No. 4

動きからの３次元復元

Structure from Motion and SLAM

担当教員：向川康博・田中賢一郎

Slide credits

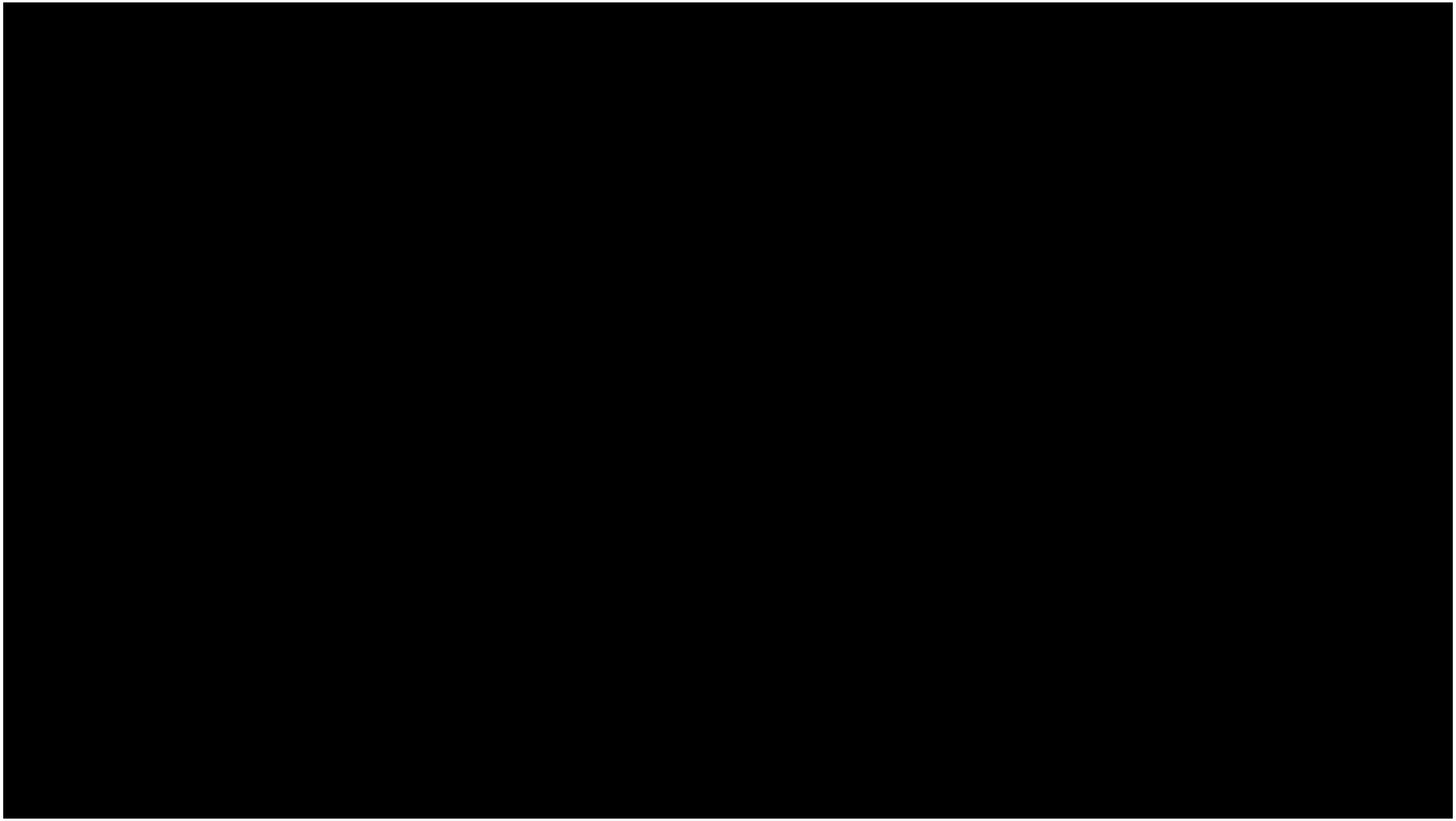
- Special Thanks: Some slides are adopted from other instructors' slides.
  - Tomokazu Sato, former NAIST CV1 class
  - James Tompkin, Brown CSCI 1430 Fall 2017
  - Ioannis Gkioulekas, CMU 16-385 Spring 2018

  - We also thank many other instructors for sharing their slides.

Today's topics

- Reminder: calibration and stereo
- 2-view Structure from Motion (SfM)
- Multi-view SfM
  - Large-scale SfM
- visual SLAM

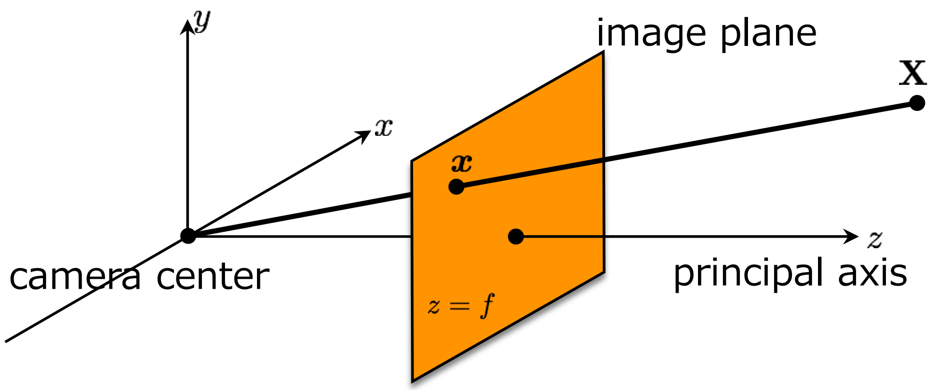
Direct Sparse Odometry



https://www.youtube.com/watch?v=C6-xwSOOdqQ

Recap: Calibration and Stereo

- Camera calibration

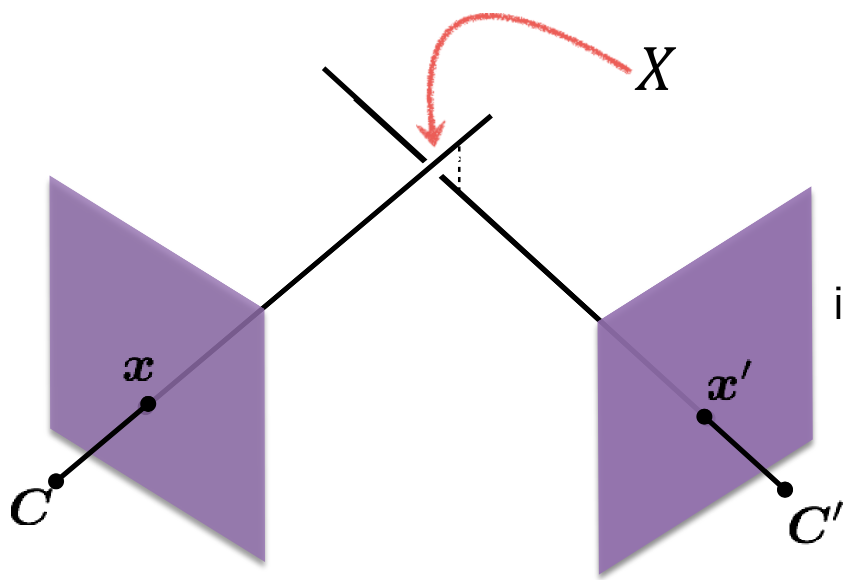


$$\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X}$$

known estimate known

Solve Perspective n-Points (PnP) problem

- Stereo



$$\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X}$$

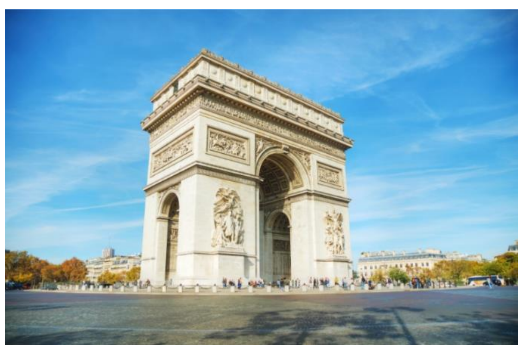
known known estimate

Solve Triangulation problem

Reconstruction（再構成）

	Structure (scene geometry)	Motion (camera geometry)	Measurements
Pose Estimation	known	estimate	3D to 2D correspondences
Triangulation	estimate	known	2D to 2D correspondences
Reconstruction	estimate	estimate	2D to 2D correspondences

How do you reconstruct?



We don't know both scene geometry and camera geometry

Reconstruction Problem

- Problems so far

Camera calibration

$$\underset{\text{known}}{\boldsymbol{x}} = \underset{\text{estimate}}{\boldsymbol{M}}\underset{\text{known}}{\boldsymbol{X}}$$

Triangulation (Stereo)

$$\underset{\text{known}}{\boldsymbol{x}} = \underset{\text{known}}{\boldsymbol{M}}\underset{\text{estimate}}{\boldsymbol{X}}$$

- Can we jointly estimate $\boldsymbol{M}$ and $\boldsymbol{X}$?

$$\underset{\text{known}}{\boldsymbol{x}} = \underset{\text{estimate}}{\boldsymbol{M}}\underset{\text{estimate}}{\boldsymbol{X}}$$

2-view SfM

Structure from Motion

Two-view SfM

1. Compute the Fundamental Matrix $\boldsymbol{F}$ from points correspondences
   8-point algorithm

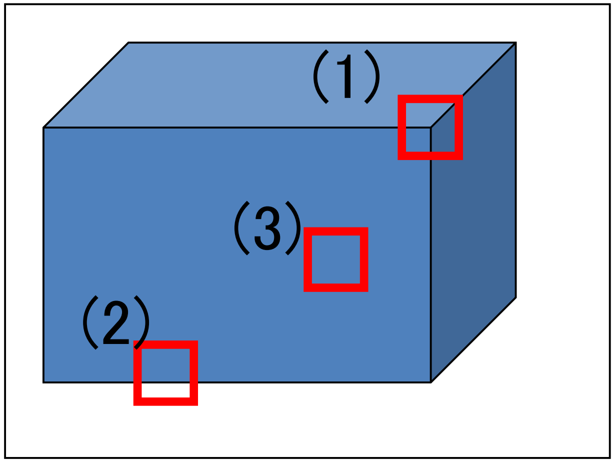
Image feature matching

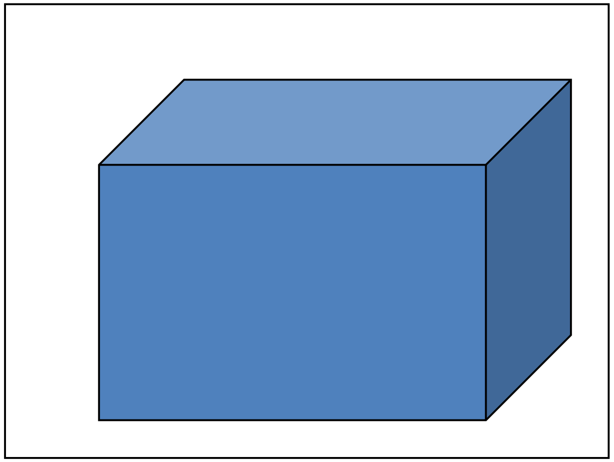
$$\boldsymbol{p}'^{\top}\boldsymbol{F}\boldsymbol{p} = 0$$

Estimate $\boldsymbol{F}$ from matched pairs

Good feature point

- Mini-report:2
  - Which is the good feature point to find corresponding point uniquely? What is the reason?
    - (1) corner
    - (2) edge
    - (3) flat



