vSLAM: Visual Simultaneous Location and Mapping

Roland Angst

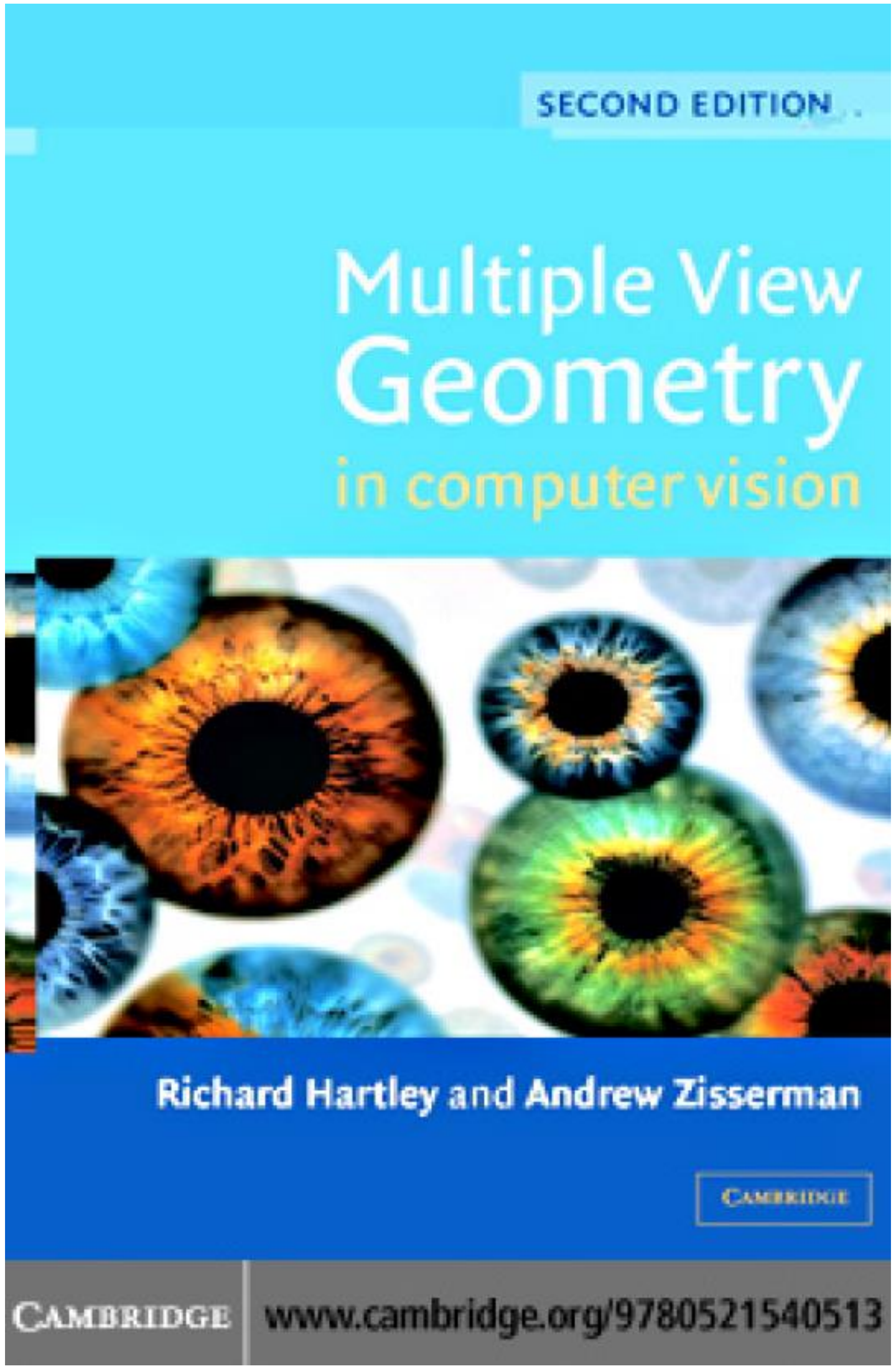
rangst@stanford.edu

www.stanford.edu/~rangst

Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

THE Reference for Most of this Lecture



Recap: Camera Matrix

- Pinhole camera model (calibrated vs. uncalibrated case)

$$\mathbf{P} = \mathbf{K}\left[\mathbf{R}, \mathbf{t}\right] \qquad \mathbf{K} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

  - Principal point **p**
    - Intersection of principal axis with image plane
    - Principal axis: line through camera centre orthogonal to image plane
  - Camera center (aka. centre of projection, pinhole) in world coordinates?

$$\mathbf{C} = \begin{pmatrix} -\mathbf{R}^T\mathbf{t} \\ 1 \end{pmatrix}$$

- Affine camera model
  - Camera center at infinity
  - Parallel projection

Recap: Two-View Geometry

- Two-view geometry
  - Fundamental matrix $\mathbf{x}'^T \mathbf{F} \mathbf{x} = 0$
    - Degree of freedoms?
  - Essential matrix $\tilde{\mathbf{x}}'^T \mathbf{E} \tilde{\mathbf{x}} = 0$ with $\mathbf{E} = \mathbf{K}^T \mathbf{F} \mathbf{K} = [\mathbf{t}]_\times \mathbf{R}$
    - Degree of freedoms?
    - 5-point algorithm
      - Needs to solve a 10^{th} degree univariate polynomial
      - Provides 10 solutions (counting multiplicities; some of them complex)
      - See also "Five-Point Motion Estimation Made Easy" by Hongdong Li and Richard Hartley
- Estimation of fundamental or essential
  - Find keypoints and extract feature descriptors
  - Putative correspondences by matching feature descriptors
  - RANSAC loop for geometric verification

