Automatic Panoramic Image Stitching

Dr. Matthew Brown, University of Bath
Automatic 2D Stitching

• The old days: panoramic stitchers were limited to a 1-D sweep

• 2005: 2-D stitchers use object recognition for discovery of overlaps
AutoStitch iPhone

“Create gorgeous panoramic photos on your iPhone”
- Cult of Mac

“Raises the bar on iPhone panoramas”
- TUAW

“Magically combines the resulting shots”
- New York Times

Available on the iPhone App Store
Case study – Image mosaicing

Any two images of a general scene with the same camera centre are related by a planar projective transformation given by:

\[
\tilde{w}' = KRK^{-1}\tilde{w}
\]

where \( K \) represents the camera calibration matrix and \( R \) is the rotation between the views.

This projective transformation is also known as the homography induced by the plane at infinity. A minimum of four image correspondences can be used to estimate the homography and to warp the images onto a common image plane. This is known as mosaicing.
Meanwhile in 1999...

- David Lowe publishes “Scale Invariant Feature Transform”
- 11,572 citations on Google scholar
- A breakthrough solution to the correspondence problem
- SIFT is capable of operating over much wider baselines than previous methods

[ Lowe ICCV 1999 ]
Local Feature Matching

- Given a point in the world...

...compute a description of that point that can be easily found in other images
Scale Invariant Feature Transform

- Start by detecting points of interest (blobs)

\[ L(I(x)) = \nabla \cdot \nabla I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \]

- Find maxima of image Laplacian over scale and space
Scale Invariant Feature Transform

- Describe local region by distribution (over angle) of gradients

- Each descriptor: 4 x 4 grid x 8 orientations = 128 dimensions
Scale Invariant Feature Transform

- Extract SIFT features from an image

- Each image might generate 100’s or 1000’s of SIFT descriptors
Feature Matching

• Goal: Find all correspondences between a pair of images

• Extract and match all SIFT descriptors from both images
Feature Matching

- Each SIFT feature is represented by 128 numbers.
- Feature matching becomes task of finding a nearby 128-d vector.
- All nearest neighbours:

  \[ \forall j \quad \text{NN}(j) = \arg \min_i \|x_i - x_j\|, \ i \neq j \]

- Solving this exactly is \(O(n^2)\), but good approximate algorithms exist.
- e.g., [Beis, Lowe ’97] Best-bin first k-d tree.
- Construct a binary tree in 128-d, splitting on the coordinate dimensions.
- Find approximate nearest neighbours by successively exploring nearby branches of the tree.
2-view Rotational Geometry

• Feature matching returns a set of noisy correspondences
• To get further, we will have to understand something about the geometry of the setup
2-view Rotational Geometry

• Recall the projection equation for a pinhole camera

\[ \begin{bmatrix} \tilde{u} \end{bmatrix} = \begin{bmatrix} K \end{bmatrix} \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} \tilde{X} \end{bmatrix} \]

\( \tilde{u} \sim [u, v, 1]^T \) : Homogeneous image position

\( \tilde{X} \sim [X, Y, Z, 1]^T \) : Homogeneous world coordinates

\( K \ (3 \times 3) \) : Intrinsic (calibration) matrix

\( R \ (3 \times 3) \) : Rotation matrix

\( t \ (3 \times 1) \) : Translation vector
2-view Rotational Geometry

- Consider two cameras at the same position (translation)
- WLOG we can put the origin of coordinates there

\[ \tilde{u}_1 = K_1 [ R_1 \mid t_1 ] \tilde{X} \]

- Set translation to 0

\[ \tilde{u}_1 = K_1 [ R_1 \mid 0 ] \tilde{X} \]

- Remember \( \tilde{X} \sim [X, Y, Z, 1]^T \) so

\[ \tilde{u}_1 = K_1 R_1 X \]

(\text{where} \( X = [X, Y, Z]^T \))
2-view Rotational Geometry

- Add a second camera (same translation but different rotation and intrinsic matrix)

\[ \tilde{u}_1 = K_1 R_1 X \]
\[ \tilde{u}_2 = K_2 R_2 X \]

- Now eliminate \( X \)

\[ X = R_1^T K_1^{-1} \tilde{u}_1 \]

- Substitute in equation 1

\[ \tilde{u}_2 = K_2 R_2 R_1^T K_1^{-1} \tilde{u}_1 \]

This is a 3x3 matrix -- a (special form) of homography
Computing H: Quiz

\[
s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}
\]

- Each correspondence between 2 images generates ___ equations
- A homography has _____ degrees of freedom
- _____ point correspondences are needed to compute the homography
- Rearranging to make H the subject leads to an equation of the form

\[
Mh = 0
\]

- This can be solved by _____
Finding Consistent Matches

- Raw SIFT correspondences (contains outliers)
Finding Consistent Matches

- SIFT matches consistent with a rotational homography
Finding Consistent Matches

- Warp images to common coordinate frame
RANSAC

- **RAndom SAmple Consensus** [Fischler-Bolles ‘81]
- Allows us to robustly estimate the best fitting homography despite noisy correspondences
- **Basic principle:** select the smallest random subset that can be used to compute H
- Calculate the support for this hypothesis, by counting the number of *inliers* to the transformation
- Repeat sampling, choosing H that maximises # inliers
RANSAC

H = eye(3,3); nBest = 0;

for (int i = 0; i < nIterations; i++)
{
    P4 = SelectRandomSubset(P);
    Hi = ComputeHomography(P4);
    nInliers = ComputeInliers(Hi);
    if (nInliers > nBest)
    {
        H = Hi;
        nBest = nInliers;
    }
}
Recognising Panoramas

[ Brown, Lowe ICCV’03 ]
Global Alignment

- The pairwise image relationships are given by **homographies**
- But over time multiple pairwise mappings will accumulate errors
- Notice: gap in panorama before it is closed...
Gap Closing
Bundle Adjustment
Bundle Adjustment

• Minimise sum of robustified residuals

\[ \sum_{i=1}^{n_p} \sum_{j \in \mathcal{V}(i)} f(u_{ij}(\Theta) - m_{ij}) \]

- \( u_{ij} \) = projected position of point \( i \) in image \( j \)
- \( m_{ij} \) = measured position of point \( i \) in image \( j \)
- \( \mathcal{V}(i) \) = set of images where point \( i \) is visible
- \( n_p \) = # points/tracks (mutual feature matches across images)
- \( \Theta \) = camera parameters

• Robust error function (Huber)

\[ f(x) = \begin{cases} 
|x|^2, & |x| < \sigma \\
2\sigma|x| - \sigma^2, & |x| \geq \sigma 
\end{cases} \]
Aerial Image Stitching

[ Images: SenseFly, Zufferey, Beleyer EPFL]
Endoscopy Imaging

[ E. Seibel, J. Wong (U. Washington) ]
Other topics

• **Blending/Gain compensation/HDR** -- how to combine multiple images to give a consistent, seamless result
• **Gravity direction** estimate -- how do we know which way is up?
• **Different projections** -- rectilinear, spherical, cylindrical, fisheye...

More details [http://cs.bath.ac.uk/brown](http://cs.bath.ac.uk/brown)
iPhone Computer Vision Apps

Also good: Kooaba Déjà vu, Photosynth, Word Lens
Aardman, Disney, Double Negative, EA, Frontier, BBC, Buro Happold, Smoke and Mirrors...

PhD and EngD positions starting September 2012

m.brown@bath.ac.uk