Place Recognition and Kidnapped Robots

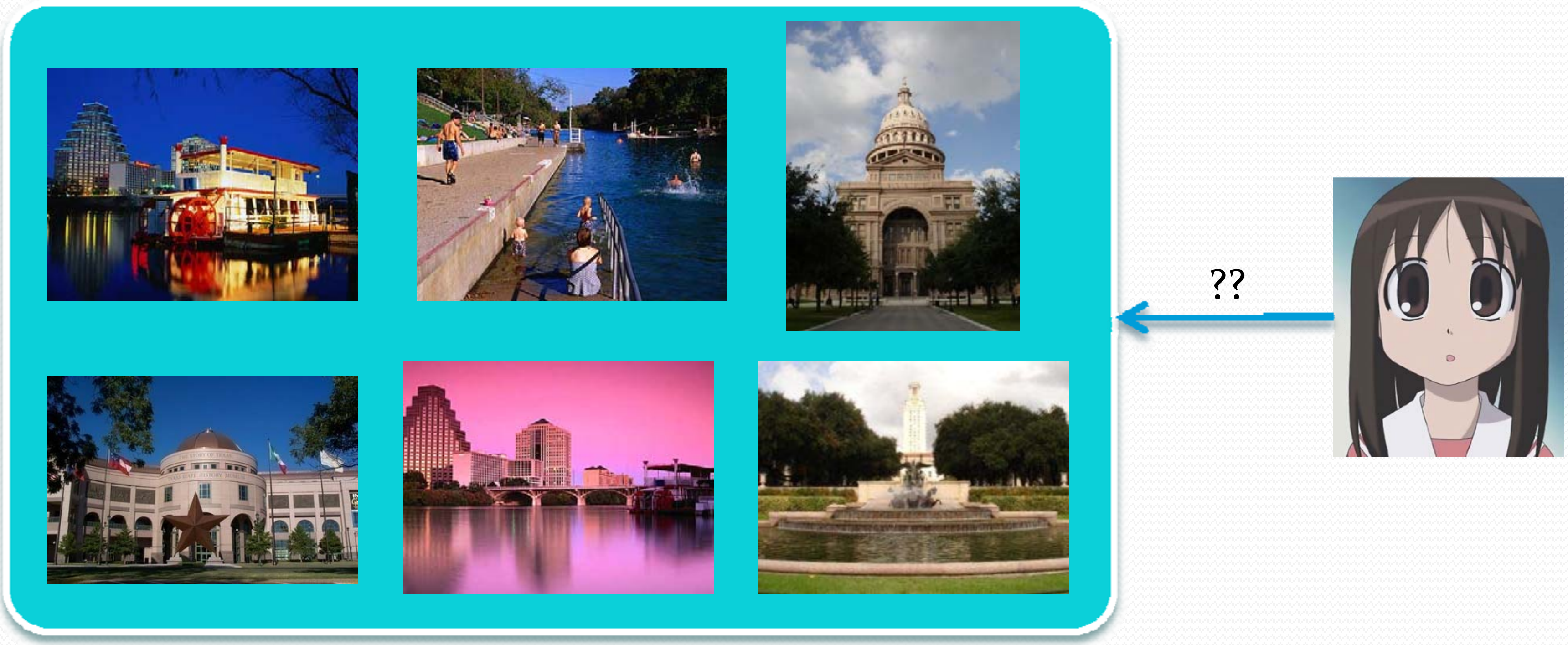
Visual Recognition and Search

April 18, 2008

Joo Hyun Kim

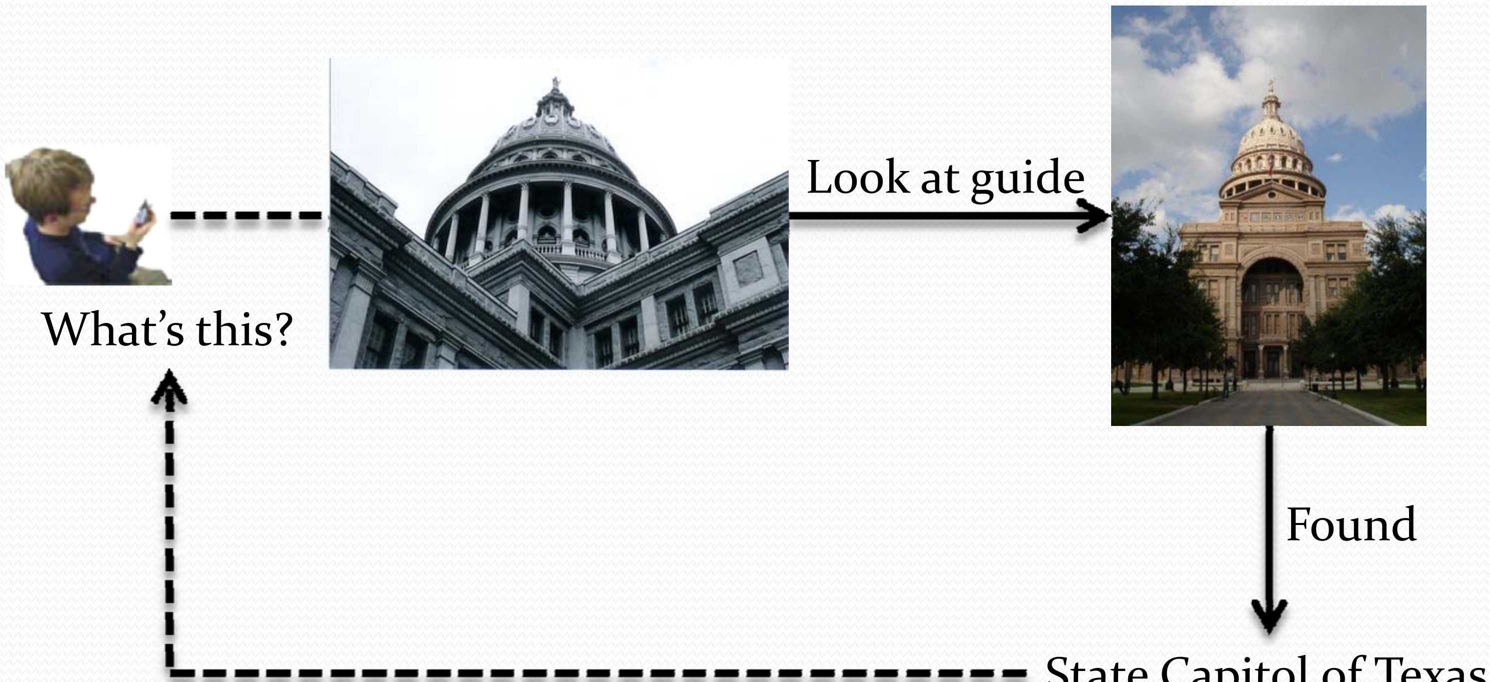
Introduction

- Suppose a stranger in downtown with a tour guide book



Austin, TX

Introduction



- Name of place
- Where is it?
- Where am I now?

The Localization Problem

- Ingemar Cox (1991):

  "Using sensory information to locate the robot in its environment is the most fundamental problem to provide a mobile robot with autonomous capabilities."

  - Position tracking (bounded uncertainty)
  - Global localization (unbounded uncertainty)
  - Kidnapping (recovery from failure)

Vision-based Localization

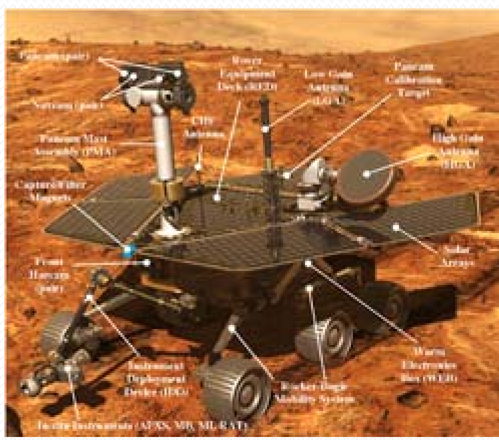
- Approaches
  - Place recognition using image retrieval
  - Appearance-based localization and mapping
    - SLAM (Simultaneous Localization and Mapping)
    - Kidnapped robot problem (global localization in known environment)

Why Visual Clues?