Example: Pre & Post RANSAC

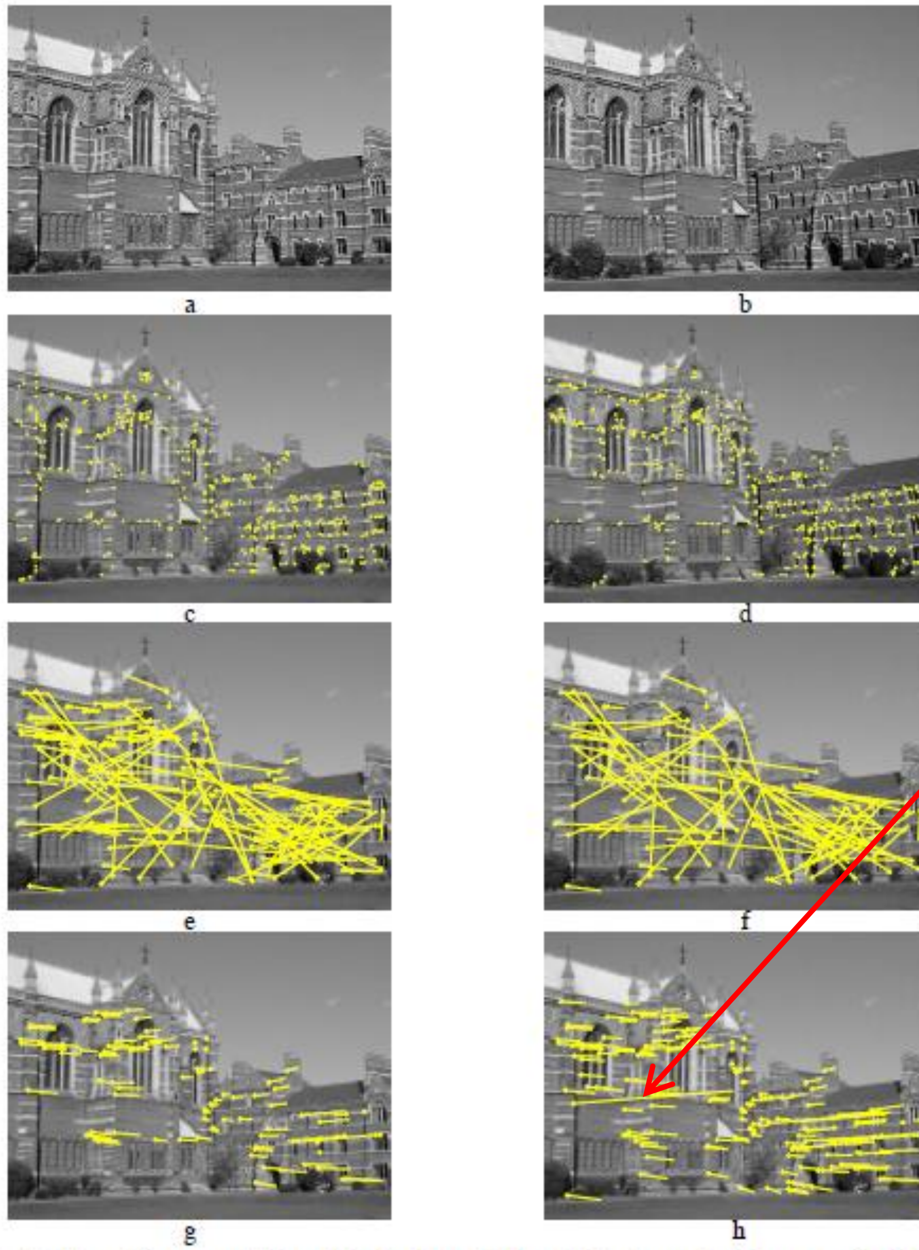


Fig. 11.4. Automatic computation of the fundamental matrix between two images using RANSAC. *(a) (b) left and right images of Keble College, Oxford. The motion between views is a translation and rotation. The images are 640×480 pixels. (c) (d) detected corners superimposed on the images. There are approximately 500 corners on each image. The following results are superimposed on the left image: (e) 188 putative matches shown by the line linking corners, note the clear mismatches; (f) outliers – 89 of the putative matches. (g) inliers – 99 correspondences consistent with the estimated* F*; (h) final set of 157 correspondences after guided matching and MLE. There are still a few mismatches evident, e.g. the long line on the left.*

Figure from: "Multiple View Geometry"
Hartley & Zisserman

Two-view geometric verification is not perfect

Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Decomposing the Essential Matrix

- Assume the correct essential has been found
- Goal: decompose essential into rotation and translation
- Problem: decomposition is not unique: 4 solutions exist
  - With: $\mathrm{svd}(\mathbf{E}) = \mathbf{U}\,\mathrm{diag}(1,1,0)\mathbf{V}^T$

$$[\mathbf{t}]_\times = \pm\mathbf{U}\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{U}^T \qquad \mathbf{R} = \mathbf{U}\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}\mathbf{V}^T \text{ or } \mathbf{R} = \mathbf{U}\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}\mathbf{V}^T$$

  - See also MVG sec. 9.6.2 (2^{nd} edition)

Decomposing the Essential Matrix

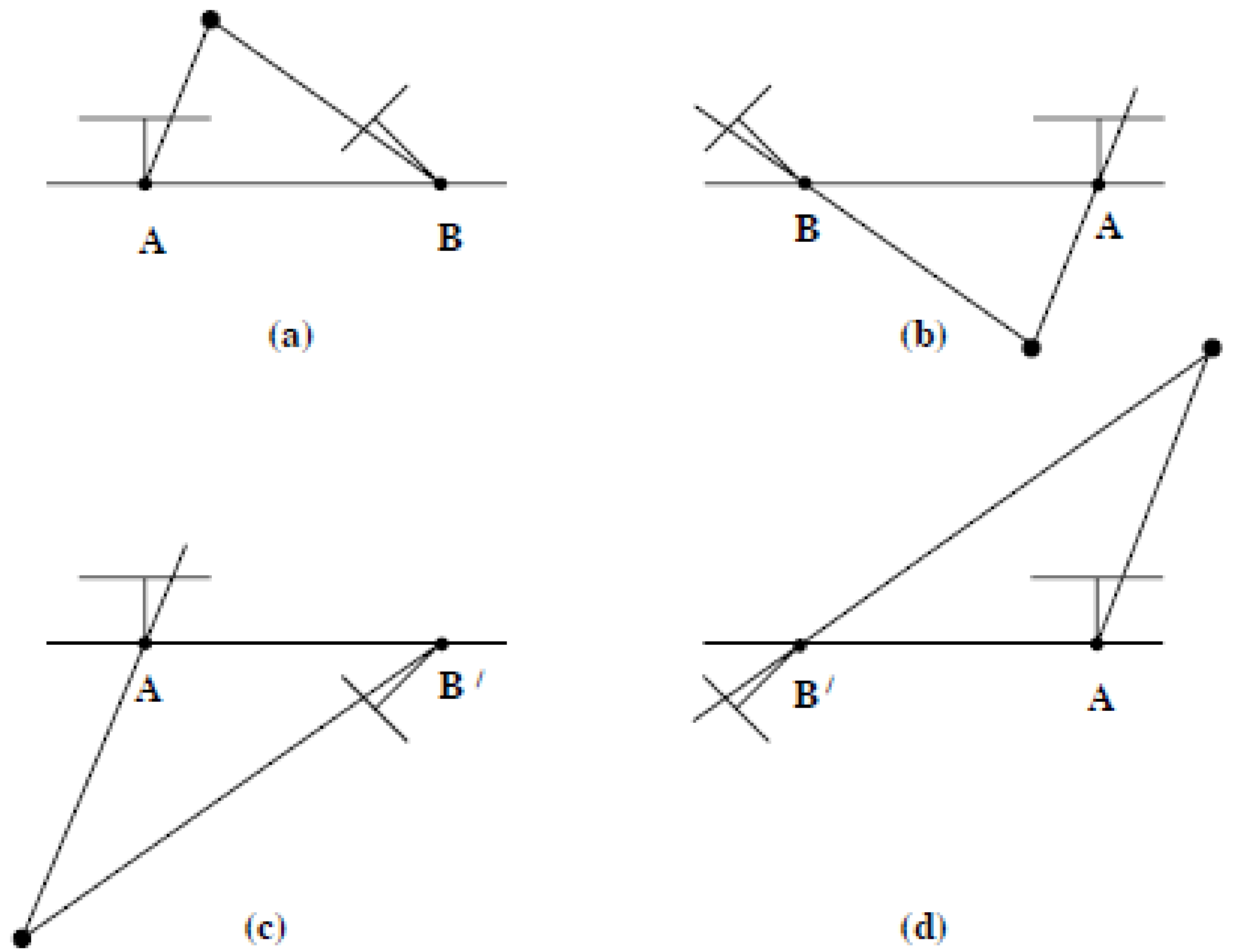


Figure from: "Multiple View Geometry"
Hartley & Zisserman

- Interpretation
  - baseline reversal
  - Rotation of one camera 180° about baseline
- Points are in front of camera only in one solution

Two-View Geometry in Practice

- Be aware of planar scenes
  - Homography explains point correspondences $\mathbf{x}' = \mathbf{H}\mathbf{x}$
  - Let's pick a **random** point $\mathbf{y}'$ in 2nd view
  - Consider line spanned by $\mathbf{x}'$ and $\mathbf{y}'$: $\mathbf{l}' = [\mathbf{y}']_\times \mathbf{x}' = [\mathbf{y}']_\times \mathbf{H}\mathbf{x}$
  - Obviously, $\mathbf{x}'$ lies on this line: $0 = \mathbf{x}'^T \mathbf{l}' = \mathbf{x}'^T \underbrace{[\mathbf{y}']_\times \mathbf{H}}_{=\mathbf{F}} \mathbf{x}$
- Less of a problem for essential matrices
  - But you'll get two equally valid but different solutions…

