1** Photo connectivity graph. This graph contains a node for each image in a set of photos of the Trevi Fountain, with an edge between each pair of photos with matching features. The size of a node is proportional to its degree. There are two dominant clusters corresponding to day (a) and night time (d) photos. Similar views of the facade cluster together in the center, while nodes in the periphery, e.g., (b) and (c), are more unusual (often close-up) views.

Is SfM always uniquely solvable?

• No...
Structure from motion ambiguity

• If we scale the entire scene by some factor $k$ and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same:

$$x = PX = \left(\frac{1}{k} P\right)(kX)$$

It is impossible to recover the absolute scale of the scene!
Structure from motion ambiguity

• More generally: if we transform the scene using a transformation \( Q \) and apply the inverse transformation to the camera matrices, then the images do not change

\[
x = PX = \left( PQ^{-1} \right) (QX)
\]
Types of ambiguity

<table>
<thead>
<tr>
<th>Type</th>
<th>Transformation Matrix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projective</td>
<td>$\begin{bmatrix} A &amp; t \ v^T &amp; v \end{bmatrix}$</td>
<td>Preserves intersection and tangency</td>
</tr>
<tr>
<td>15dof</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Affine</td>
<td>$\begin{bmatrix} A &amp; t \ 0^T &amp; 1 \end{bmatrix}$</td>
<td>Preserves parallelism, volume ratios</td>
</tr>
<tr>
<td>12dof</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Similarity</td>
<td>$\begin{bmatrix} sR &amp; t \ 0^T &amp; 1 \end{bmatrix}$</td>
<td>Preserves angles, ratios of length</td>
</tr>
<tr>
<td>7dof</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Euclidean</td>
<td>$\begin{bmatrix} R &amp; t \ 0^T &amp; 1 \end{bmatrix}$</td>
<td>Preserves angles, lengths</td>
</tr>
<tr>
<td>6dof</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Structure from Motion

• Repetitive structures
SfM applications

- 3D modeling
- Surveying
- Robot navigation and mapmaking
- Archeological reconstruction
- Visual effects
Google Street View
Visual Turing Test
Figure 5. Visual Turing test. In each image pair, the ground truth image is on the left and our result is on the right.
Challenges – Indoor Reconstruction

- Texture-poor surfaces
- Complicated visibility
- Prevalence of thin structures (doors, walls, tables)
Museums
Google Art
State-of-the-art

• sFM used for large-scale internet level 3D reconstruction

• Future: expect 3D imagery
  – Kinect, sFM, etc.
Where are we?

• First: low-level vision, features

• Second: 3D reconstruction

• Next: Recognition

• Last few lectures: Computational photography