- Why are visual clues useful in these problems?
  - Cameras are low-cost sensors
  - that provide a huge amount of information.
  - Cameras are passive sensors that do not suffer from interferences.
  - Populated environments are full of visual clues that support localization (for their inhabitants).

Why Important?

- Application areas
  - Explorer robots (space, deep sea, mines)
  - Navigation
  - Military (missiles, vehicles without driver)




Outline

- Place recognition using image retrieval
  - Large-scale image search with textual keywords
  - Query expansion on location domains
- Vision-based localization and mapping
  - Robot localization in indoors environment
  - Vision-based SLAM and global localization
  - Location and orientation prediction with single image
- Conclusion
- Discussion points

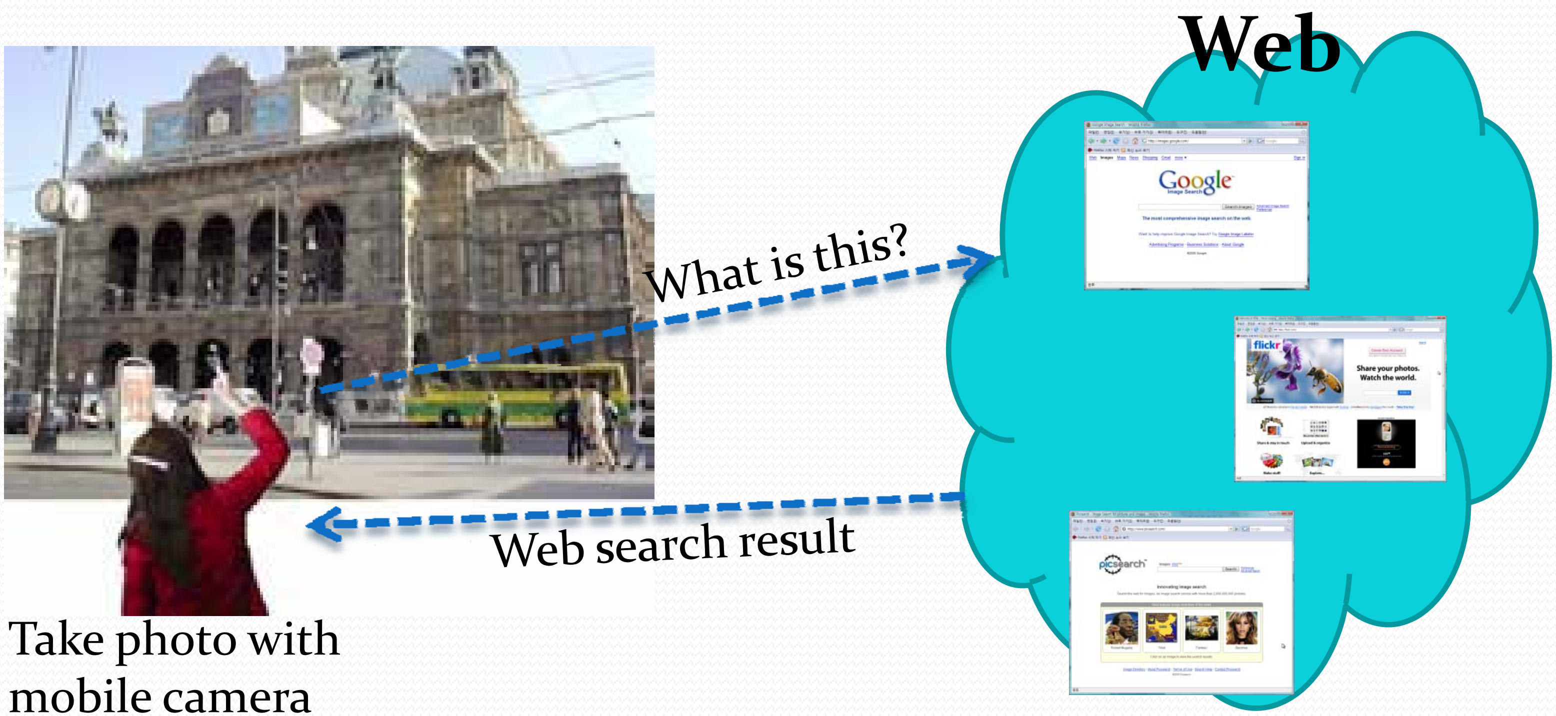
Place Recognition using Image Retrieval

- Large-scale image search with textual keywords
  - Searching the Web with Mobile Images for Location Recognition,
    - T. Yeh, K. Tollmar, and T. Darrell, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2004.

- Query expansion on location domains
  - Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval,
    - O. Chum, J. Philbin, J. Sivic, M. Isard, A. Zisserman, in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2007.