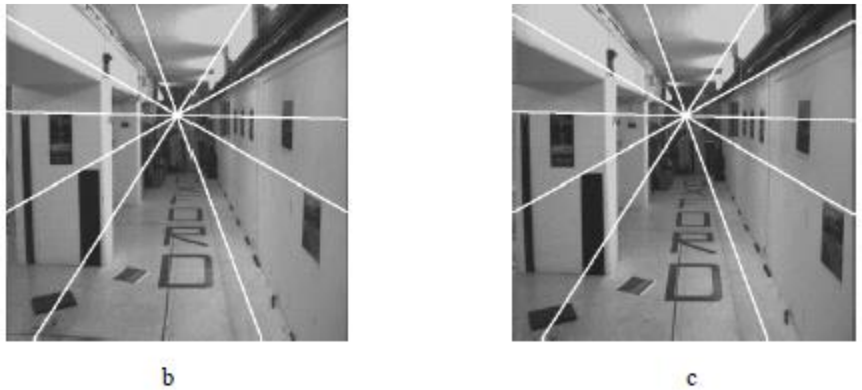


Figure from: "Multiple View Geometry" Hartley & Zisserman

- Ill-conditioned motions
  - Pure rotations: reveals no 3D structure
  - Forward motions
  - Rotation-translation ambiguity: translation vs. rotation around an axis far away
    - Severe problem for nearly planar scenes with small depth variation
    - Especially important for narrow field of view (like on mobile phones)

Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - **Triangulation**
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Triangulation

- Assume known
  - Camera poses (for at least two frames)
  - Image correspondences
- Shoot rays through image points and intersect in 3D
  - Rays won't intersect due to image noise
  - Minimizing meaningful reprojection error is non-trivial
  - Example: 2-view triangulation

$$\min_{\hat{\mathbf{x}},\hat{\mathbf{x}}'} \left\| \frac{1}{x_3}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \frac{1}{\hat{x}_3}\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} \right\|_2^2 + \left\| \frac{1}{x'_3}\begin{pmatrix} x'_1 \\ x'_2 \end{pmatrix} - \frac{1}{\hat{x}'_3}\begin{pmatrix} \hat{x}'_1 \\ \hat{x}'_2 \end{pmatrix} \right\|_2^2$$
$$\text{s.t.}\ \ \hat{\mathbf{x}}'^T \mathbf{F} \hat{\mathbf{x}} = 0$$

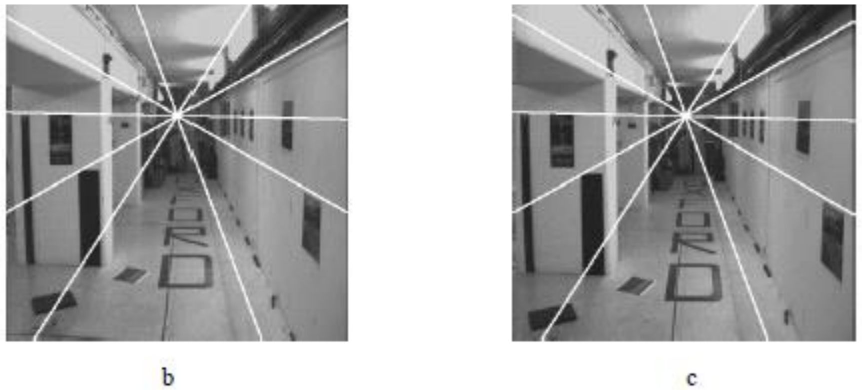


Figure from: "Multiple View Geometry" Hartley & Zisserman

    - Leads to roots of 6^{th} degree univariate polynomial
    - Quiz: Are there 3D points which can't be triangulated from two views?
      - Yes: points on baseline (ie. Points which project onto epipoles)

Triangulation

- Assume known
  - Camera poses (for at least two frames)
  - Image correspondences
- Direct Linear Transform (DLT)
  - Simple method, minimizes algebraic error
  - Eliminate scale factor (= projective depth)

$$\mathbf{x}_f \cong \mathbf{P}_f \mathbf{X} \Leftrightarrow \lambda_f \mathbf{x}_f = \mathbf{P}_f \mathbf{X}$$

$$[\mathbf{x}_f]_\times \mathbf{x}_f = \mathbf{0}_{3\times 1} = [\mathbf{x}_f]_\times \mathbf{P}_f \mathbf{X}$$

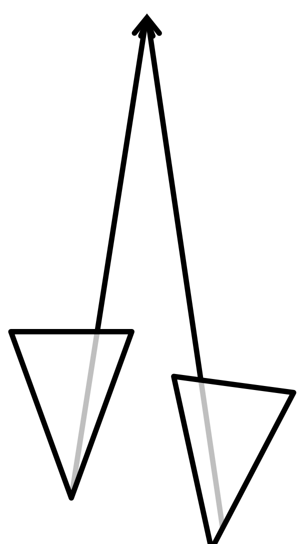
  - Stack measurements from all images and solve with SVD

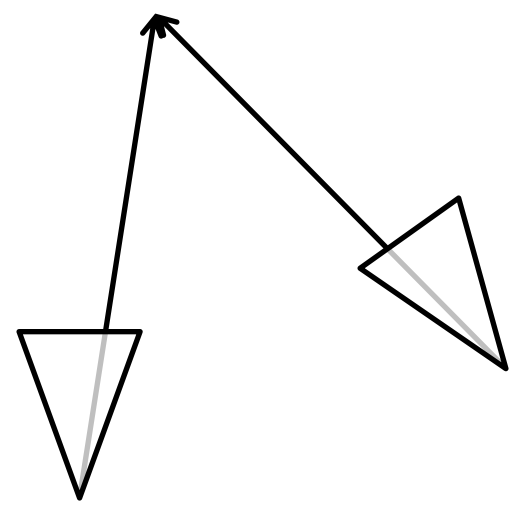
$$\min_{\mathbf{X}\in\mathbb{R}^4} \| [\Downarrow_f \ [\mathbf{x}_f]_\times \mathbf{P}_f] \mathbf{X} \|_2^2$$

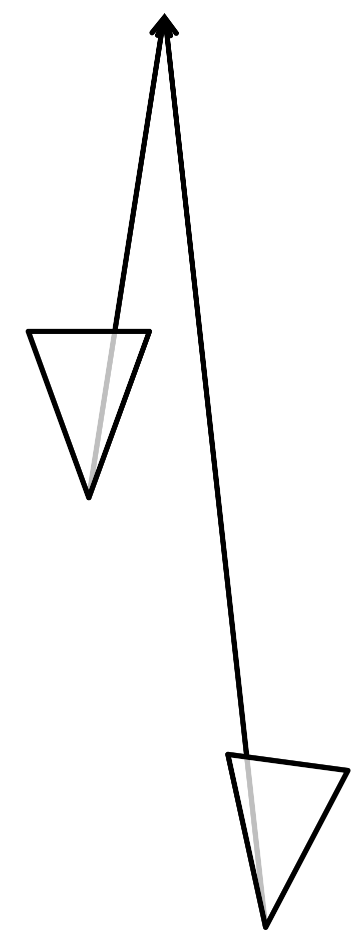
  - What about points at infinity?