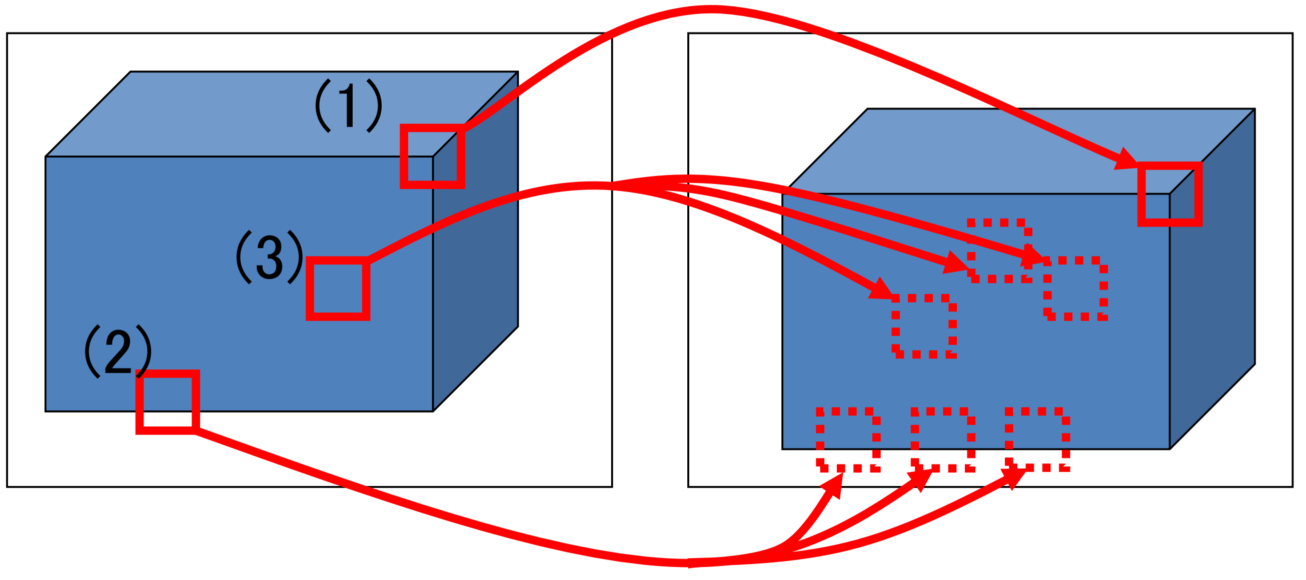


Good feature point

(1) corner
- ✓ easy to find

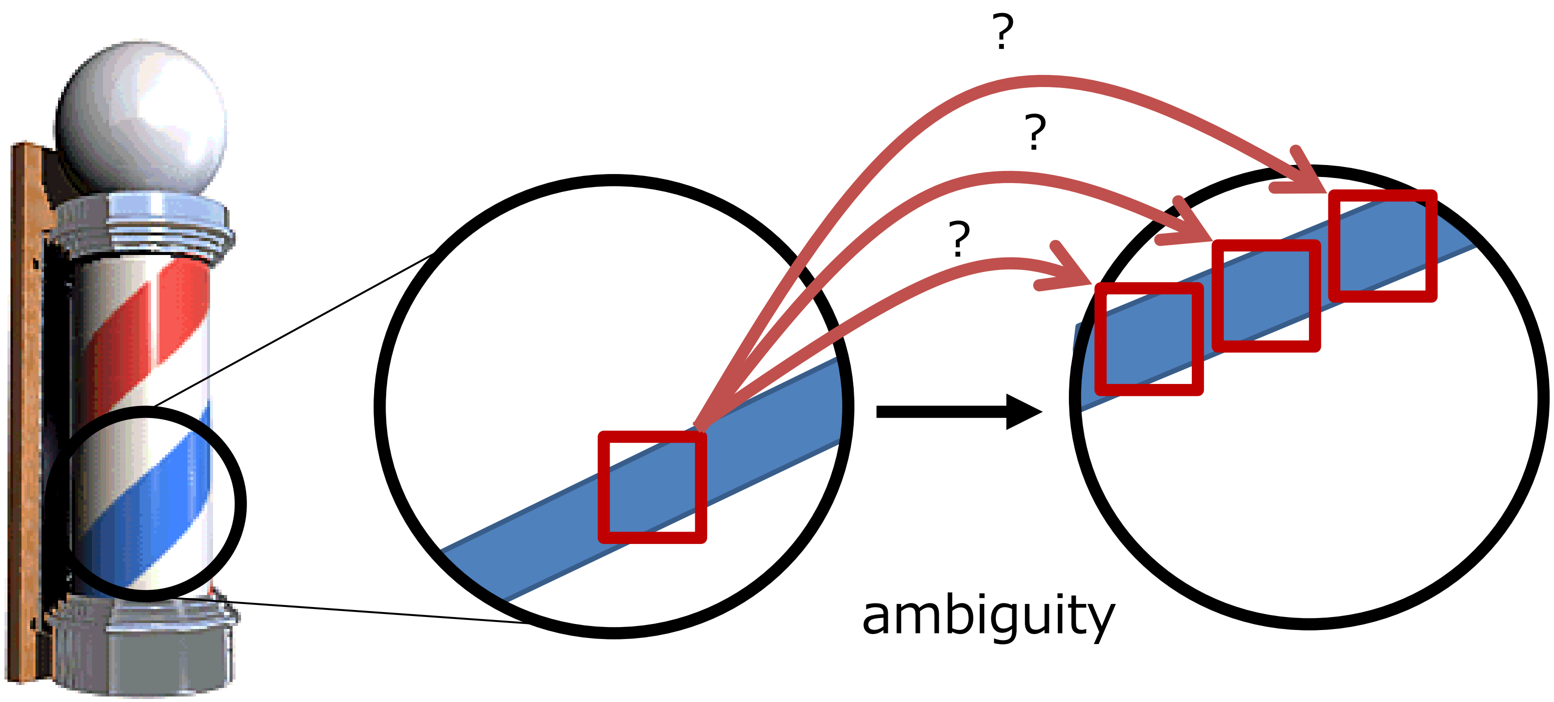
(2) edge
- ✓ difficult due to **aperture problem**（窓問題）

(3) flat
- ✓ difficult because no clues



Aperture problem (窓問題)

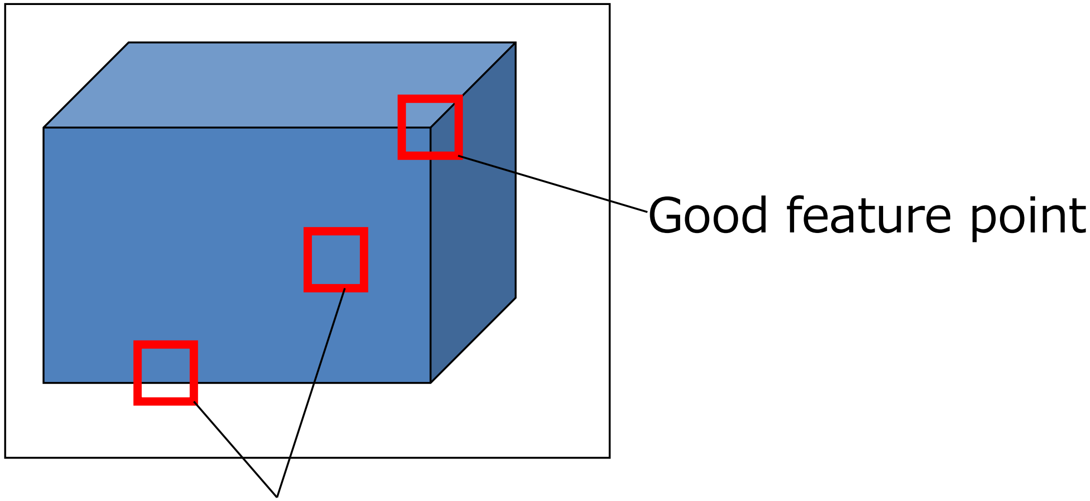
- Human eyes interpret conveniently.
- Corresponding point cannot be uniquely determined.



barber-pole illusion

Interest operators

- Corner detectors to find small areas which are suitable as feature points
  - Moravec corner detector (1980)
  - Harris corner detector (1988)
  - KLT(Kanade-Locus-Tomasi) corner detector(1991)
- Check whether small area contains edges in multiple directions



Bad feature points: corresponding point cannot be determined uniquely.

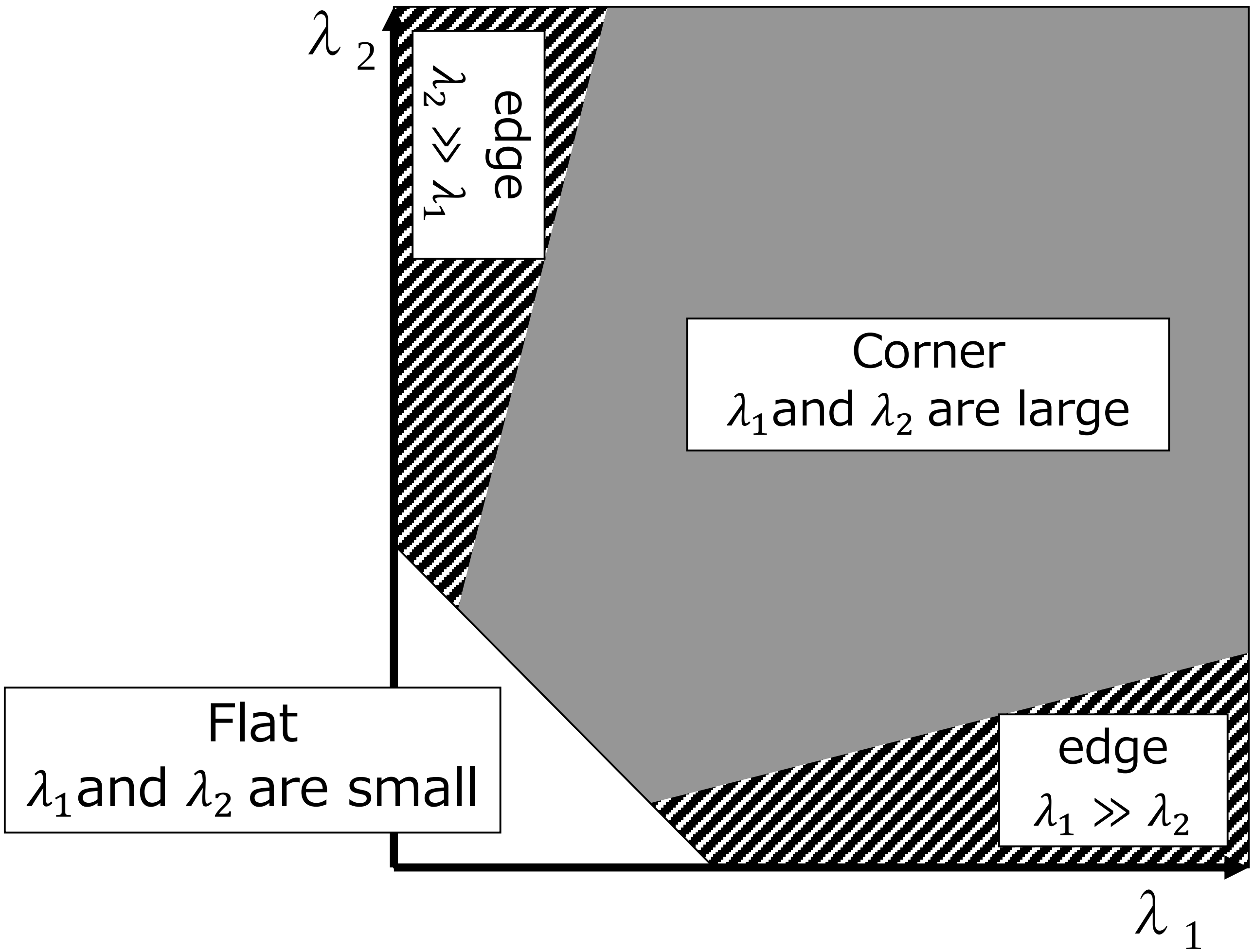
Harris corner detector

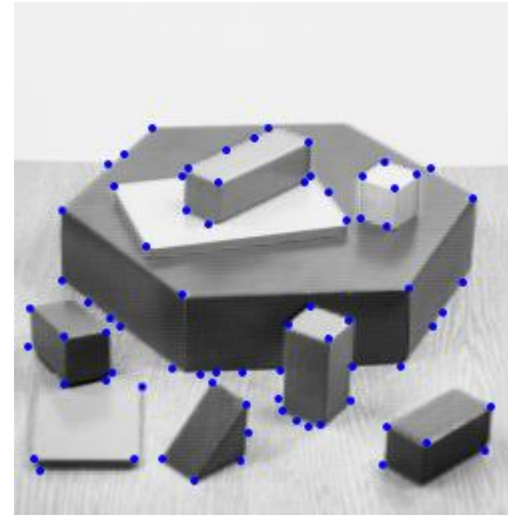
- Principal component analysis of gradient in small area
- Classification by Eigenvalues λ_1 and λ_2
  1. X and Y direction differentiation
  $$\boldsymbol{X = \partial I/\partial x \qquad Y = \partial I/\partial y}$$
  2. Calculation of variance and covariance
  $$\mathbf{A = X^2 \otimes w \qquad B = Y^2 \otimes w \qquad C = (XY) \otimes w}$$
  $$\mathbf{w_{u,v} = exp - (u^2 + v^2)/2\sigma^2}$$
  3. Eigenvalues of the variance-covariance matrix
  $$\boldsymbol{M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad \lambda_1, \lambda_2}$$