Large-Scale Image Search With Textual Keywords

- Searching web to get information about the location



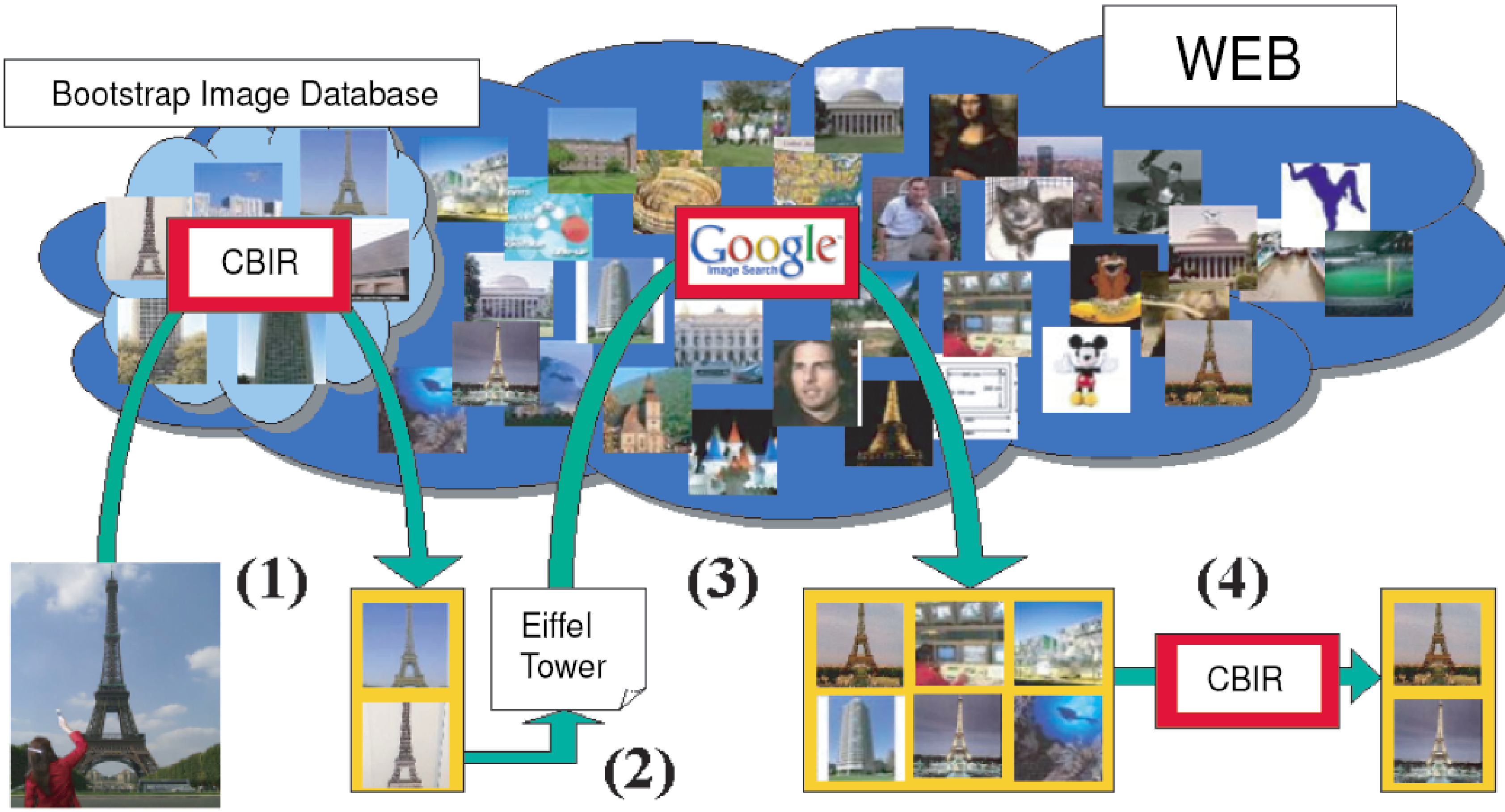
Take photo with mobile camera

[Searching the Web with Mobile Images for Location Recognition - T. Yeh, K. Tollmar, and T. Darrell, CVPR 2004]

Overview

- Recognize location using photos taken by mobile devices
- Bootstrap CBIR on small size dataset
- Perform keyword-based search over large-scale dataset

Overview



Bootstrap Image Database
WEB
CBIR
Google
Image Search
(1)
Eiffel
Tower
(2)
(3)
(4)
CBIR

Bootstrap Image-based Search

- Use small size of bootstrap image database
- Perform Content-Based Image Search over bootstrap database
- Two image matching metrics
  - Energy spectrum (windowed Fourier transform)

$$I(f_x, f_y) = \sum_{x,y=0}^{N-1} i(x,y)h(x,y)e^{-j2\pi(f_x x + f_y y))}$$

  - Steerable filter (wavelet decompositions)

$$m(x) = \sum_{x'} |\lambda(i)| \cdot w(x' - x) \quad \text{s.t.} \quad \lambda = G_{\theta_i}(S_j(I))$$

w: averaging window
G: steerable filter for $\frac{1}{3}k\pi \ (k = 1, 2, \ldots, 6)$
S: scaling operator

Extracting Textual Information

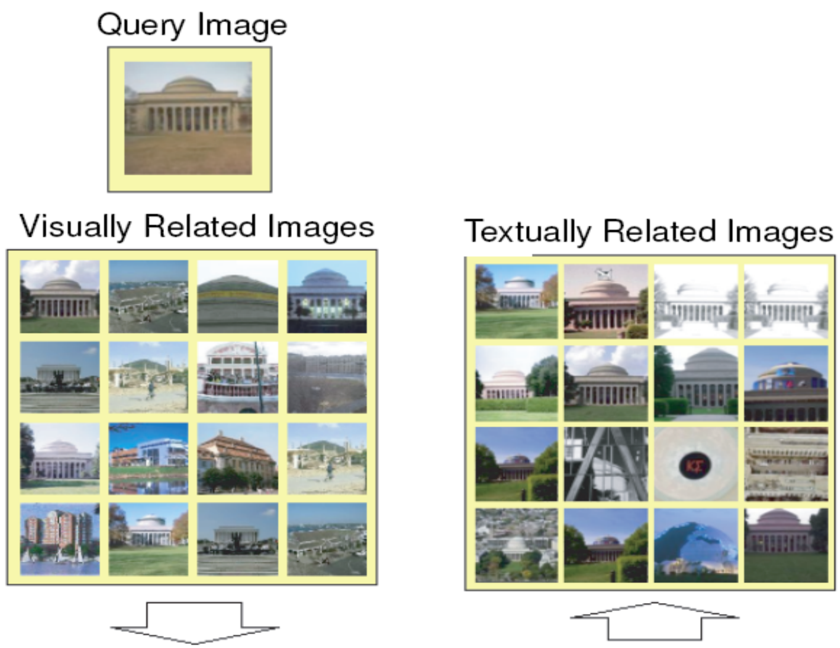
- Extract useful textual keyword to extend search

- Use TF-IDF (term frequency, inverse document frequency) metric
  - $\mu(w) = \dfrac{df(w)}{tf(w)}$
  - Top *n* word combinations are used

Content-filtered Keyword Search

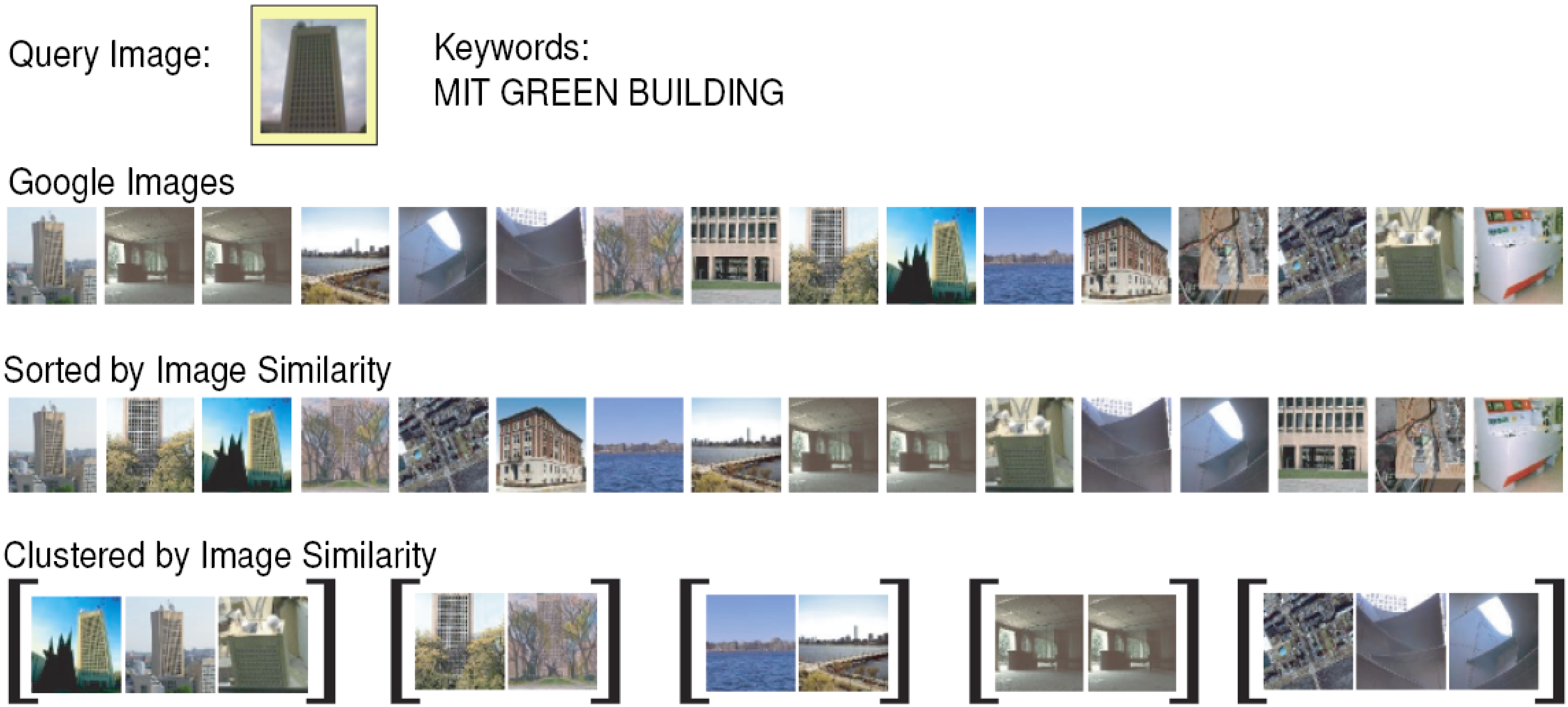
- Filter keyword search results to get visually-relevant result
- Two possible results for the keyword search
  1) $I_q \underset{image}{\longleftrightarrow} I_v, P_v \underset{text}{\longleftrightarrow} P_t \Longrightarrow I_q \longleftrightarrow P_t$
  2) $I_q \underset{image}{\longleftrightarrow} I_v, P_v \underset{text}{\longleftrightarrow} I_t \Longrightarrow I_q \longleftrightarrow I_t$
- Apply visual similarity to case 2) results and filter them
- Perform bottom-up clustering to the result to see meaningful results