Triangulation

- Depth uncertainty of triangulated 3D points mostly depends on angle between intersected rays
  - Small angle → inaccurate triangulation







- Small baseline → small angle
- Large baseline → large angle?
  - Not always true! Example?
  - Forward motions…

Goals of this Lecture

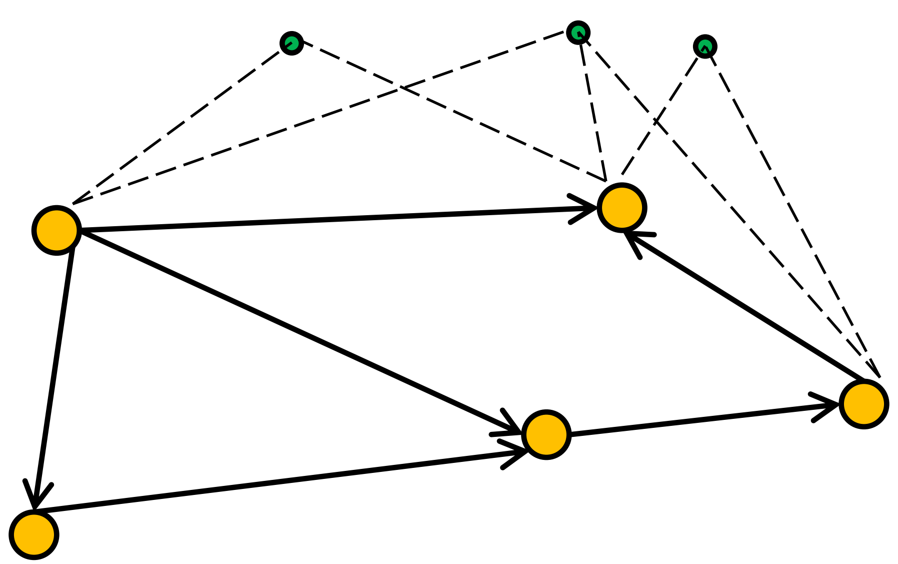
- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - **Bundle adjustment**
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Bundle Adjustment

- We know how to perform 2-view reconstruction
- Assume we have initial guess of 3D reconstruction
- Goal: refine a meaningful geometric error
  - Reprojection error
  - Cycle consistency when camera sees same points again after making a loop

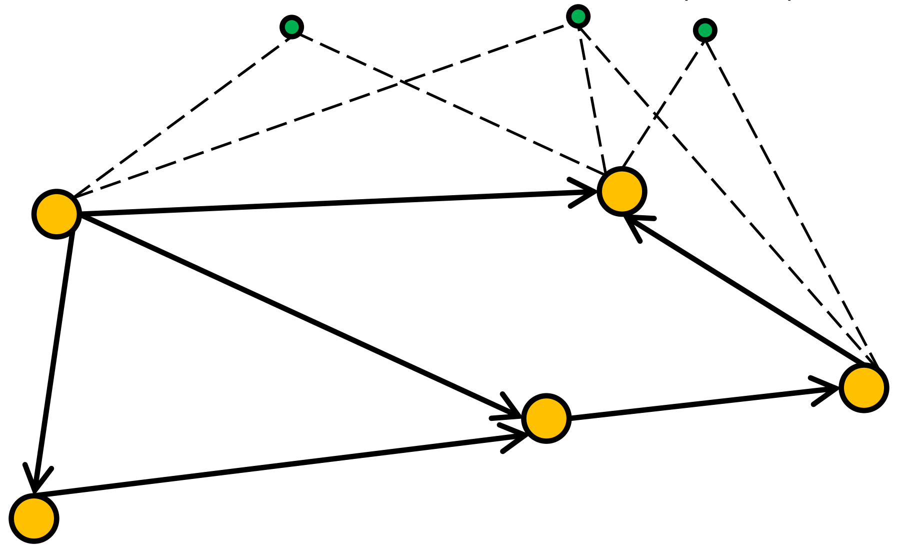
Point-Pose-Graph

- Conceptual representation of SfM
  - Vertices: camera poses & 3D points
  - Edges
    - Edges between camera vertices if estimate of **relative pose** is available (eg. from essential matrix)
    - Edges between camera and 3D point if point has been seen in this camera (= measurements)



Bundle Adjustment

- Unknowns
  - 3D coordinates of points & camera poses
- Data evidence
  - 2D feature point correspondences
- Initial guess available
  - Decompose pairwise essentials + three view verified
- Refine initial guess by minimizing reprojection error while adhering to cycle constraints
  - Modern BA frameworks phrase optimization problem as optimization over point-pose graph
  - "g2o: A General Framework for Graph Optimization" Kümmerle et.al. [ICRA11]



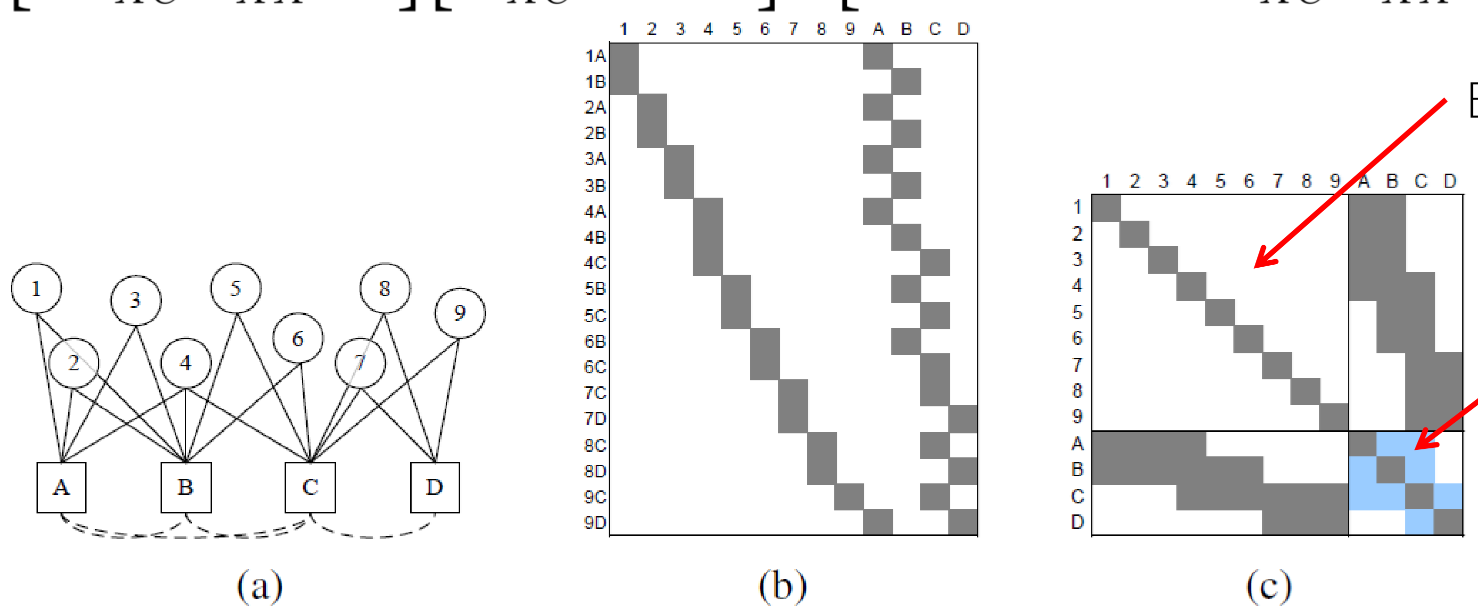
Bundle Adjustment: Parameterization of Unknowns

- Rotation matrices
  - Euler angles (avoid them if possible)
  - Unit-quaternions
  - Angle-axis & exponential-map
- 3D points (aka. Landmarks in robotics community)
  - Inhomogeneous coordinates (x,y,z)
    - Problem: points at infinity (or 'sufficiently' far away)
  - Homogeneous coordinates (x,y,z,w)
    - Problem: arbitrary scale per point leads to rank deficiency in Hessian
  - Inverse depth parameterization of point relative to a camera (eg. the one which has observed the point first)
    - No problems with points at infinity
    - Reprojection error becomes 'more linear' → Important for filtering based SLAM systems
    - "Inverse Depth Parameterization for Monocular SLAM" Civera, Davison, Montiel [Trans. On Robotics 08]