Principal component analysis
of image gradient

Harris corner detector

- Eigenvalue-based classification





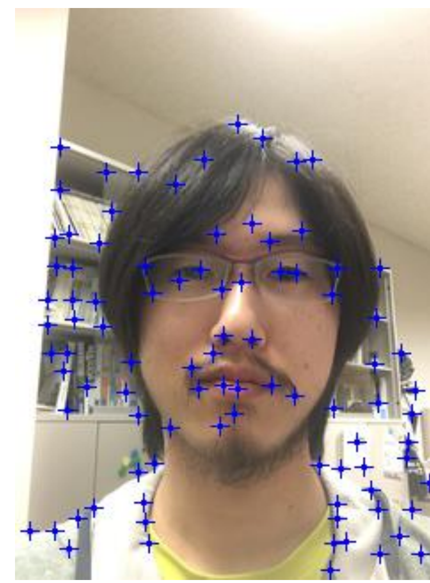


Image feature descriptor
- SIFT -

- Features that are invariant to changes in scale and rotation of the image
- Intensity gradient histogram
  - Rotate coordinate axis in gradient direction (invariant to rotational change)
  - Normalize the vector sum (to reduce the influence of lighting changes)

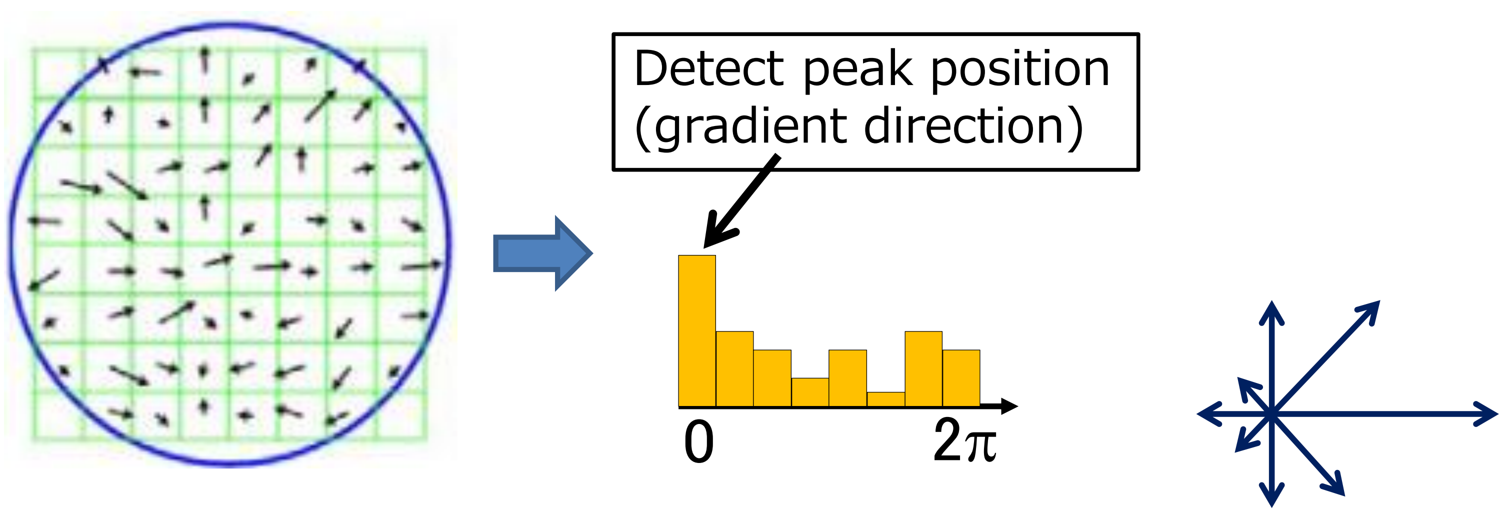


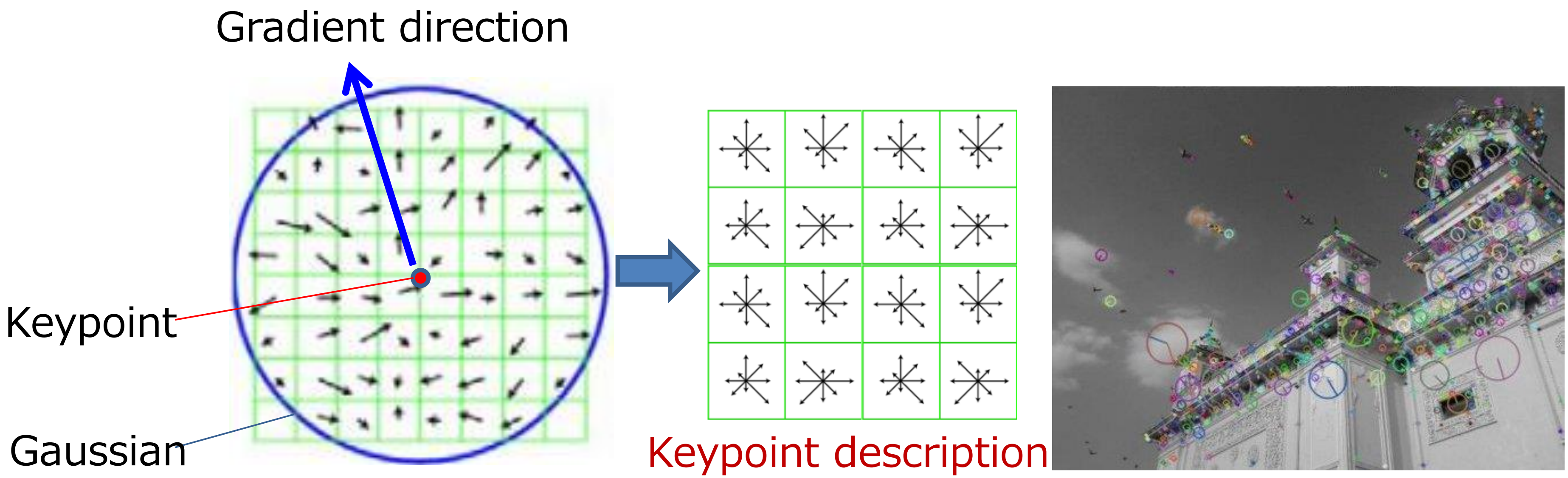
Image gradient

Angle histogram

Image feature descriptor

- SIFT -

- Accumulate while overlapping gradient information around key points with Gaussian function
- Histogram in 8 directions every 4 × 4 = 16 blocks
  - 128 dimensional feature vector



Two-view SfM

1. Compute the Fundamental Matrix $\boldsymbol{F}$ from points correspondences
   8-point algorithm

Image feature matching

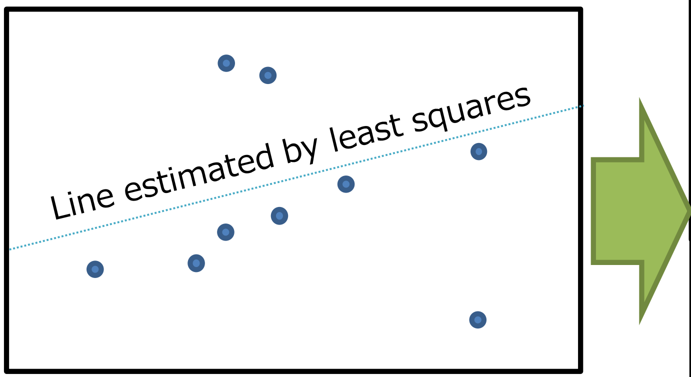
$$\boldsymbol{p}'^{\top}\boldsymbol{F}\boldsymbol{p} = 0$$

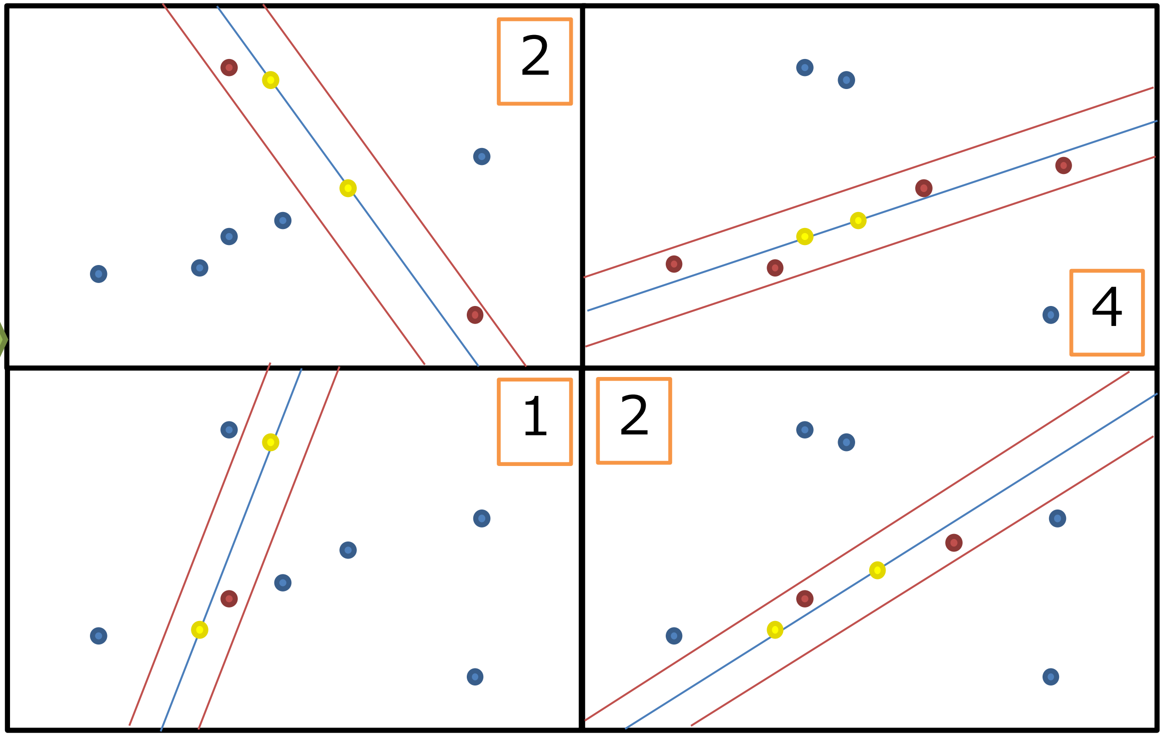
Estimate $\boldsymbol{F}$ from matched pairs

RANSAC
(RANdom SAmpling Consensus)

Eliminating outliers (外れ値).