An Example Search Scenario



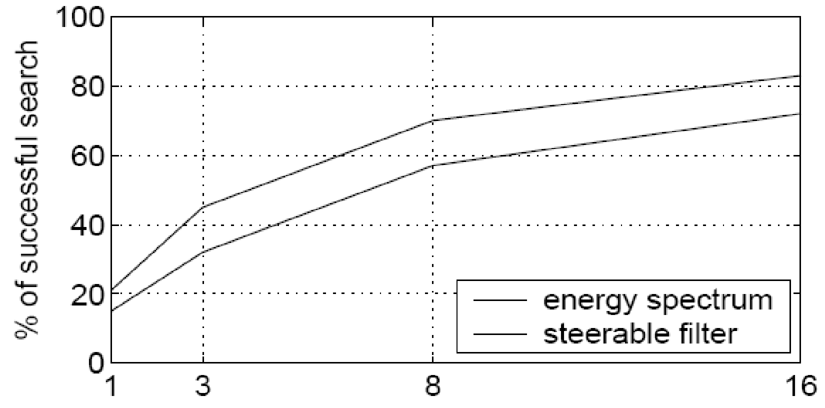
Unigram	μ	Bigram	μ
MIT	128	foundation relations	3
story	59	MIT dome	2
engineering	33	da lucha	2
kruckmeyer	29	view realvideo	2
boston	28	cancer research	2

Content-filtering Example



Experiments

- Bootstrap database
  - 2000+ web-crawled landmark images from *mit.edu*
- Query images
  - Take 100 images using Nokia 3650 camera phone
- Result



k nearest neighbors

Summary

- Web search for place recognition using mobile images
- Hybrid image-and-keyword search over real-world database
- Find both visually and textually relevant images

Query Expansion on Location Domains

- Objective
  - Retrieve visual objects (Oxford buildings in this case) in a large image database

- Approach
  - Query expansion
    - Use highly ranked query results as new query
    - Expand the initial query with richer query results

[Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval,
- O. Chum, J. Philbin, J. Sivic, M. Isard, A. Zisserman, ICCV 2007]

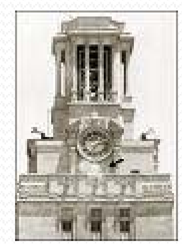
Query Expansion

- Query expansion
  - Reformulate seed query to improve retrieval performance

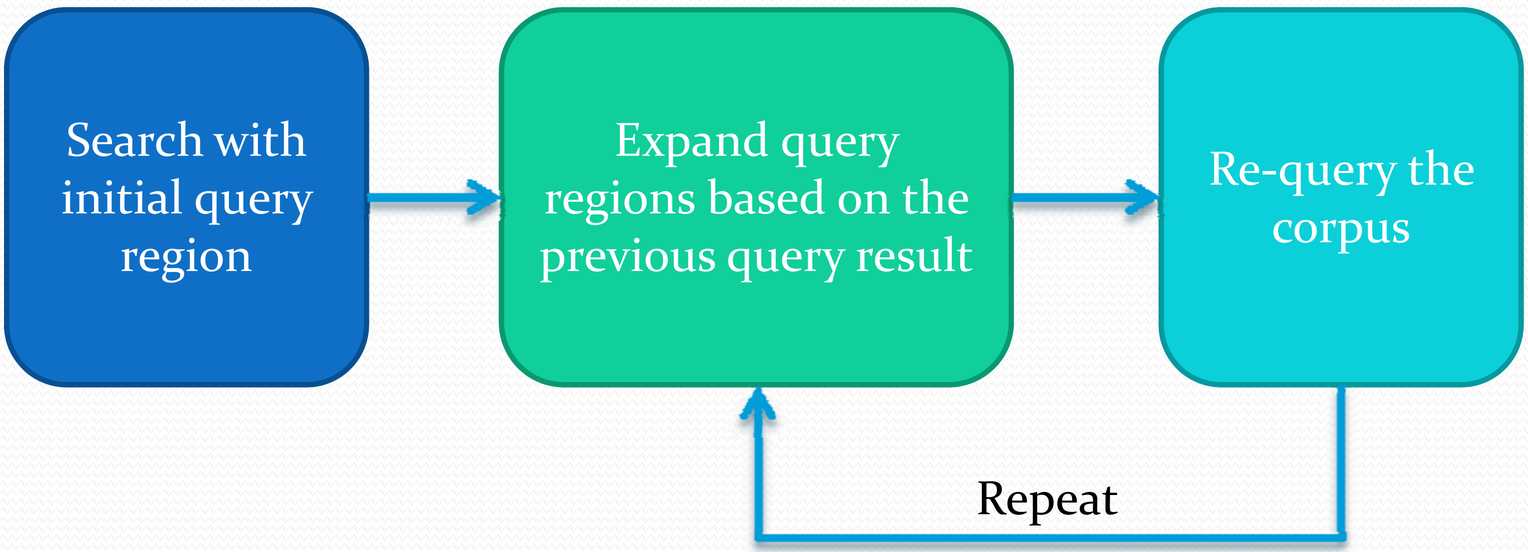
- Text query expansion
  - Manchester United ↔ Man Utd, EPL, Cristiano Ronaldo, Ryan Giggs
- Image query expansion

↔

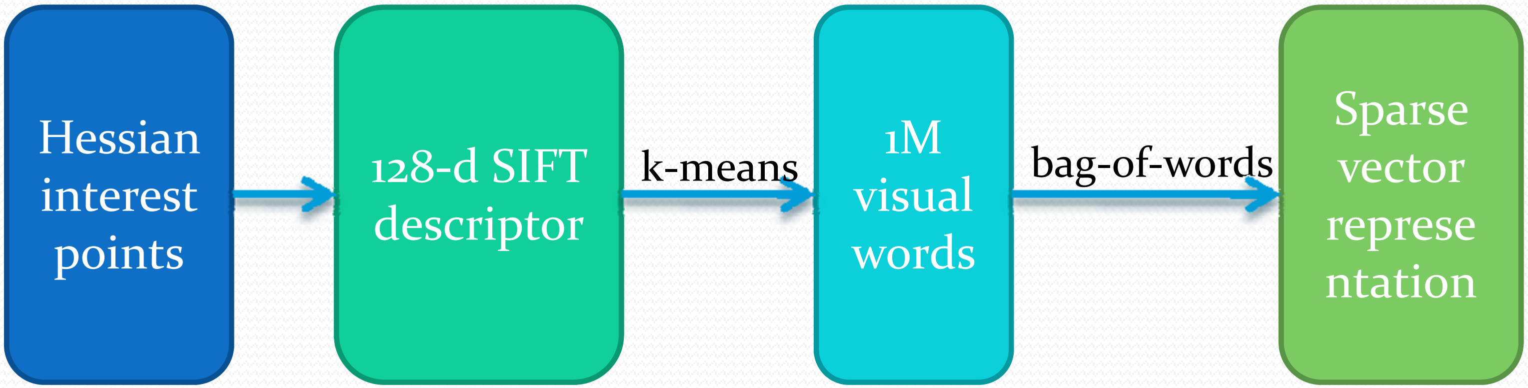





Approach Overview



Data Representation



Spatial Verification

- Verify query results to find spatially-relevant images
- Use affine invariant semi-local region associated with each interest point
- Perform RANSAC-like scoring mechanism
- Select the best hypothesis (isotropic scale & translation) based on the number of inliers

