CS6670: Computer Vision

Noah Snavely

Lecture 22: Structure from motion

Readings

- Szeliski, Chapter 7.1 – 7.4

Road map

- What we've seen so far:
  - Low-level image processing: filtering, edge detecting, feature detection
  - Geometry: image transformations, panoramas, single-view modeling Fundamental matrices

- What's next:
  - Finishing up geometry
  - Recognition
  - Image formation

Back to image filter

- Here's an image with many high f

Large-scale structure from motion

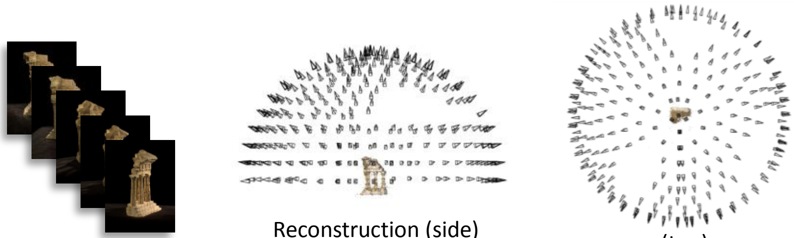
Dubrovnik, Croatia. 4,619 images (out of an initial 57,845).
Total reconstruction time: 23 hours
Number of cores: 352

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



Reconstruction (side)

(top)

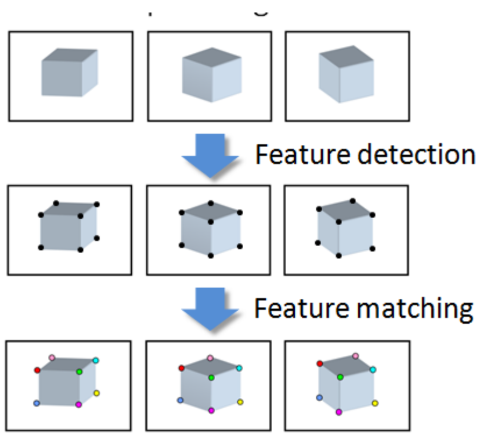
- Input: images with points in correspondence
  $p_{i,j} = (u_{i,j}, v_{i,j})$
- Output
  - structure: 3D location $\mathbf{x}_i$ for each point p_i
  - motion: camera parameters $\mathbf{R}_j$, $\mathbf{t}_j$ possibly $\mathbf{K}_j$
- Objective function: minimize *reprojection error*

Also doable from video

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*

Input



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)

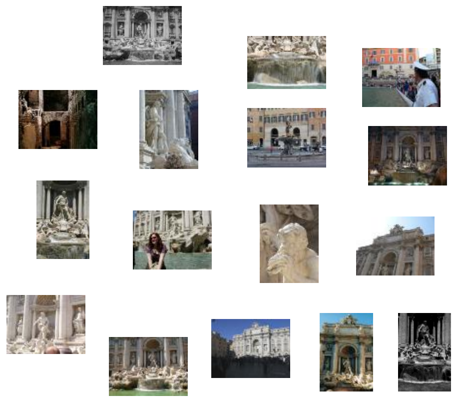
Photo Tourism

First step: how to get correspondence?

- Feature detection and matching

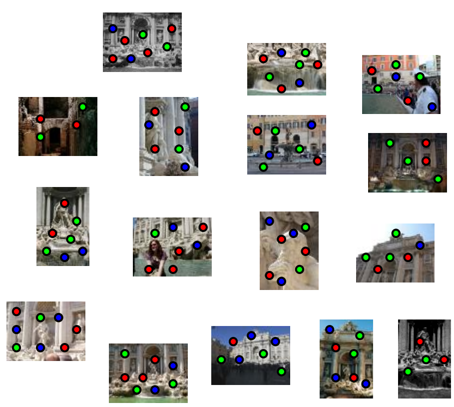
Feature detection

Detect features using SIFT [Lowe, IJCV 2004]