line fitting example





1. Randomly sample two points from given points

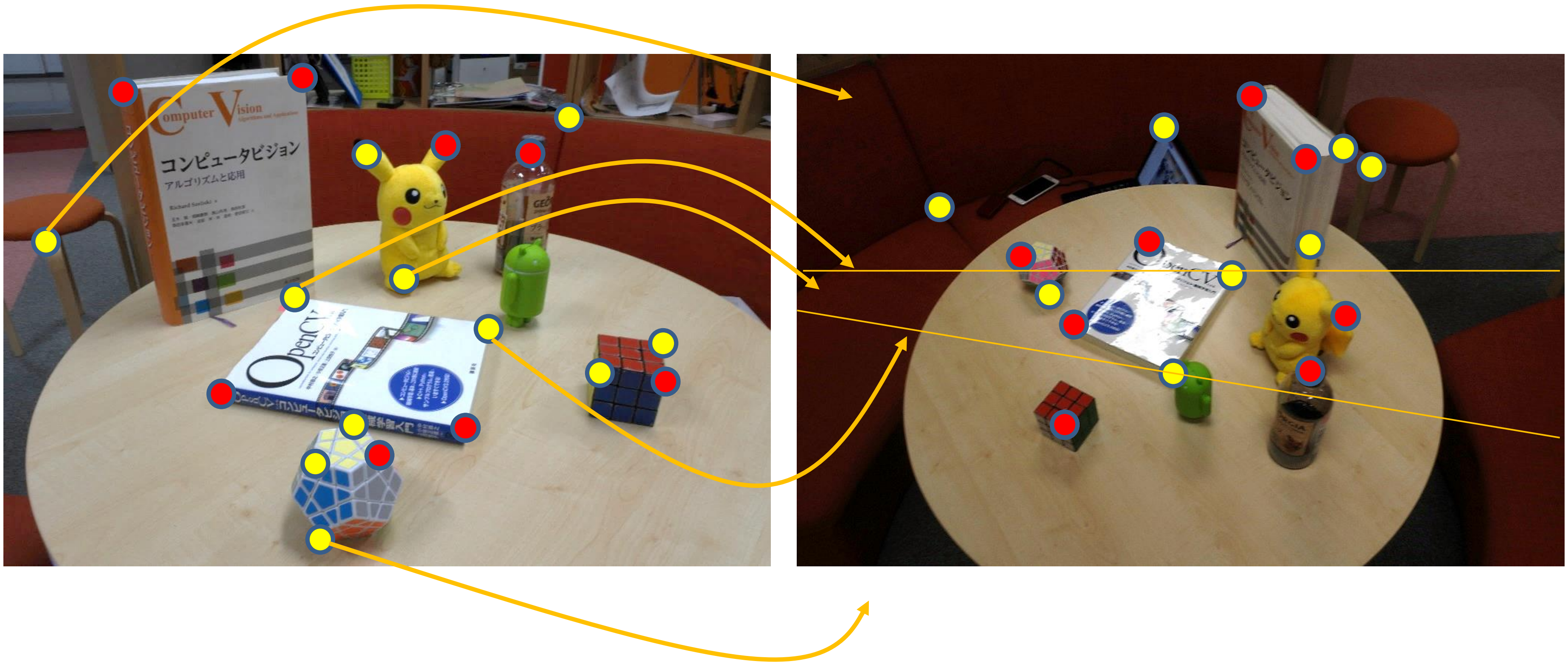
2. Count inliers for the line that passes selected two points.

3. Repeat 1 and 2 for given number of iterations.

4. Select the line that maximize the number of inliers.

8-point algorithm

- Randomly sampled 8 points



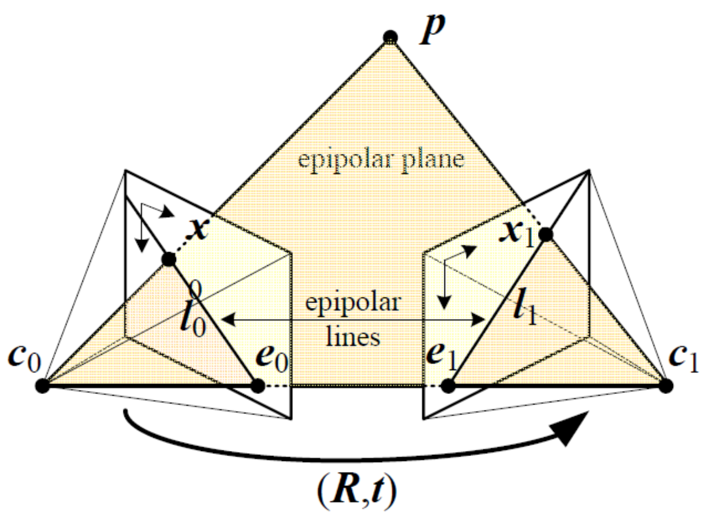
Do the remaining points agree?

Two-view SfM

1. Compute the Fundamental Matrix $\boldsymbol{F}$ from points correspondences
   8-point algorithm

2. Compute the camera matrices $\boldsymbol{M}$ from the Fundamental matrix
   $\boldsymbol{M} = [\boldsymbol{I}|\boldsymbol{0}]$ **and** $\boldsymbol{M}' = \left[[\boldsymbol{e}_{\boldsymbol{x}}]\boldsymbol{F}|\boldsymbol{e}'\right]$

Estimation of camera pose from pairs of feature points



Even if we do not know 3D positions of feature points, we can estimate camera's rotation $\mathbf{R}$ and translation direction $\mathbf{t}$ by epipolar constraint from multiple 2D positions of corresponding feature pairs.

$$M = [I|0] \text{ and } M' = [[e_x]F|e']$$

It should be noted that, we cannot recover the scale of $\mathbf{t}$ (actual distance between c_0 and c_1).

東武ワールドスクエア

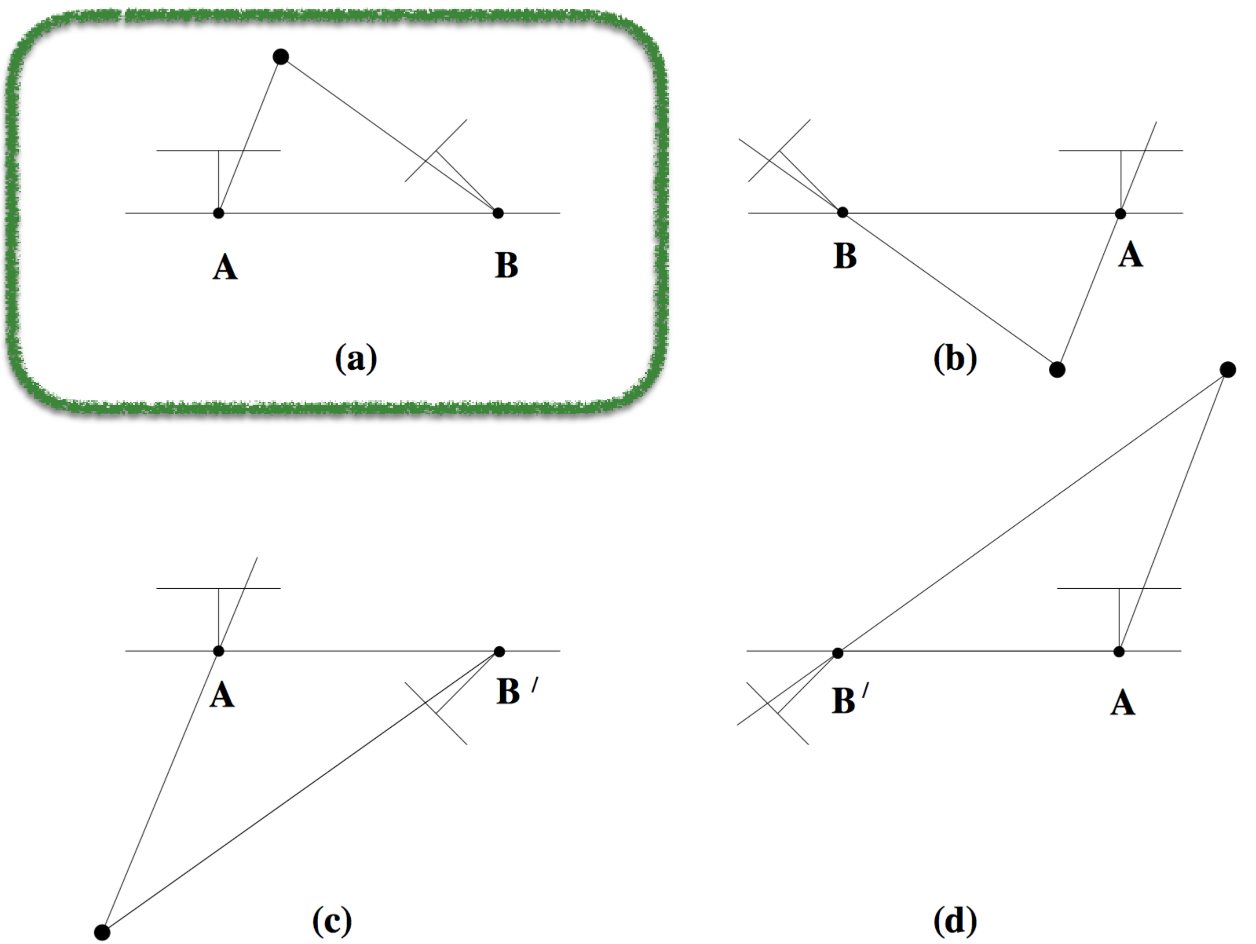
Unknown real size

- Miniature effect

https://ameblo.jp/9999999999999999999/entry-10332221773.html

Front/back ambiguity

- Find the configuration where the points is in front of both cameras



Two-view SfM

1. Compute the Fundamental Matrix $\boldsymbol{F}$ from points correspondences
   8-point algorithm

2. Compute the camera matrices $\boldsymbol{M}$ from the Fundamental matrix
   $\boldsymbol{M} = [\boldsymbol{I}|\boldsymbol{0}]$ **and** $\boldsymbol{M}' = \left[[\boldsymbol{e}_x]\boldsymbol{F}|\boldsymbol{e}'\right]$

3. For each point correspondence, compute the point $\boldsymbol{X}$ in 3D space
   Triangulate with $\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X}$ and $\boldsymbol{x}' = \boldsymbol{M}'\boldsymbol{X}$

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:

$$\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X} = \left(\frac{1}{k}\boldsymbol{M}\right)(k\boldsymbol{X})$$

It is impossible to recover the absolute scale of the scene!

Structure from motion ambiguity

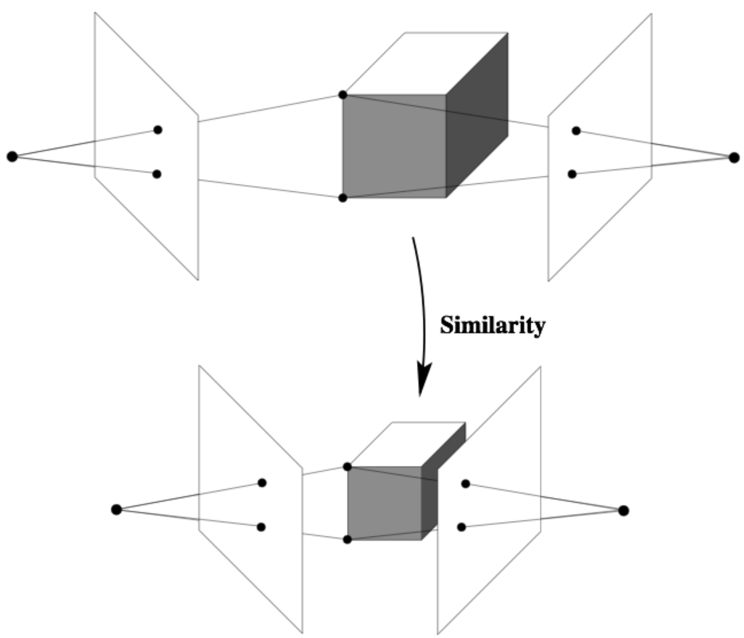
- If we scale the entire scene by some factor k and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same

- More generally: if we transform the scene using a transformation $\boldsymbol{Q}$ and apply the inverse transformation to the camera matrices, then the images do not change