Query Expansion Model

- Query expansion baseline
  - Requery with average frequency vectors of top m=5 results
- Transitive closure expansion
  - Requery with the previous query result
  - Find the transitive closure of query result
- Average query expansion
  - New query performed with averaged frequency vector
  - Use matching regions for the original query region

$$d_{avg} = \frac{1}{m+1}\left(d_0 + \sum_{i=1}^{m} d_i\right)$$ (m < 50)

Query Expansion Model

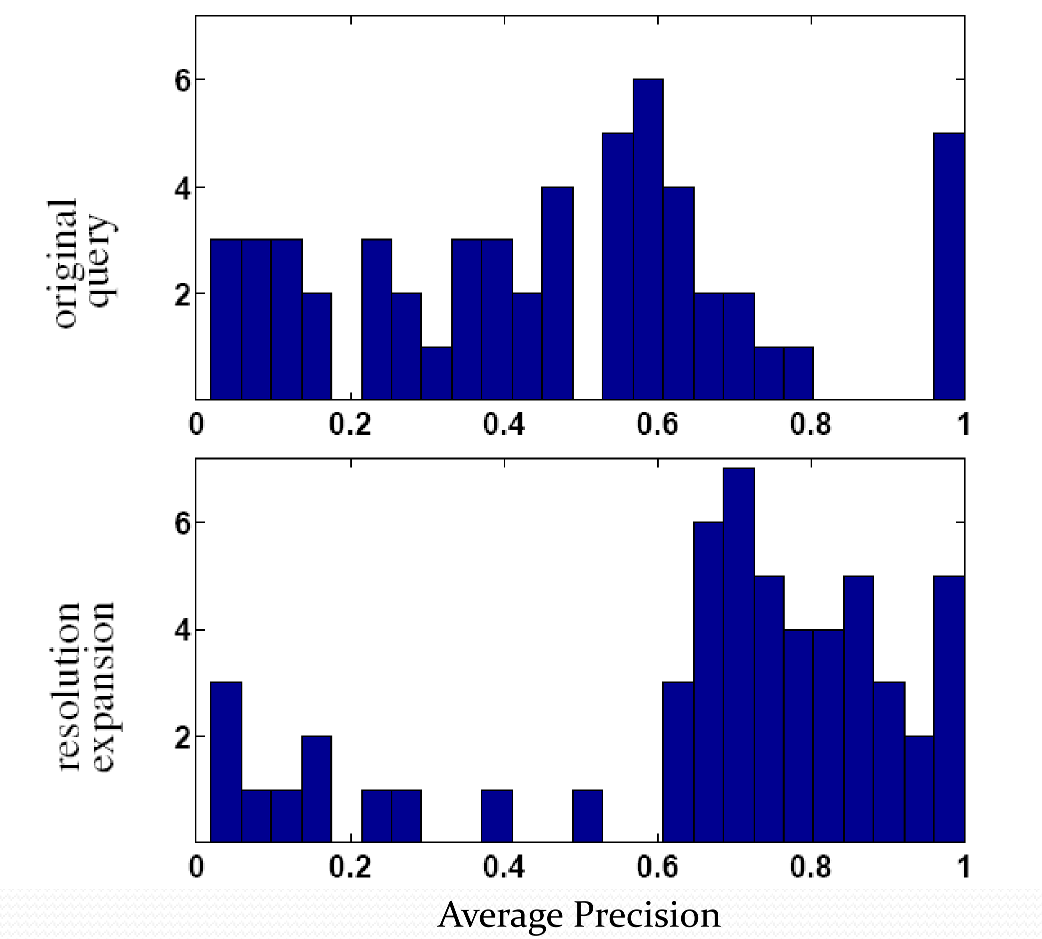
- Recursive average query expansion
  - Generate average query recursively with previously verified results
  - Ends when verified results > 30 or no new result found
- Multiple image resolution expansion
  - Categorize query results into three different resolution scale bands (0, 4/5), (2/3, 3/2), (5/4, ∞) according to median scale image
  - Reconstruct average images from each scale band

Results

	Ground truth		*Oxford + Flickr1* dataset						*Oxford + Flickr1 + Flickr2* dataset					
	OK	Junk	ori	qeb	trc	avg	rec	sca	ori	qeb	trc	avg	rec	sca
All Souls	78	111	41.9	49.7	85.0	76.1	85.9	**94.1**	32.8	36.9	80.5	66.3	73.9	**84.9**
Ashmolean	25	31	53.8	35.4	51.4	66.4	74.6	**75.7**	41.8	25.9	45.4	57.6	**68.2**	65.5
Balliol	12	18	50.4	52.4	44.2	63.9	**74.5**	71.2	40.1	39.4	39.6	55.5	**67.6**	60.0
Bodleian	24	30	42.3	47.4	49.3	**57.6**	48.6	53.3	32.3	36.9	43.5	**46.8**	43.8	44.9
Christ Church	78	133	53.7	36.3	56.2	63.1	**63.3**	63.1	52.6	18.9	55.2	**61.0**	57.4	57.7
Cornmarket	9	13	54.1	60.4	58.2	74.7	74.9	**83.1**	42.2	53.4	56.0	65.2	68.1	**74.9**
Hertford	24	31	69.8	74.4	77.4	89.9	90.3	**97.9**	64.7	70.7	75.8	87.7	87.7	**94.9**
Keble	7	11	79.3	59.6	64.1	90.2	**100**	97.2	55.0	15.6	57.3	**67.4**	65.8	65.0
Magdalen	54	103	9.5	6.9	25.2	28.3	**41.5**	33.2	5.4	0.2	16.9	15.7	**31.3**	26.1
Pitt Rivers	7	9	**100**	**100**	**100**	**100**	**100**	**100**	**100**	90.2	**100**	**100**	**100**	**100**
Radcliffe Cam.	221	348	50.5	59.7	88.0	71.3	73.4	**91.9**	44.2	56.8	86.8	70.5	72.5	**91.3**
Total	539	838	55.0	52.9	63.5	71.1	75.2	**78.2**	46.5	40.5	59.7	63.1	67.0	**69.6**

- Dataset: Oxford building dataset (5K images)
- Flickr1: 100K unlabeled dataset
- Flickr2: 1M unlabeled dataset

Results



Histogram of average precision for 55 queries

Example Query Result

Summary

- Use query expansion in place recognition domain
- Works well in a large scale database
- Query-expanded result are better than original base query

Outline

- Place recognition using image retrieval
  - Large-scale image search with textual keywords
  - Query expansion on location domains
- Vision-based localization and mapping
  - Robot localization in indoors environment
  - Vision-based SLAM and global localization
  - Location and orientation prediction with single image
- Conclusion
- Discussion points

Vision-based localization and mapping

- Robot localization in indoors environment
  - Qualitative Image Based Localization in Indoors Environments, by J. Kosecka, L. Zhou, P. Barber, and Z. Duric, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2003.
  - Location Recognition and Global Localization Based on Scale-Invariant Keypoints, by J. Kosecka and X. Yang, CVPR workshop 2004.
- Vision-based SLAM and global localization
  - Vision-based Mobile Robot Localization and Mapping Using Scale-Invariant Features, by Se, S. and Lowe, D. and Little, J. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2001.
  - Vision-Based Global Localization and Mapping for Mobile Robots, Se, S., Lowe, D., & Little, J. IEEE Transactions on Robotics, 2005.
  - Image-Based Localisation, R. Cipolla, D. Robertson and B. Tordoff. Proceedings of the10th International Conference on Virtual Systems and Multimedia, 2004.