Bundle Adjustment: Numerical Details

- non-linear robust LS with residuals $\mathbf{r}_{ij} = \mathbf{x}_{ij} - f(\mathbf{X}_i, \mathbf{R}_j, \mathbf{K}_j)$
  - Linearize residual and compute update direction: $\Delta\mathbf{x}^t = \arg\min_{\Delta\mathbf{x}} \left\| \mathbf{r}^{t-1} + \mathbf{J}\Delta\mathbf{x} \right\|_2^2$
  - Gauss-Newton approximation of Hessian: $\mathbf{H} \approx \mathbf{J}^T\mathbf{J} = \begin{bmatrix} \mathbf{H}_{XX} & \mathbf{H}_{XC} \\ \mathbf{H}_{XC}^T & \mathbf{H}_{CC} \end{bmatrix}$
- Choose 'smart' parameterization for rotations & robust cost function (not L2)
- Computation of update direction: Gauss-Newton with Schur-complement trick

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{H}_{XC}^T\mathbf{H}_{XX}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{XX} & \mathbf{H}_{XC} \\ \mathbf{H}_{XC}^T & \mathbf{H}_{CC} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{XX} & \mathbf{H}_{XC} \\ \mathbf{0} & \mathbf{H}_{CC} - \mathbf{H}_{XC}^T\mathbf{H}_{XX}^{-1}\mathbf{H}_{XC} \end{bmatrix}$$



Block-diagonal: easy to invert

Fill-in after Schur-complement trick occurs whenever two cameras observe the same 3D point

Image from: http://szeliski.org/Book/

Figure 7.9 (a) Bipartite graph for a toy structure from motion problem and (b) its associated Jacobian $\boldsymbol{J}$ and (c) Hessian $\boldsymbol{A}$. Numbers indicate 3D points and letters indicate cameras. The dashed arcs and light blue squares indicate the fill-in that occurs when the structure (point) variables are eliminated.

Bundle Adjustment: Gauge Freedom

- Choice of global coordinate system is arbitrary
  - Often fixed to first camera $\mathbf{P}_1 = [\mathbf{I}_3, \mathbf{0}_{3\times 1}]$
    - 1st camera has no error
    - Introduces bias since error is not distributed evenly across all cameras
- Relative BA
  - Idea: Let's not select and designate a single global coordinate system
  - Instead: Choose multiple coordinate systems to express variables
    - Express 3D points relative to camera which first observed point
    - Relative transformations between coordinate systems allow to transform 3D points to other coordinate system
  - Pro
    - Error is more evenly spread
    - Loop closures can be handled better
  - Con
    - Jacobian matrices of BA become denser due to chaining relative transformations
  - "Relative Bundle Adjustment Based on Trifocal Constraints" Steffen, Frahm, Förstner [ECCV10]

Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

PnP Motivation: Sequential SfM, vSLAM

- SLAM: Simultaneous Location And Mapping
  - Terminology used in robotics
  - vSLAM: visual SLAM based entirely on images
    - Known as sequential SfM in computer vision
- Sequential SfM (aka. Incremental SfM)
  - Initialize structure and motion from two views
  - For each new image
    - Compute camera pose given 3D structure from previous iteration (PnP problem)
    - Refine camera poses (new & previous ones) and structure with BA
    - 'Densify' structure by triangulating new 3D points
- vSLAM = "sequential SfM in realtime" with video stream from camera

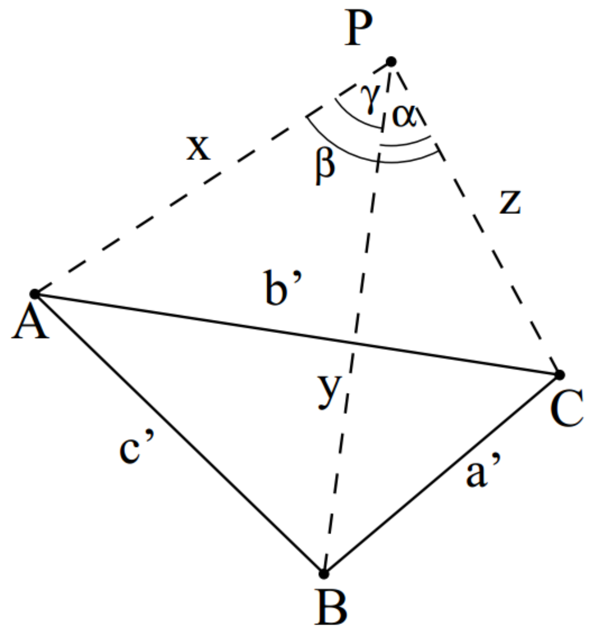
PnP Problem

- Perspective n-Point camera pose computation
  - Compute camera pose from n given 3D-2D point correspondences
  - Calibrated case: How many correspondences are minimally required?
    - 3 (be aware: up to four solutions)
    - P3P: “Review and Analysis of Solutions to the Three Point Perspective Pose Estimation Problem” Haralick et.al. [IJCV94]
    - OpenCV methods: `solvePNP(…)` and `solvePnPRansac(…)`
- Efficiency of PnP makes sequential SfM so attractive
  - RANSAC efficiency largely depends on minimal sample size!

P3P

- P3P again boils down to solving polynomial equations…

$$\begin{cases} Y^2 + Z^2 - YZp - a'^2 = 0 \\ Z^2 + X^2 - XZq - b'^2 = 0 \\ X^2 + Y^2 - XYr - c'^2 = 0. \end{cases}$$



$$p = 2\cos\alpha,\ q = 2\cos\beta,\ \ r = 2\cos\gamma.$$

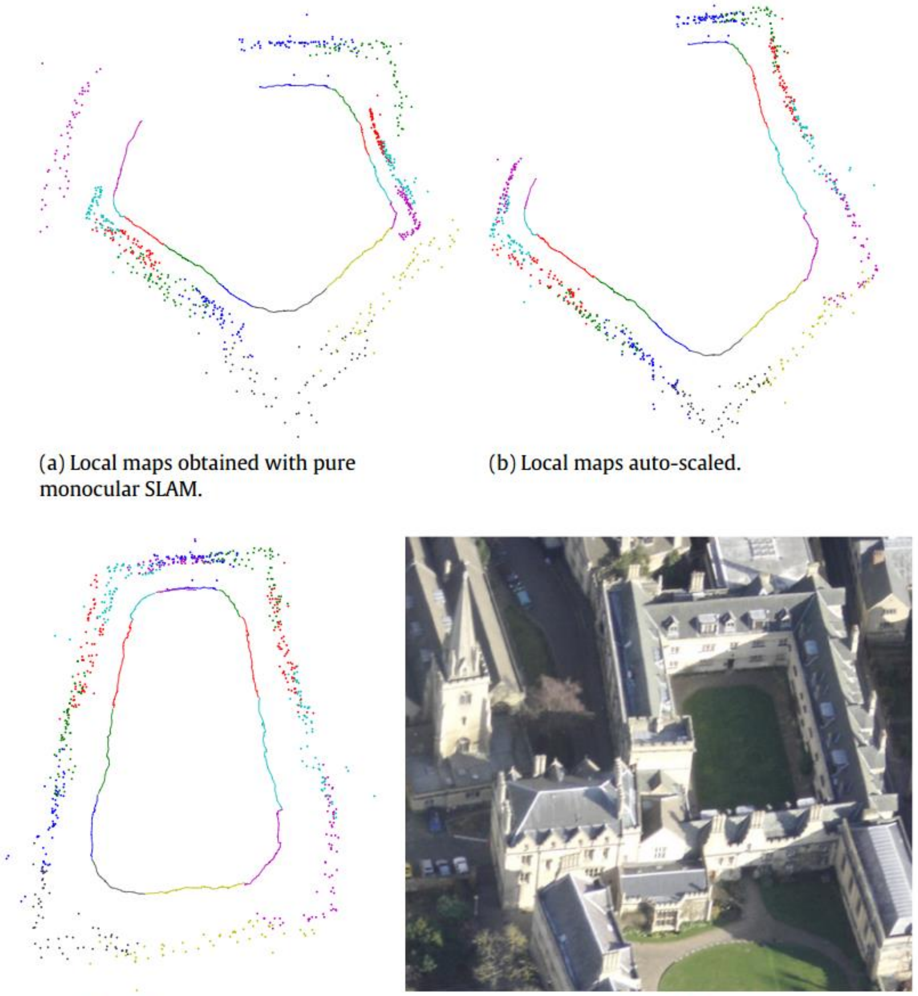
- Figure from: “Complete Solution Classification for the Perspective-Three-Point Problem” Gao et.al. [PAMI03]

Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Loop Closure and Scale Drift

- Loop closure problem
  - Accumulation of error in sequential SfM or SLAM leads to gaps in cycles
  - 3D structure might not overlap when closing a loop
  - Visual SLAM and sequential SfM especially suffer from scale drift
- Loop detection
  - Detect which parts should overlap
  - Leads to cycles in pose-graph
  - Cycles stabilize BA



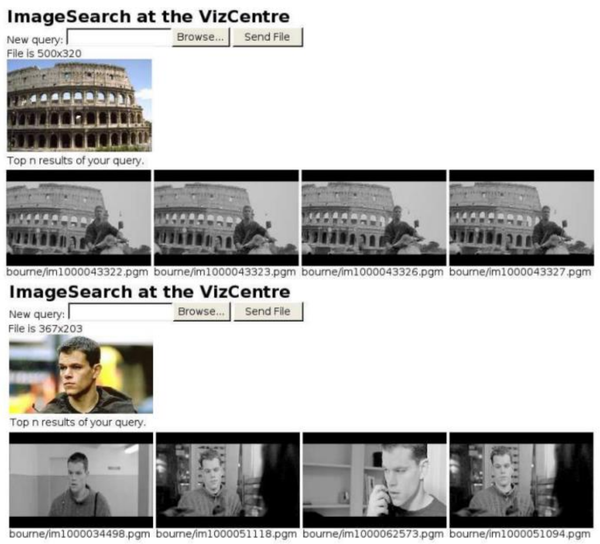
(a) Local maps obtained with pure monocular SLAM. (b) Local maps auto-scaled. (c) After loop closure. (d) Aerial view of the courtyard.

"A comparison of loop closing techniques in monocular SLAM"

Williams et.al. [RAS09]

Loop Detection

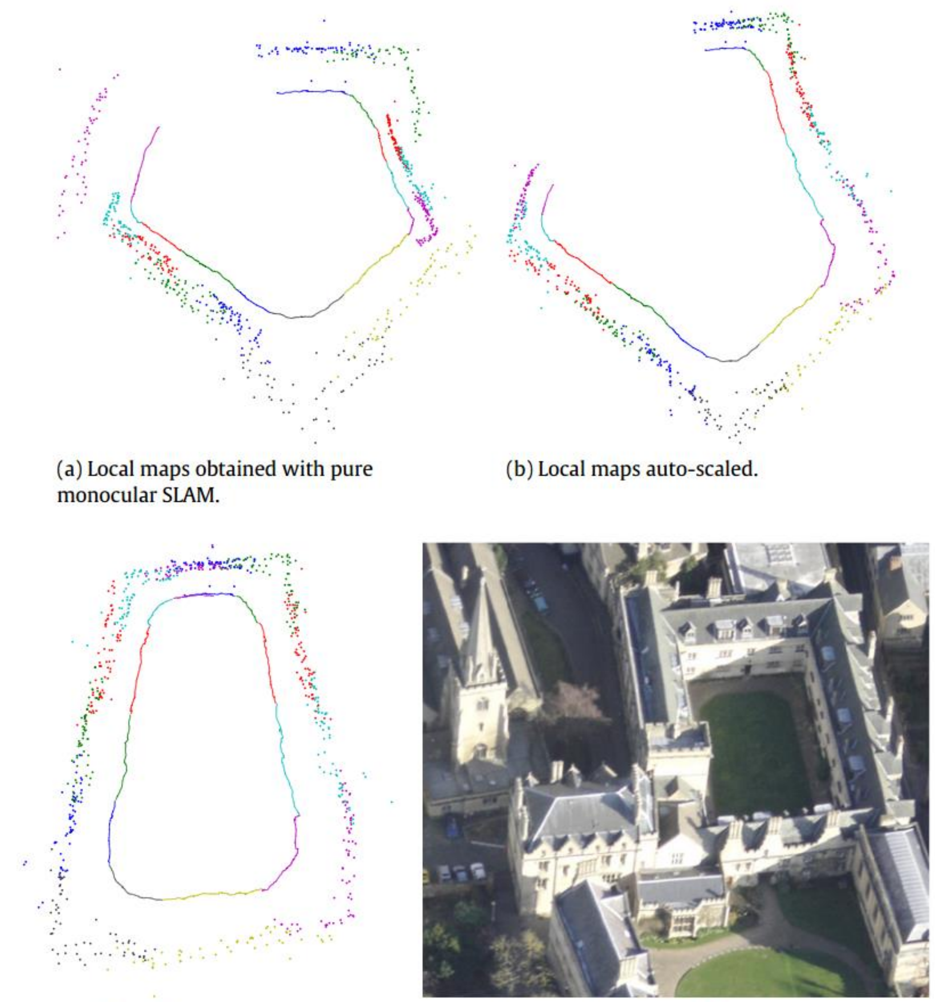
- Appearance based approaches most popular
  - Similar techniques used for image retrieval



"Scalable recognition with vocabulary tree"

Nister & Stewenius [CVPR06]

- Extract discriminative feature descriptors of keyframes
  - SIFT, SURF, etc.
- Store descriptors in efficient search data structure
  - Inverted index, vocabulary tree, …
- Issue a query with descriptors of query image and verify if any of top-K results is geometrically consistent



(a) Local maps obtained with pure monocular SLAM.

(b) Local maps auto-scaled.

(c) After loop closure.

(d) Aerial view of the courtyard.

"A comparison of loop closing techniques in monocular SLAM"

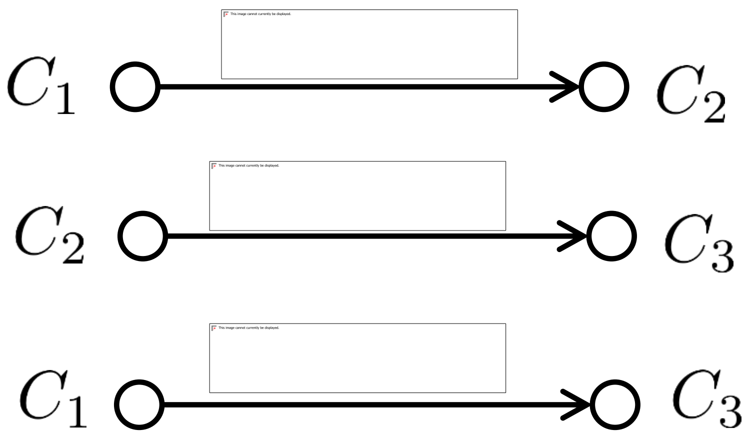
Williams et.al. [RAS09]

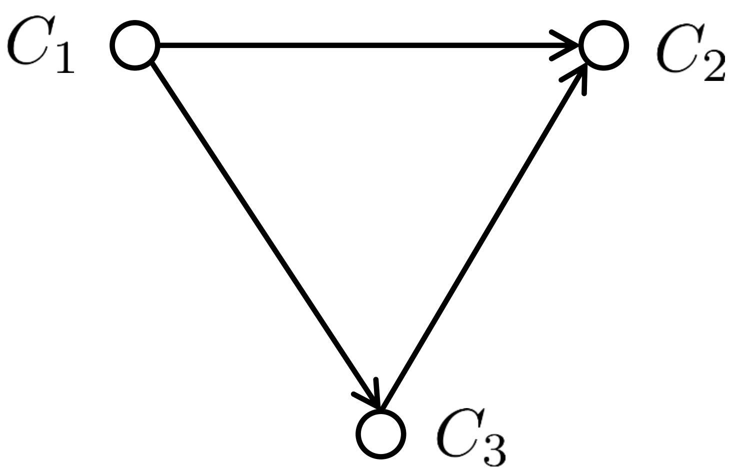
Goals of this Lecture

- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Hierarchical Structure-from-Motion (SfM)