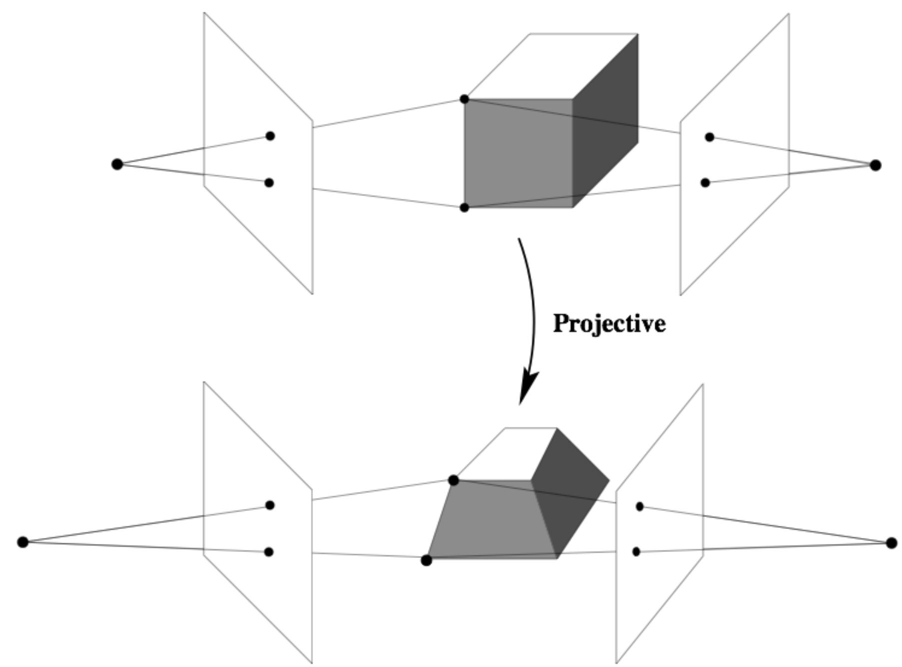
$$\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X} = \left(\boldsymbol{M}\boldsymbol{Q}^{-1}\right)(\boldsymbol{Q}\boldsymbol{X})$$

Reconstruction ambiguity

Calibrated cameras
(similarity projection ambiguity)



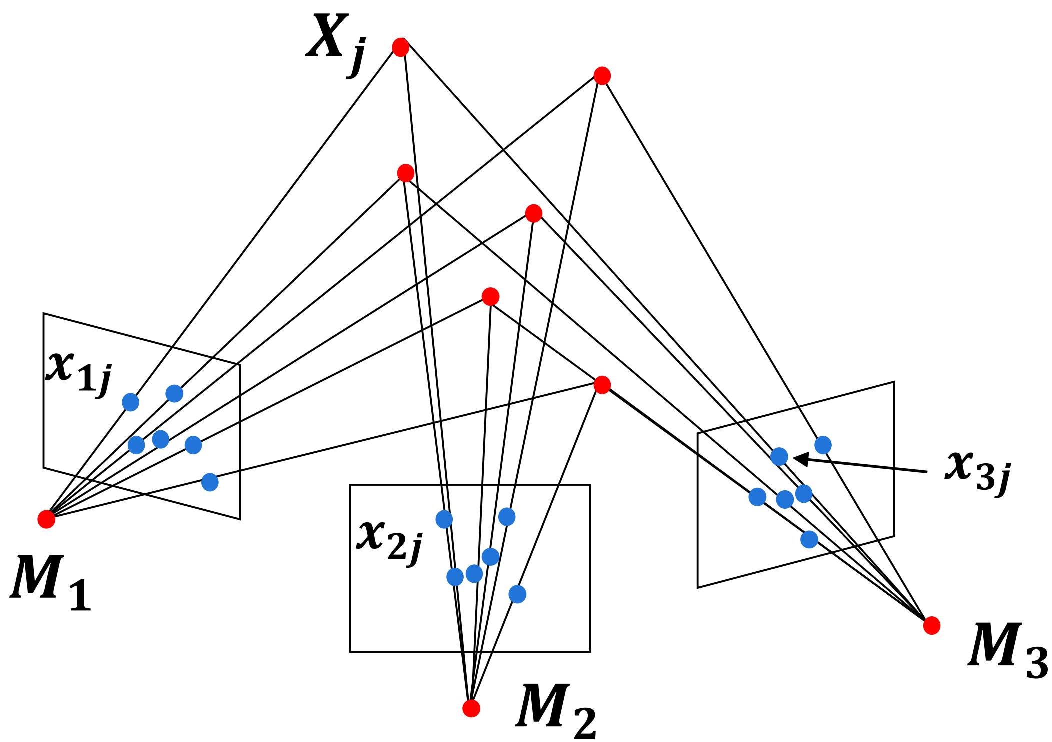
Uncalibrated cameras
(projective projection ambiguity)



Multi-view SfM

Projective structure from motion

- Given: m images of n fixed 3D points
  - $\boldsymbol{x_{ij}} = \boldsymbol{M_i X_j}$, $i = (1, \ldots, m,)$ $j = (1, \ldots, n)$
- Problem: estimate m projection matrices $\boldsymbol{M_i}$ and n 3D points $\boldsymbol{X_j}$ from the mn correspondences $\boldsymbol{x_{ij}}$



Sequential SfM

Multi-view, ordered images

Initialization

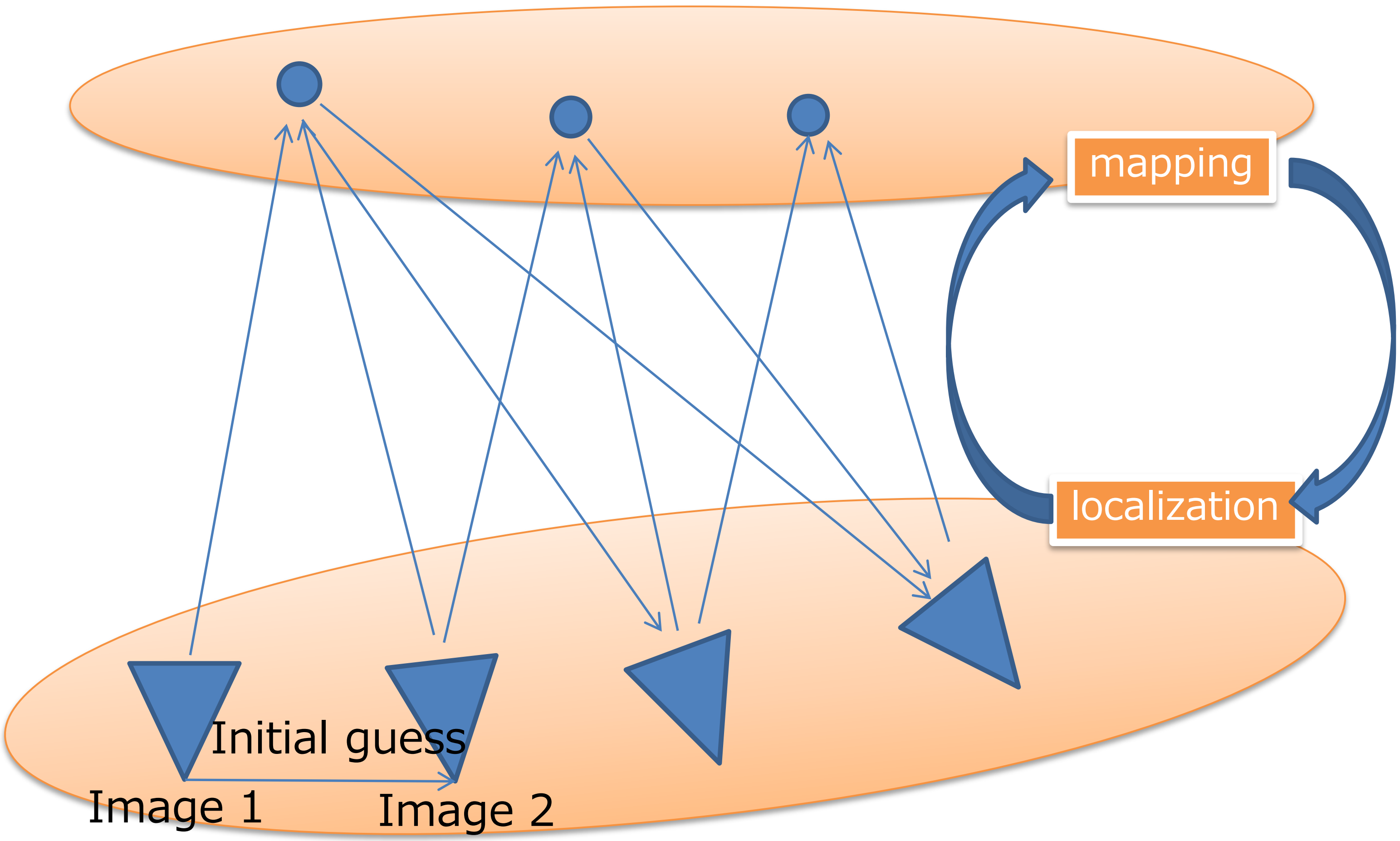
- Initialize by 2-view SfM

1. Compute the Fundamental Matrix F from points correspondences
   8-point algorithm

2. Compute the camera matrices M from the Fundamental matrix
   $\boldsymbol{M} = [\boldsymbol{I}|\boldsymbol{0}]$ **and** $\boldsymbol{M}' = \left[[\boldsymbol{e}_x]\boldsymbol{F}|\boldsymbol{e}'\right]$

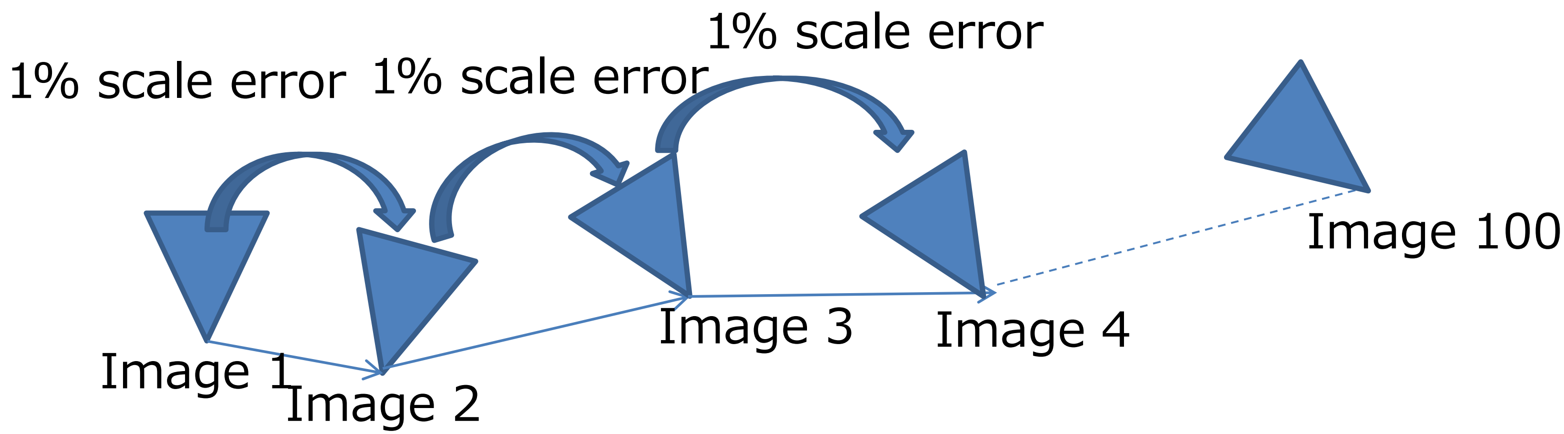
3. For each point correspondence, compute the point $\boldsymbol{X}$ in 3D space
   Triangulate with $\boldsymbol{x} = \boldsymbol{M}\boldsymbol{X}$ and $\boldsymbol{x}' = \boldsymbol{M}'\boldsymbol{X}$

Idea for sequential SfM



If we know camera poses for a pair of image 1 and 2, we can continue to estimate camera poses and 3-D structure for new input by repeating 'mapping' and 'localization'.