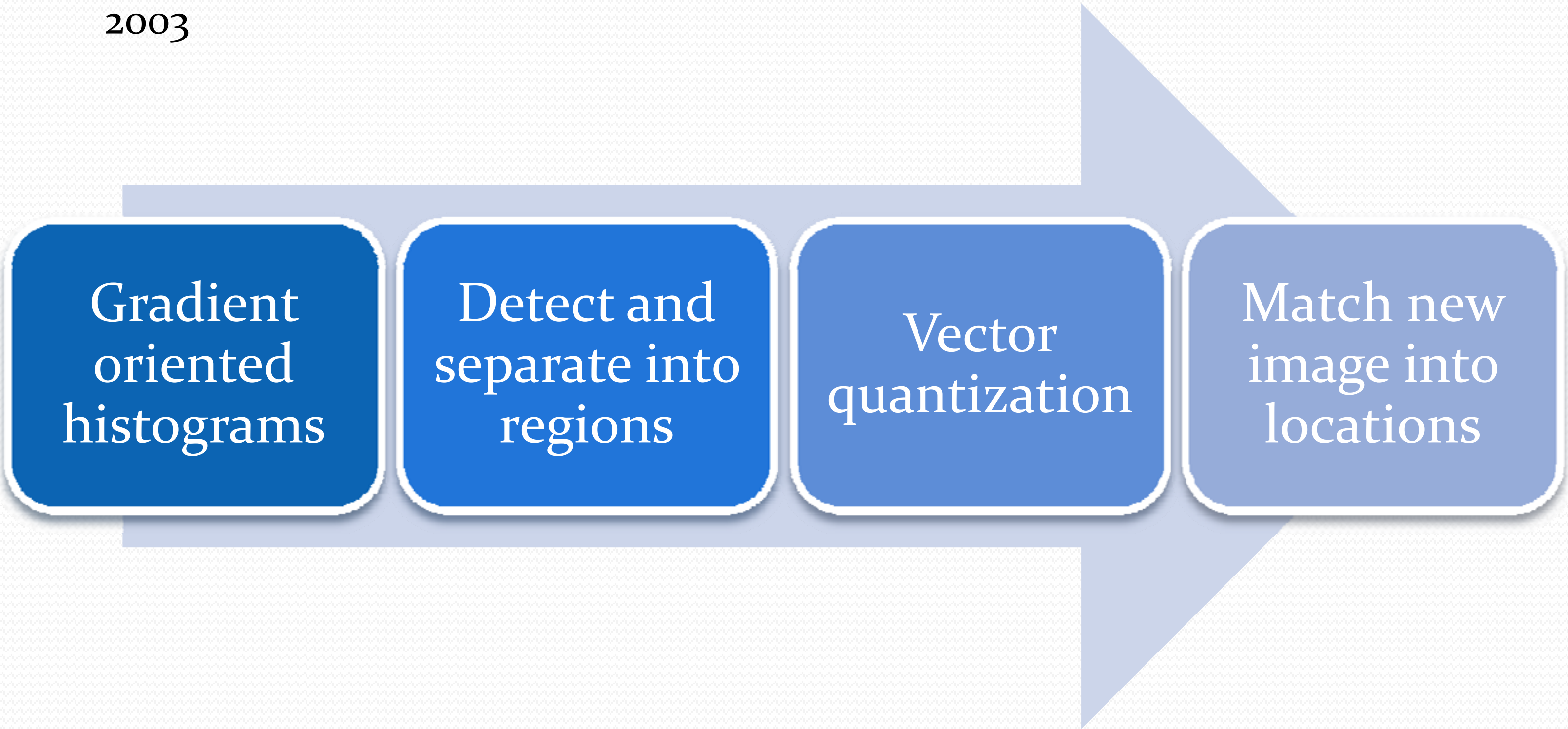
Robot Localization in Indoors Environment

- Objective
  - Global localization by means of location recognition using only visual appearances
  - Infer a topological model of indoor environment
  - Classify current location with single image

- Approach
  - Divide each location automatically by sudden changes of features
  - Use SIFT features to represent each location
  - Use HMM model to exploit location neighborhood relationships

Overview

- One approach for robot localization
  - Qualitative Image Based Localization in Indoors Environments, Kosecka et al. CVPR 2003



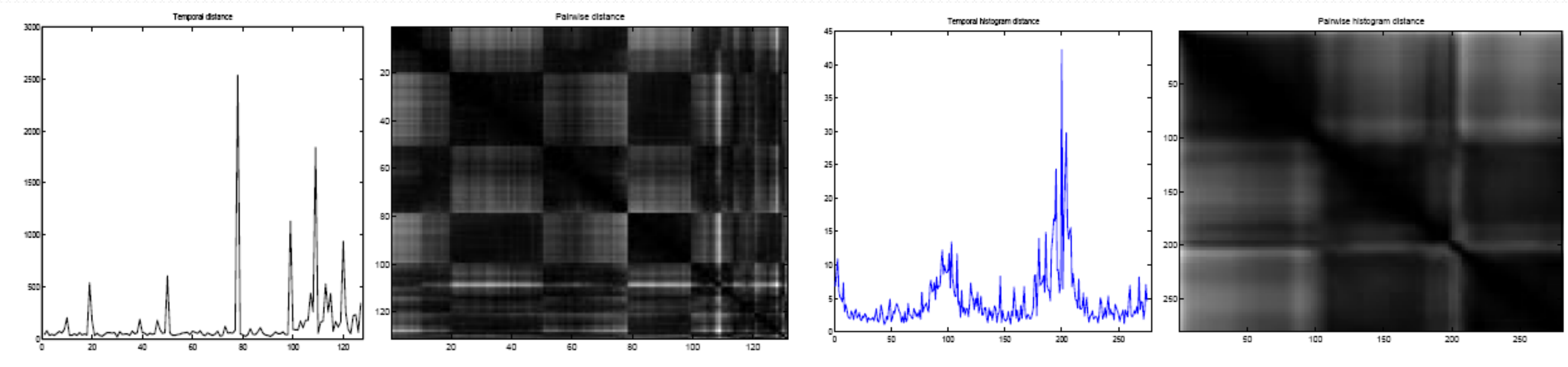
Measurement Phase

- Gradient orientation histogram
  - Distinctive feature of location tolerant to changes of lighting
  - Properly reflect change of location
- Feature comparison metric
  - χ^2 distance measure

$$\chi^2(h_i, h_j) = \sum_k \frac{(h_i(k) - h_j(k))^2}{h_i(k) + h_j(k)}$$

Measurement Phase

- Shows clear distinction between different regions



Still images

Videos

[Comparison of orientation histograms]

Learning Phase

- Automatic label assignment

Search for peaks in histogram distance → Separate into different locations

- Get prototype vectors
  - Represent each class
  - Learning Vector Quantization (LVQ)
    - Iterative approach to get codebook vectors

$$\mathbf{m}_c(t+1) = \mathbf{m}_c(t) \pm \alpha(t)(\mathbf{x}_i - \mathbf{m}_c(t))$$

($m_c(t)$: closest codebook vector to input x_i)