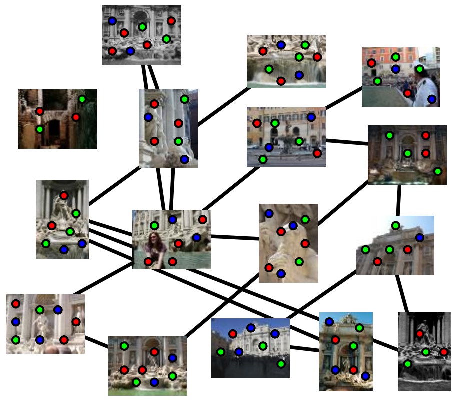
Feature detection

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

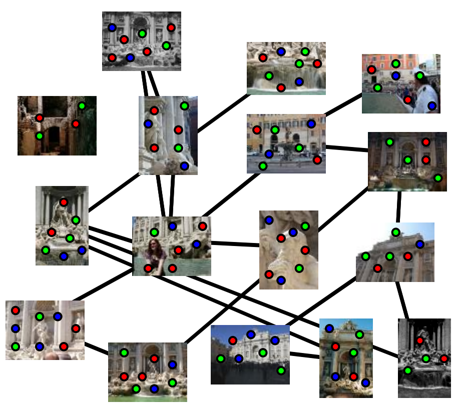
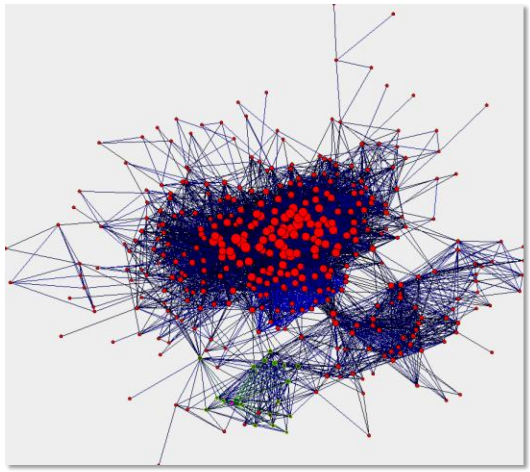
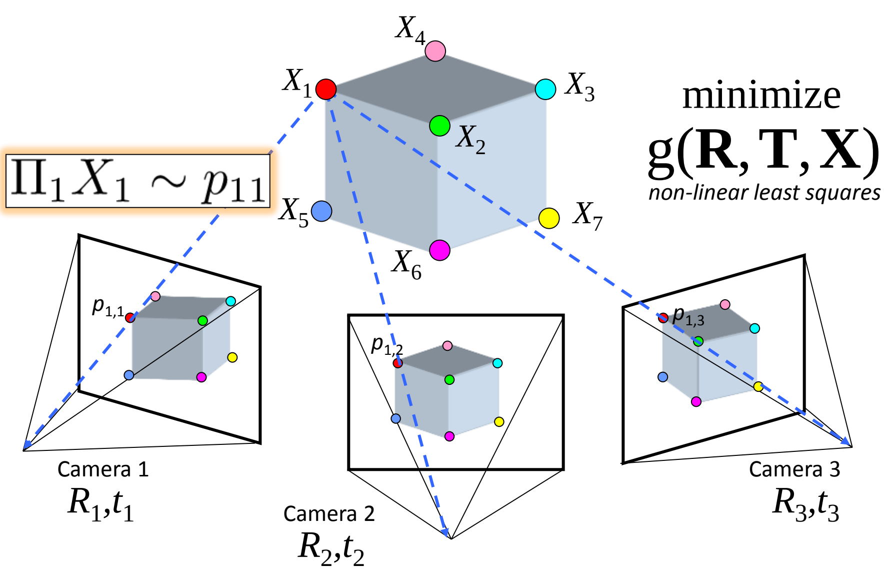


Image connectivity graph



(graph layout produced using the Graphviz toolkit: http://www.graphviz.org/)

Structure from motion



Problem size

- What are the variables?
- How many variables per camera?
- How many variables per point?

- Trevi Fountain collection

466 input photos

+ > 100,000 3D points

= very large optimization problem

Structure from motion

- Minimize sum of squared reprojection errors:

$$g(\mathbf{X}, \mathbf{R}, \mathbf{T}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \underbrace{w_{ij}}_{\text{indicator variable: is point } i \text{ visible in image } j\text{?}} \cdot \left\| \underbrace{\mathbf{P}(\mathbf{x}_i, \mathbf{R}_j, \mathbf{t}_j)}_{\text{predicted image location}} - \underbrace{\begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix}}_{\text{observed image location}} \right\|^2$$

- Minimizing this function is called *bundle adjustment*
  - Optimized using non-linear least squares, e.g. Levenberg-Marquardt

Is SfM always uniquely solvable?

Is SfM always uniquely solvable?

- No…