Problem of accumulative errors by chaining relative poses



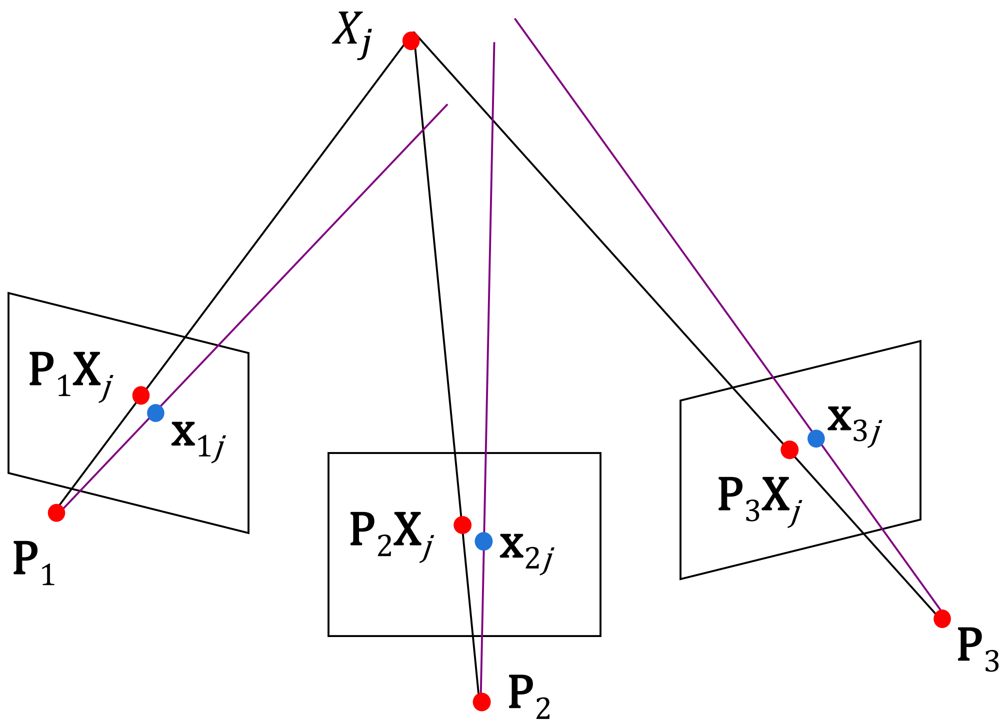
If relative camera poses are estimated with +1% biased scale error for each pair, the scale error at 100 frame will be $1.01^{100} = 2.70 = 270\%$.

*This kind of effect is called as scale drift.

Bundle adjustment

- Non-linear method for refining structure and motion
- Minimizing reprojection error

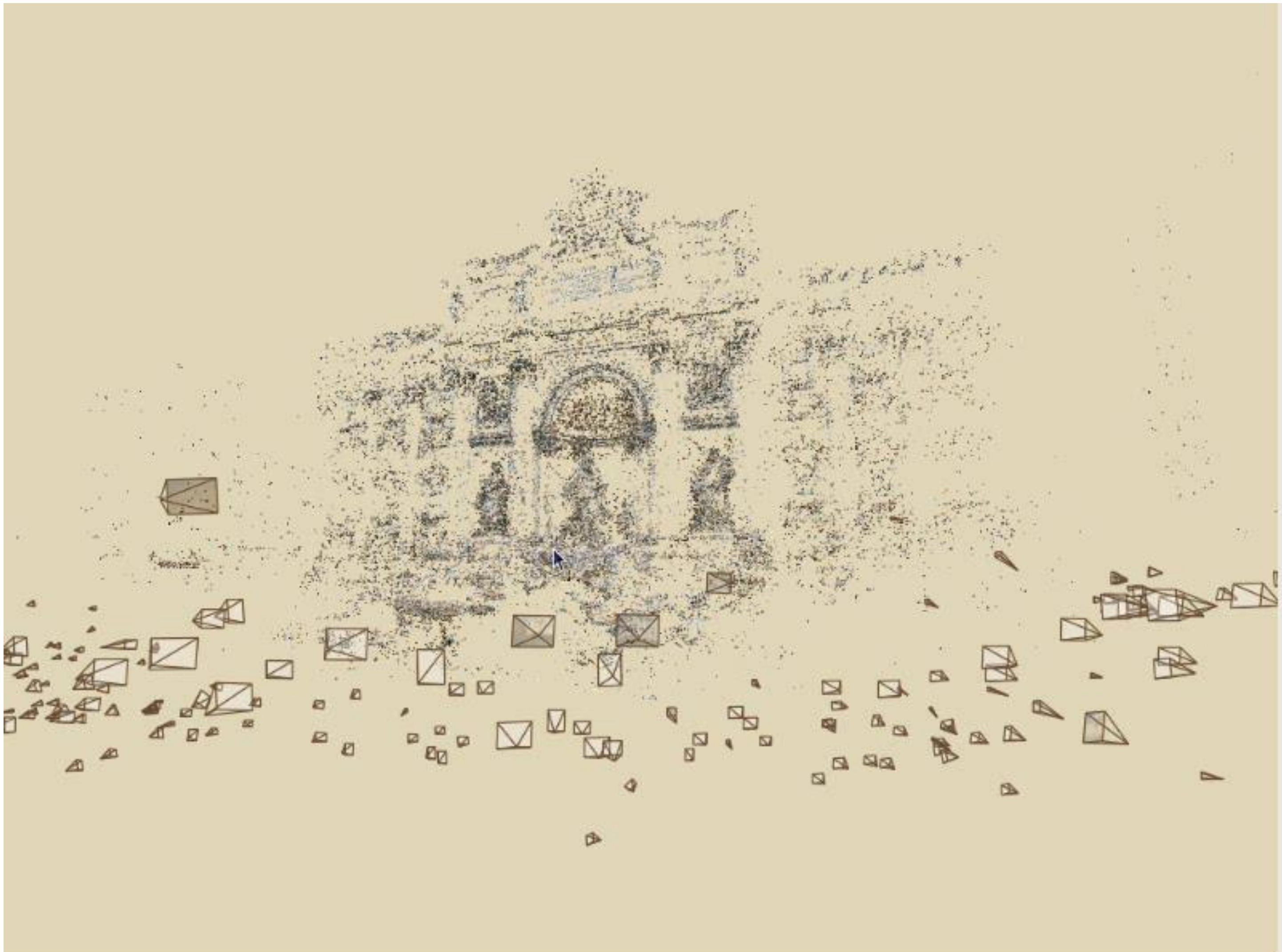
$$E(\boldsymbol{M},\boldsymbol{X}) = \sum_{i=1}^{m}\sum_{j=1}^{n} D\left(\boldsymbol{x}_{ij}, \boldsymbol{M}_i\boldsymbol{X}_j\right)^2$$



Large-scale SfM

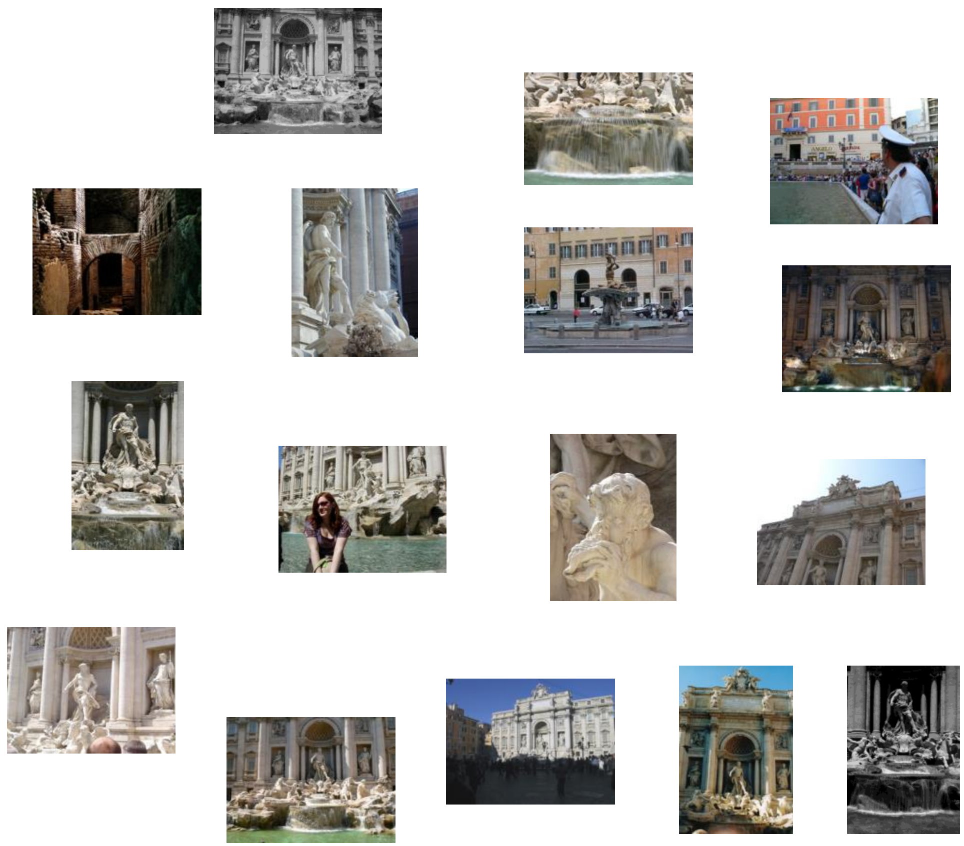
Multi-view, non-ordered images

Photo Tourism



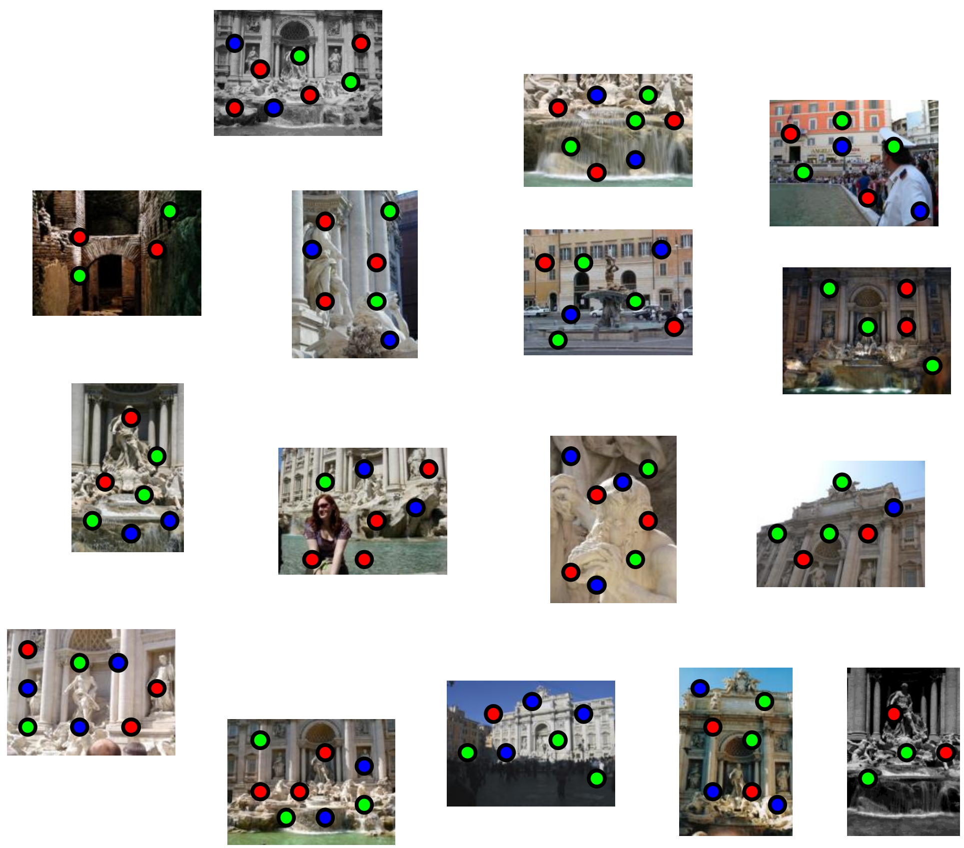
Feature detection

Detect features using SIFT [Lowe, IJCV 2004]