Recognition Phase

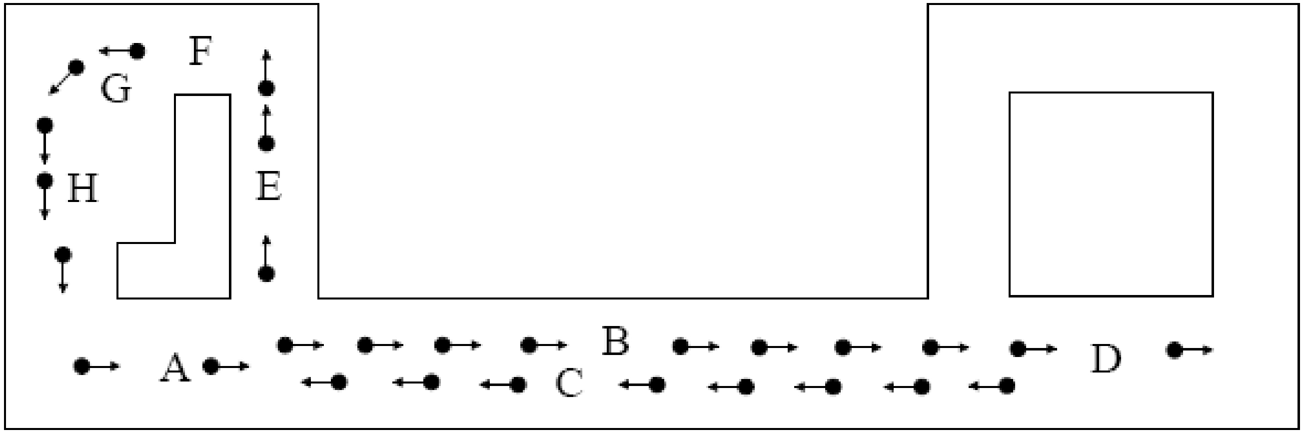
- Given a new image,

Get histogram h → Compare with prototype vectors → Get two nearest neighbors belong to different classes

- Confidence level of classification
  - $C_\chi = \dfrac{\chi^2(h, h_{2^{nd}})}{\chi^2(h, h_{1^{st}})}$
  - When C_χ is low, perform sub-image comparison

Experiments

- Datasets
  - 185 images taken along 4th floor corridor
  - Video sequence taken by mobile robot



Result

Prototype vectors for each location

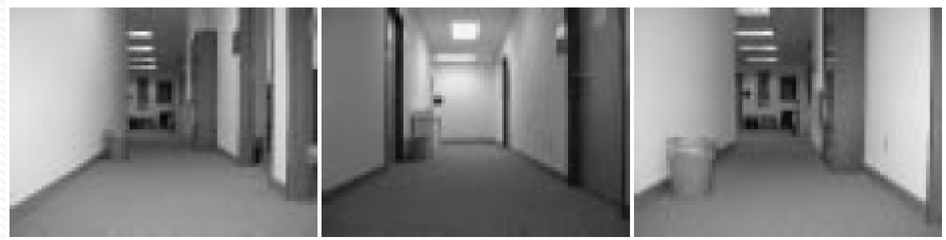
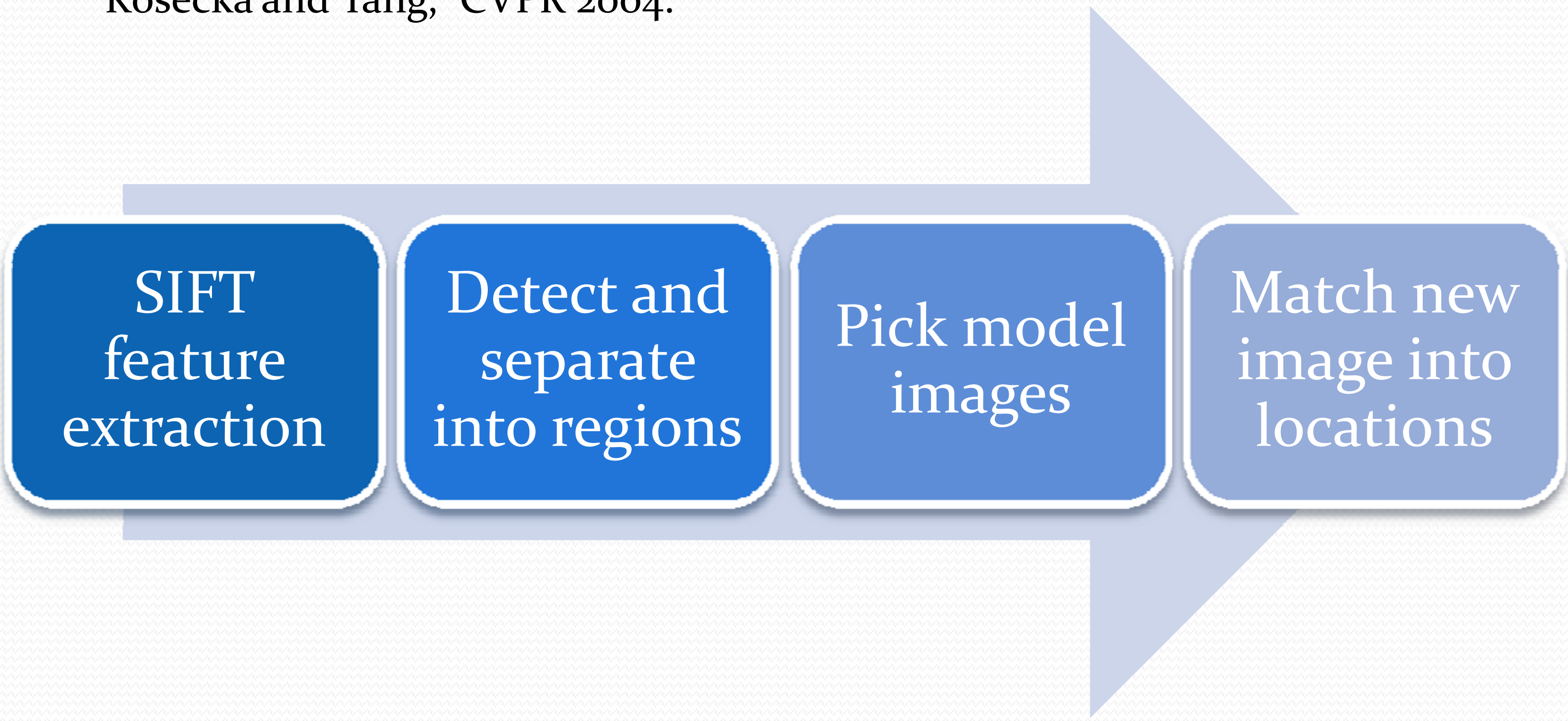


Figure 5. Example of an image from location F (left), misclassified as one from location E (middle) and then re-classified correctly as F (right) using sub-image comparison.

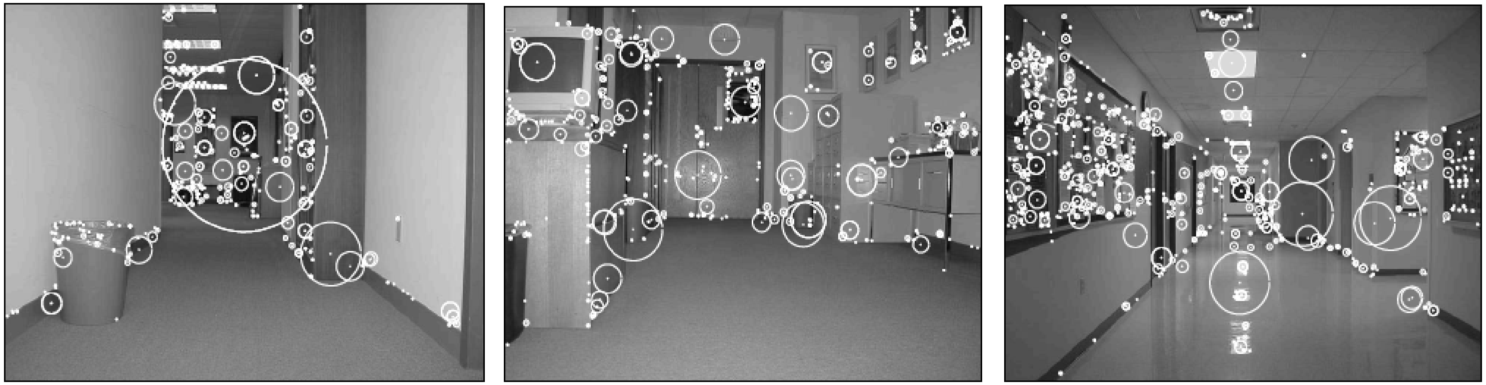
Overview

- Different approach on same problem
  - Location Recognition and Global Localization Based on Scale-Invariant Keypoints, Kosecka and Yang, CVPR 2004.