- For each pair of images
  - perform 2-view reconstruction → set of two view reconstructions
- Triplet generation
  - Assemble pairwise reconstructions which share a common camera into triplets





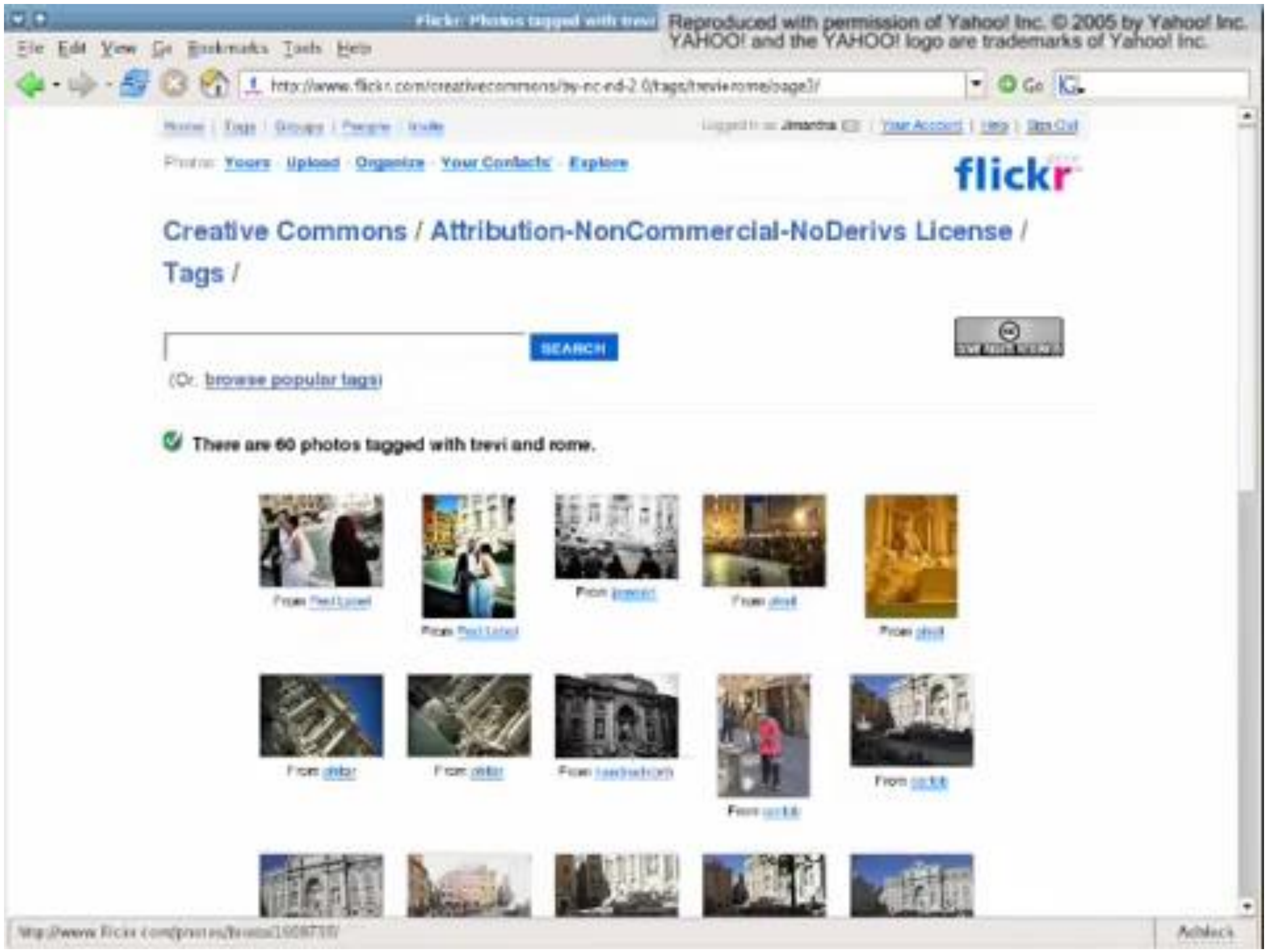
Hierarchical Structure-from-Motion (SfM)

- Increase robustness by three-view verification (loop consistency)
  - Cycle consistent relative poses
  - Remove spurious matches which survived two-view verification (eg. due to repetitive texture)
  - Slight complication: translations from pairwise reconstructions are only known up to scale
    - Choose arbitrary scale between first image pair, eg. $\|\mathbf{t}_{12}\|_2 = 1$
    - 3D points jointly seen in views 1,2, and 3 provide scale for $\mathbf{t}_{13}$ $\mathbf{t}_{23}$
- Register verified triplets (using shared edges)
  - Again pay attention to different scale in neighboring triplets
- Merge sub-reconstructions
  - Sprinkle BA steps in-between

Hierarchical Structure-from-Motion (SfM)

- Challenges
  - Generation of high-quality correspondences
  - Handling thousands of images: Avoid pairwise matching of images
  - Large scale optimization problem with many local minima
  - Repetitive structures
    - windows and building facades are highly repetitive…
  - …

Results



- Photo Tourism [2006]
  - http://phototour.cs.washington.edu/

Goals of this Lecture

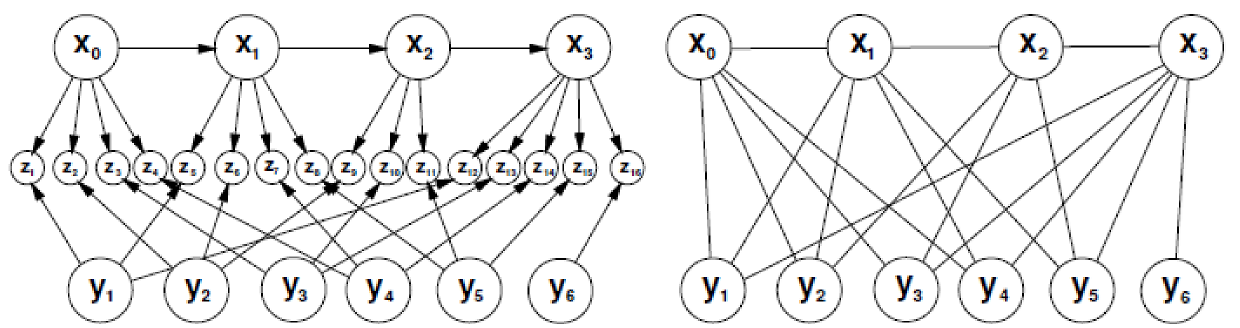
- Explaining building blocks for 3D reconstructions
  - Two-view geometry in more detail
  - Triangulation
  - Bundle adjustment
  - PnP
  - Loop closure with visual location recognition
- Putting all the pieces together
  - Hierarchical SfM
  - vSLAM

Visual SLAM

- Stream of temporally ordered images
- Simultaneously compute 3D map and camera pose w.r.t. map
- Two main approaches
  - Filtering
  - Key-frame based

Filtering vs. Key-Frames

- Recall SfM point-pose graph
  - Bipartite graph
    - 3D landmarks vs. camera poses
  - Filtering: marginalize over previous camera poses
    - State: 3D landmarks + current camera pose
  - Key-Frame BA: keep subset of frames as keyframes
    - State: 3D landmarks + camera poses for all key frames
    - Nowadays preferred
    - BA can refine state in a thread separate from tracking component
    - “Parallel Tracking and Mapping for Small AR Workspaces” Klein & Murray [ISMAR07]