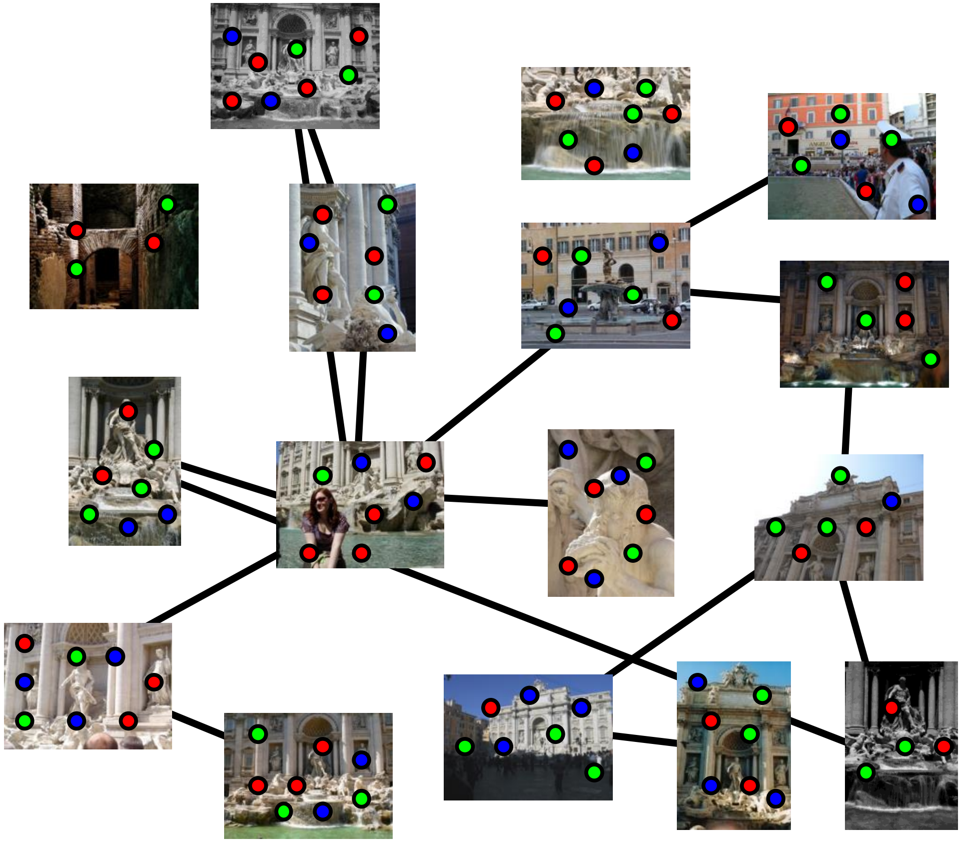
Feature description

Describe features using SIFT [Lowe, IJCV 2004]



Feature matching

Refine matching using RANSAC to estimate fundamental matrix between each pair



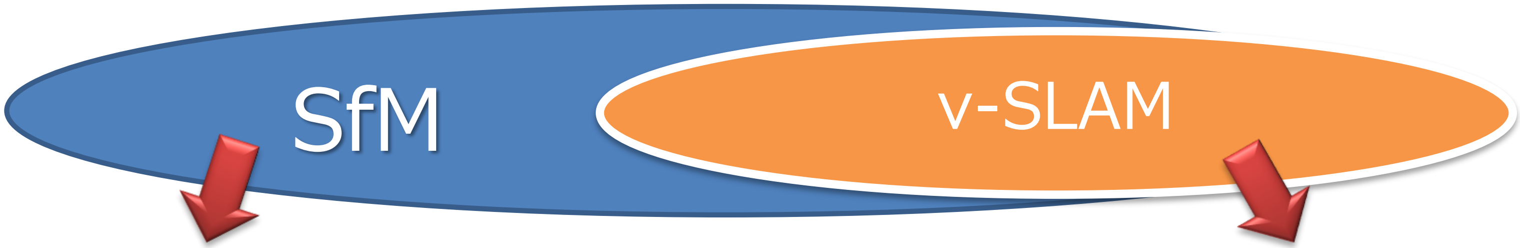
Incremental structure from motion

Final reconstruction

visual SLAM

Simultaneous Localization and Mapping

Relationship between SfM and v-SLAM

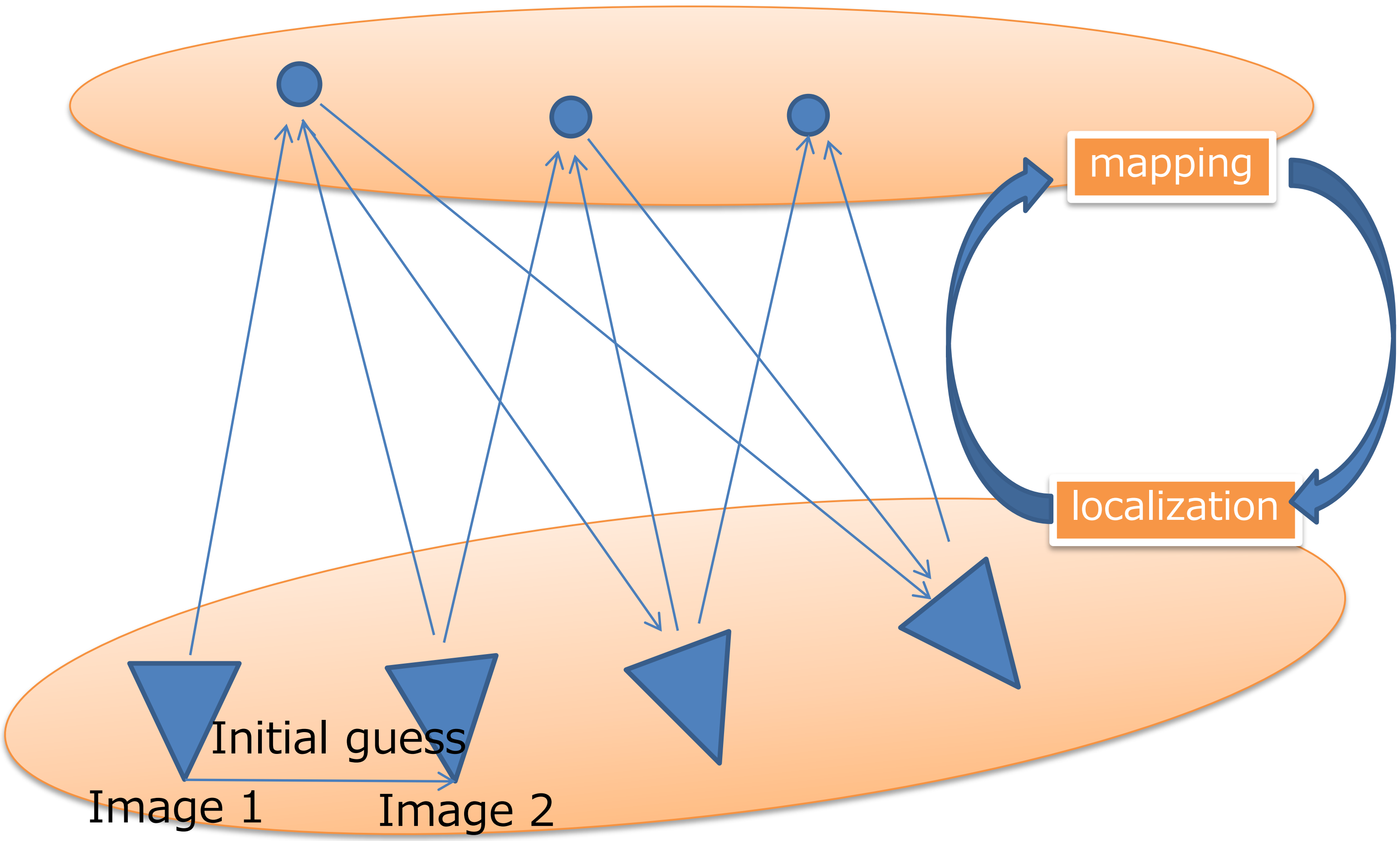


Offline, high accuracy

Real-time, successive output

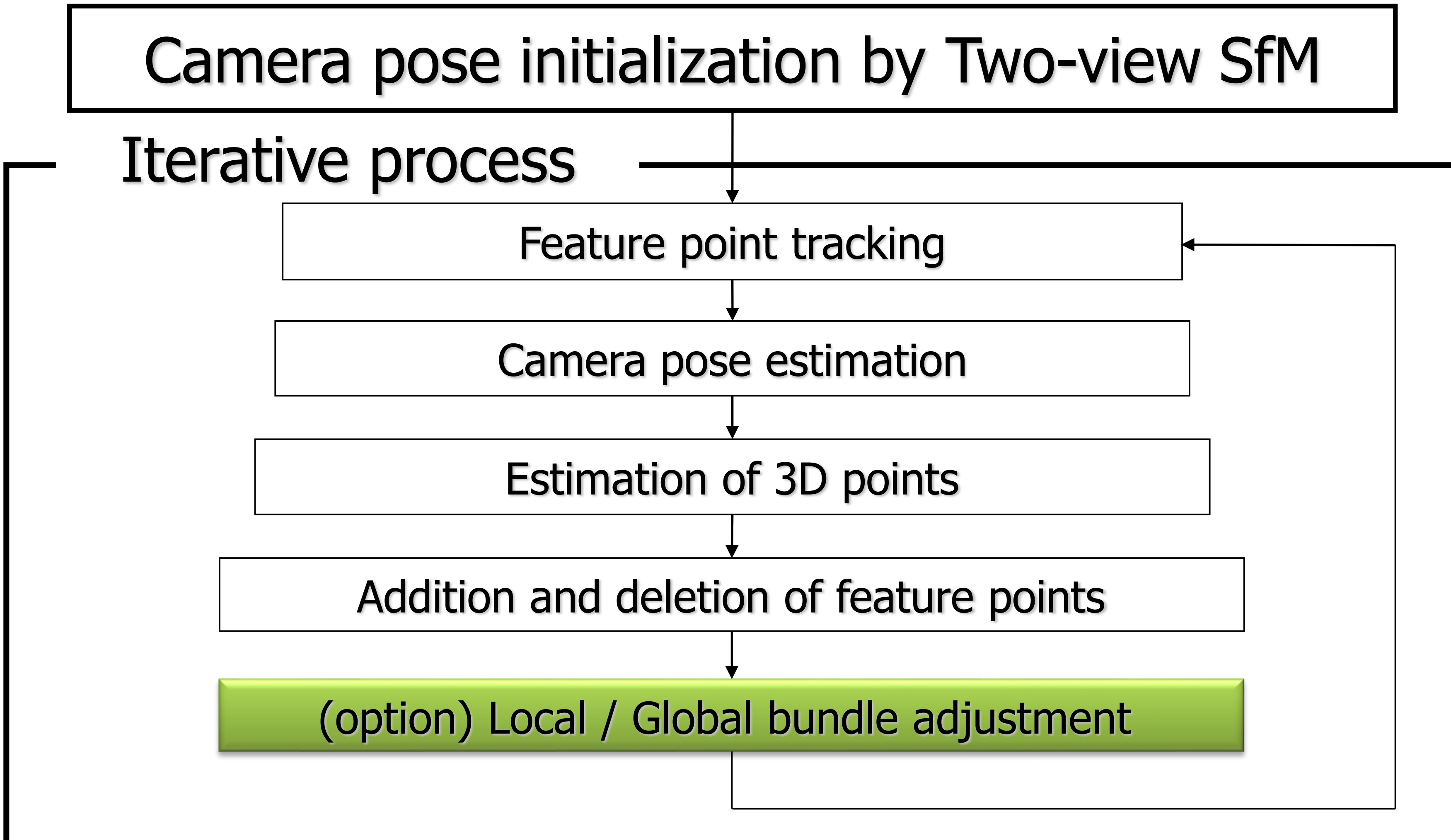
	SfM	v-SLAM
Input	Images (non-ordered is OK)	Video
Real-time	Not necessary	Required
Output	Batch	Successive
Available information for estimation	All frames	Data acquired before the target frame
Recovery from failure	Easier	Not easy
Feature tracking	Not easy (for non-successive image input)	Easier
Accumulation of errors	Smaller	Bigger / Faster