(a) Bayesian Network (b) Markov Random Field

“Real-Time Monocular SLAM: Why Filter?” Strasdat et.al. [ICRA10]

Information to Keep Track of

- State
  - Camera poses of keyframes
  - 3D coordinates of reconstructed points
- Data evidence
  - 2D locations of detected keypoints
  - Descriptors of keypoints
  - Additional data: timestamps, IMU data, …
- Bookkeeping: Data association
  - Which 2D keypoints correspond to a certain 3D point?
  - Sometimes replicated multiple times for faster queries
    - Which keyframes have observed a given 3D point?
    - Which 3D point corresponds to a given 2D keypoint?
  - Some systems keep track of multiple descriptors per 3D point
    - Handles appearance changes of 3D points
    - Also helpful for relocalization

Keyframe-Based SLAM: Operation modes

- vSLAM system has 3 main modes of operation
  - Bootstrapping
    - Compute an initial 3D map
    - Mostly based on concepts from two-view geometry
  - Normal mode
    - Assumes a 3D map is available and incremental camera motion
    - Track points and use PnP for camera pose estimation
  - Recovery mode
    - Assumes a 3D map is available, but tracking failed: no incremental camera motion anymore
    - Relocalize camera pose w.r.t. previously reconstructed map

System Components

- Bootstrapping
  - Initial 3D map generation
- 3D tracker and PnP pose estimator
  - Processes incoming frames as quickly as possible
- Relocalization
  - Recovering from tracking failure
  - Can also be used for loop closure detection
- Mapping data structure
  - Point-pose graph
- Bundle adjustment
  - Runs in separate thread and refines estimates
  - Accesses mapping data structure

The Life of a Frame

Bootstrapping "The black art"

- IP Detection
- 2D Tracker
- Selection of first two keyframes
- Epipolar Pose Estimator
- Initial 2-view reconstruction
- Goto 'Normal' case

Normal case

- 3D Tracker
- Keyframe selection
- PnP Pose Estimator
- Refine Pose with BA
- Densify 3D points

'Failure' recovery mode and loop closure

- Extract 'expensive' features
- Query DB for top K results
- Geometric verification
- PnP Pose Estimator
- Goto 'Normal' case or insert loop in pose graph

Separate Thread

- Bundle Adjustment

vSLAM Results

"Double Window Optimisation for Constant Time Visual SLAM"
Strasdat et.al. [ICCV11]

The Life of a Frame

Bootstrapping “The black art”	IP Detection	2D Tracker	Selection of first two keyframes	Epipolar Pose Estimator	Initial 2-view reconstruction	Goto ‘Normal’ case
Normal case	3D Tracker	Keyframe selection	PnP Pose Estimator	Refine Pose with BA	Densify 3D points	
‘Failure’ recovery mode and loop closure	Extract ‘expensive’ features	Query DB for top K results	Geometric verification	PnP Pose Estimator	Goto ‘Normal’ case or insert loop in pose graph	
Separate Thread	Bundle Adjustment					

IP Detection

- Avoid clusters of interest points
  - RANSAC estimates suffer when many IPs are close together
- Roughly uniformly distributed IP
  - Introduce grid
  - Avoid imbalanced number of IPs in grid cells
- Be aware of complexity of IP detector and descriptor
  - SIFT is powerful, but expensive to compute
- Many options available
  - IP Detectors: FAST, Harris corner, Scale-space extrema (SIFT), MSER, …
  - Descriptors: image patch, BRISK, SIFT, …

The Life of a Frame

Bootstrapping “The black art”	IP Detection	2D Tracker	**Selection of first two keyframes**	Epipolar Pose Estimator	Initial 2-view reconstruction	Goto ‘Normal’ case
Normal case	3D Tracker	Keyframe selection	PnP Pose Estimator	Refine Pose with BA	Densify 3D points	
‘Failure’ recovery mode and loop closure	Extract ‘expensive’ features	Query DB for top K results	Geometric verification	PnP Pose Estimator	Goto ‘Normal’ case or insert loop in pose graph	
Separate Thread	Bundle Adjustment					

Initial Selection of Two Keyframes

- Avoid “non-parallax views”
  - Pure rotation of camera
    - In practice: “pure” depends on [unknown] depth of points
    - Motion of points at infinity will always appear as due to pure rotation
  - Low-parallax views
    - Small translations and forward motion
- Avoid planar scenes
  - Fundamental matrix is ill-defined for planar scenes
  - Essential can be estimated, but care must be taken!

Initial Selection of Two Keyframes

- How to avoid theses cases without knowing 3D structure and camera poses?
- Check for planar scene
  - Can correspondences be explained with homography?
    - If yes, raise red flag
- Check for sufficiently large parallax
  - Compensate for displacements due to camera rotation
    - Can be done very efficiently if gyroscope is available
  - Are remaining displacements sufficiently large?
    - If yes, good for triangulation
  - Compensation for camera rotation
    - Decompose essential into rotation and translation
    - Apply rotation as homography to image measurements (similar to stereo rectification)
    - Remaining displacement between feature points is due to translation

The Life of a Frame

Bootstrapping “The black art”	IP Detection	2D Tracker	Selection of first two keyframes	Epipolar Pose Estimator	Initial 2-view reconstruction	Goto ‘Normal’ case
Normal case	3D Tracker	Keyframe selection	PnP Pose Estimator	Refine Pose with BA	Densify 3D points	
‘Failure’ recovery mode and loop closure	Extract ‘expensive’ features	Query DB for top K results	Geometric verification	PnP Pose Estimator	Goto ‘Normal’ case or insert loop in pose graph	
Separate Thread	Bundle Adjustment					

Active Search

- Also known as Guided Search
- Avoid searching naïvely for IP and matching descriptors
- Setting: Incremental camera motion and known depth of 3D points
  - Good initial guess available where to expect corresponding point
  - Can also include motion model of camera (eg. constant velocity)
    - Or IMU measurements
  - For example: patch-based KLT tracker (Kanade-Lucas-Tomasi)
    - See also lecture on Wednesday
- Active Search and PnP makes vSLAM efficient!

The Life of a Frame

Bootstrapping "The black art"

- IP Detection
- 2D Tracker
- Selection of first two keyframes
- Epipolar Pose Estimator
- Initial 2-view reconstruction
- Goto 'Normal' case

Normal case

- 3D Tracker
- Keyframe selection
- PnP Pose Estimator
- Refine Pose with BA
- Densify 3D points

'Failure' recovery mode and loop closure

- Extract 'expensive' features
- Query DB for top K results
- Geometric verification
- PnP Pose Estimator
- Goto 'Normal' case or insert loop in pose graph

Separate Thread

- Bundle Adjustment

Bundle Adjustment

- Bundle adjustment is a big topic on its own
- Recent approaches
  - “Double Window Optimisation for Constant Time Visual SLAM” Strasdat et.al. [ICCV11]
    - Split BA objective into two terms
      - Cycle consistency of loops
      - Reprojection error
    - Minimize within window of recent frames
  - “Towards Linear-time Incremental Structure from Motion” Changchang Wu [3DV13]
    - Carefully designed sequential SfM system
    - Conjugate gradient with early termination instead of Cholesky

Ideas for Class Projects

- BA
  - Implementation of conjugate gradient based BA approach with double window optimization
- Exploit IMU data
  - Gyroscope, accelerometer, compass
  - Motion field for feature tracking
  - Accelerometer provides measurements in metric units
    - Very noisy measurements
    - Estimation of absolute scale still possible
- Self-calibration App (aka. auto-calibration)
  - Assumptions about intrinsics lead to constraint for each frame on camera matrices
  - Examples: Square pixels, constant but unknown focal length, …
- Line-based SfM
  - Lines are strong cues for pose estimation
  - Especially in indoor scenes
- Dense reconstructions on the phone

?