Idea for sequential SfM



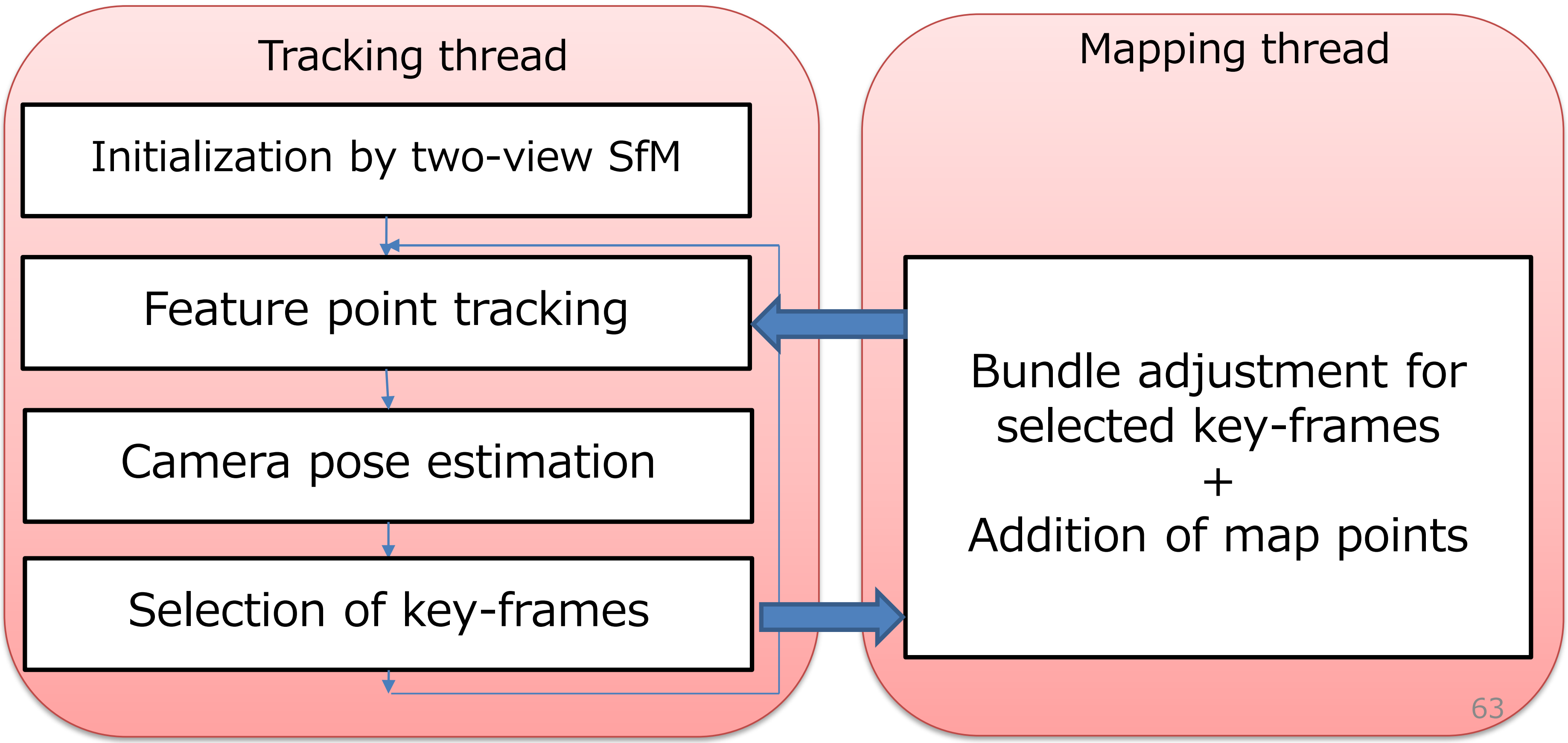
If we know camera poses for a pair of image 1 and 2, we can continue to estimate camera poses and 3-D structure for new input by repeating 'mapping' and 'localization'.

Basic pipeline of sequential visual SLAM



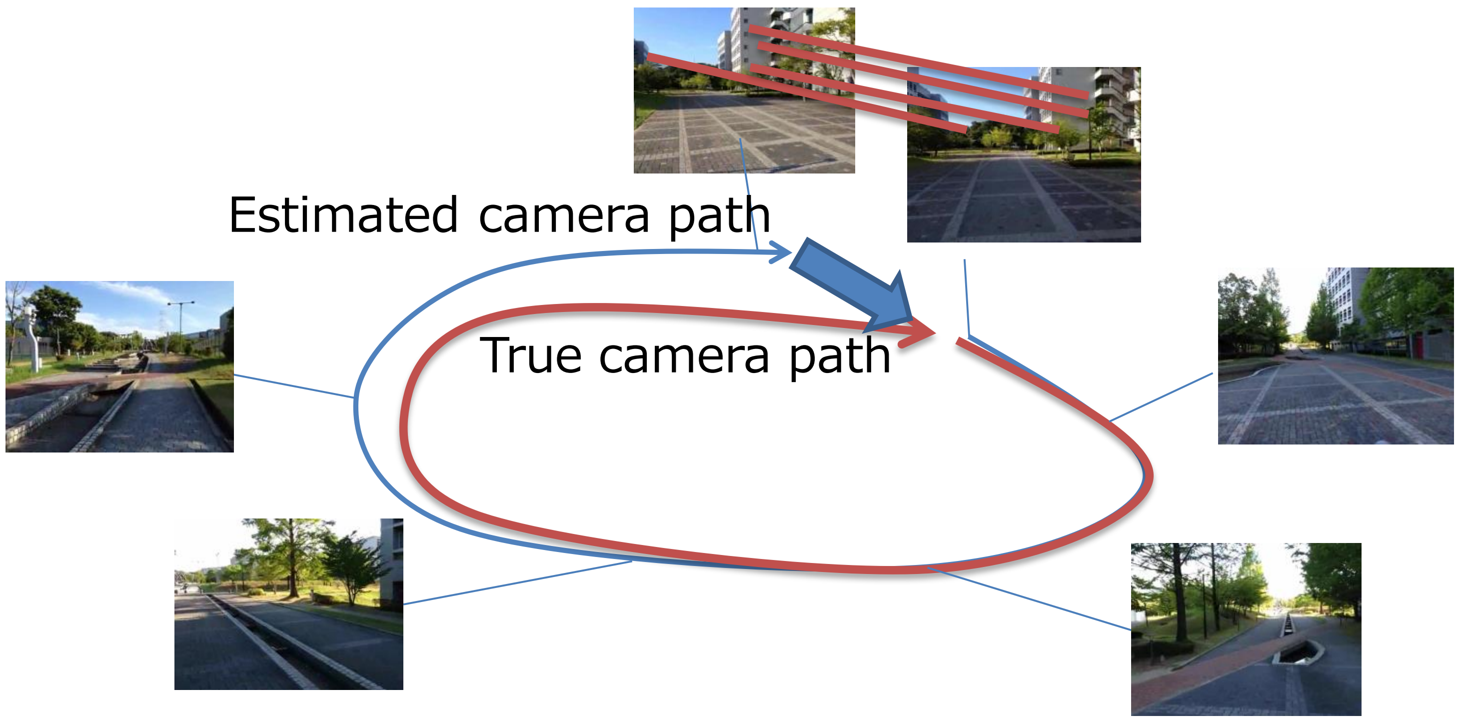
Parallel Tracking and Mapping

Local bundle adjustment is asynchronously processed for minimizing accumulation of errors using selected key-frames in order not to prevent real-time processing.



Reduction of accumulation errors / re-start from tracking failure

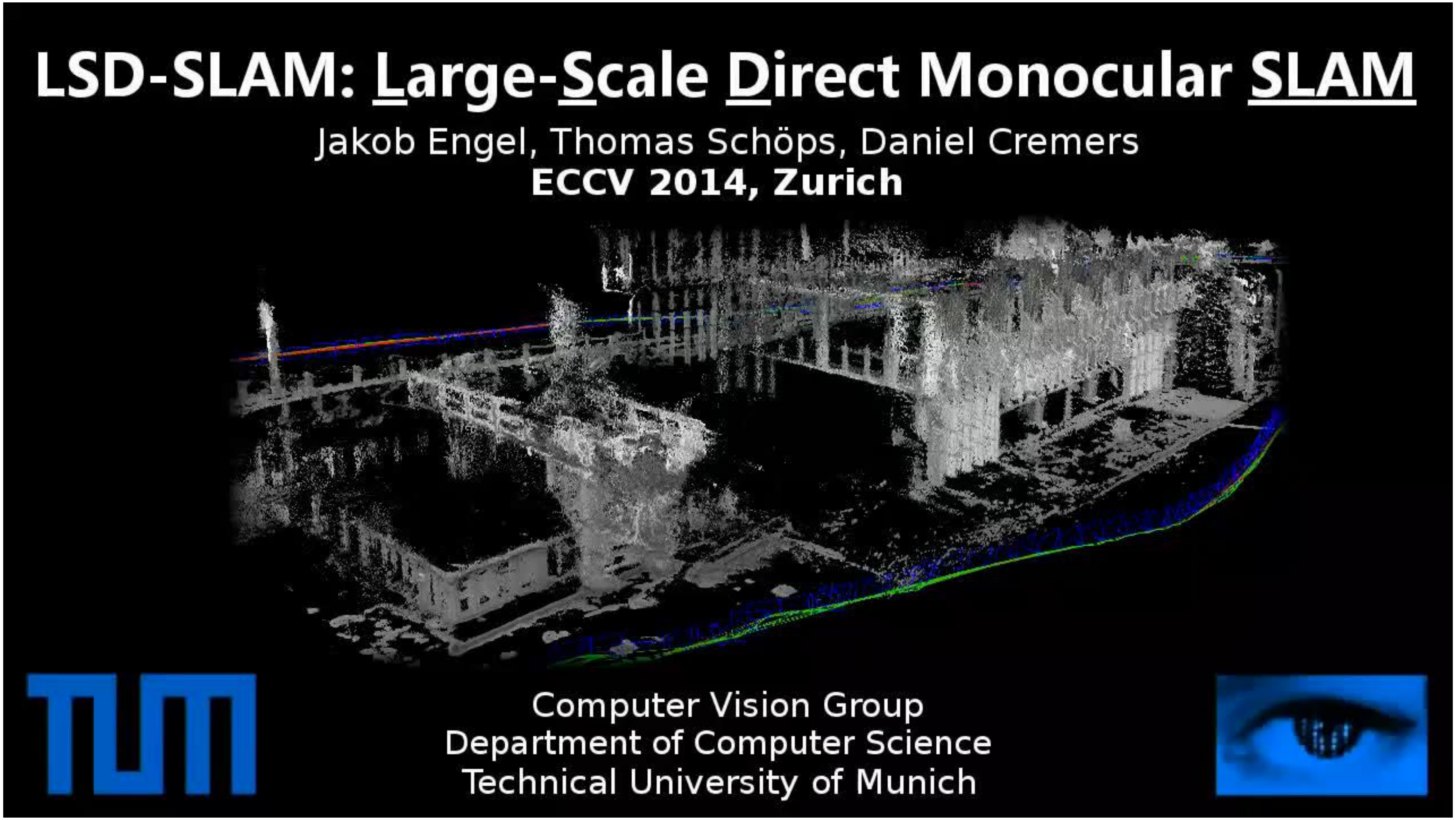
Loop closing, re-localization



Basic flow

1. Find similar image of current input from already observed images
2. Find corresponding points between current and selected images.
3. Optimize data using bundle adjustment / Estimate camera pose.

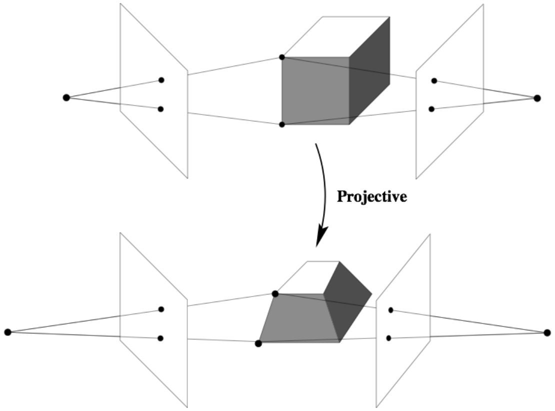
LSD-SLAM: Large Scale Direct Monocular SLAM



https://www.youtube.com/watch?v=GnuQzP3gty4

Final report

- Explain the reason of these strange views



Inconsistent?

- Physically consistent

$$x = MX$$

- Humans interpret conveniently.
  - 90 degree, rectangle, parallel,...

$$x = MQ^{-1}QX$$

  - Estimation of skewed shape

$$QX$$