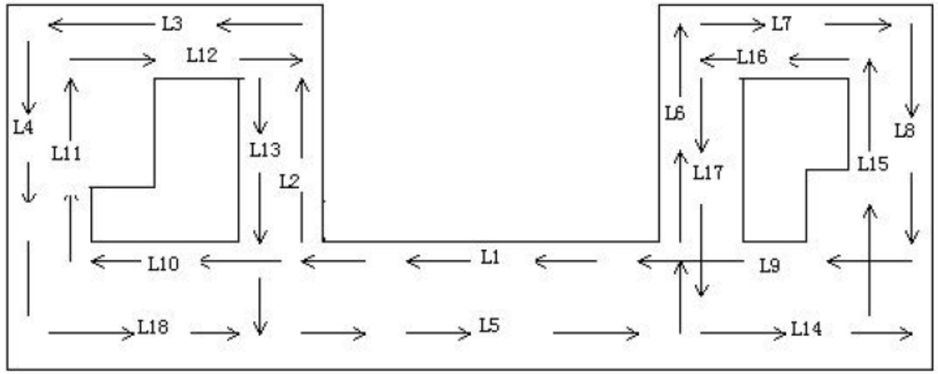
Feature Extraction

- SIFT features
  - Invariant to scale, rotation, and affine transformation
  - $$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$
    $$= L(x, y, k\sigma) - L(x, y, \sigma).$$



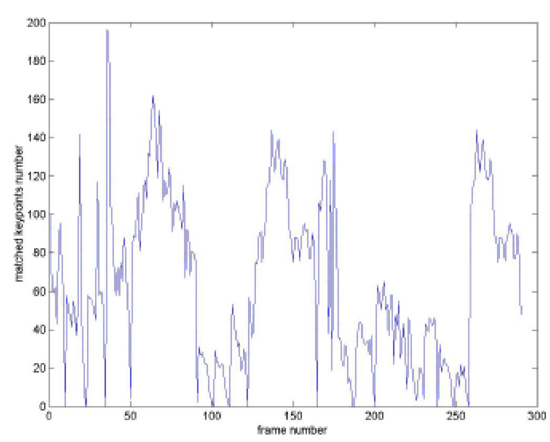
Environment Model

- Dataset
  - Photos taken along the corridor of 4th floor
  - Images were taken in every 2-3 meters
  - Whole sequence divided into 18 locations
  - Move only 4 possible directions (N, S, W, E)

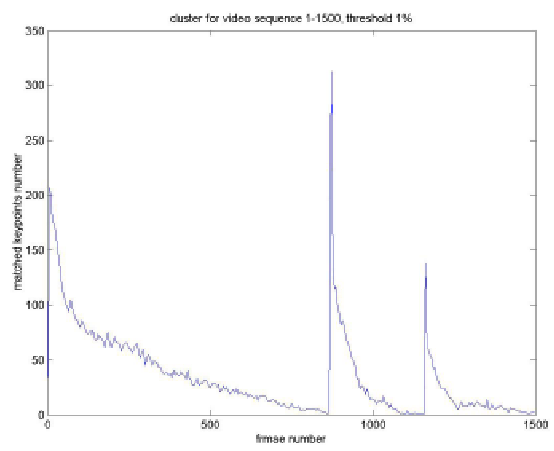


Environment Model

- Detecting transitions between locations
  - Sudden change of location appearances
  - Detect when the number of matching features between successive frames is low

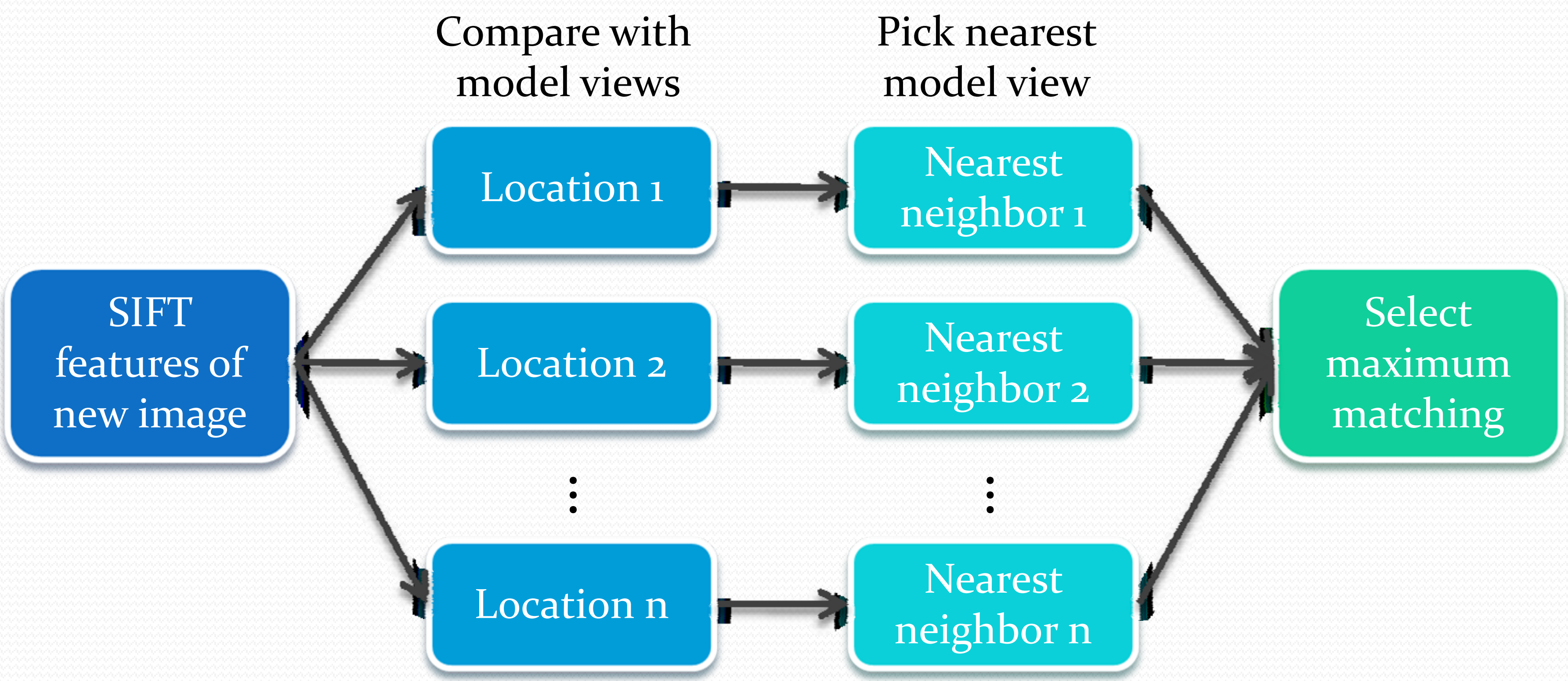


Matching keypoints between consecutive images (still images)



Matching keypoints between first and current frames (video)

Location Recognition

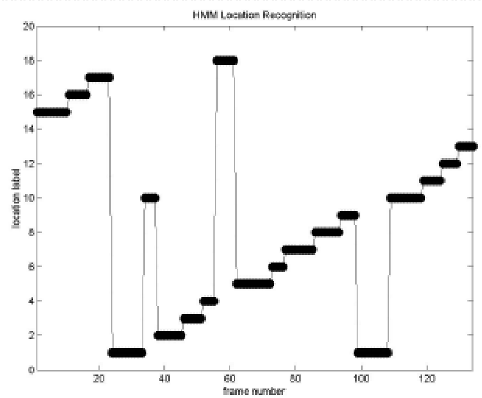


Spatial Relationship Model

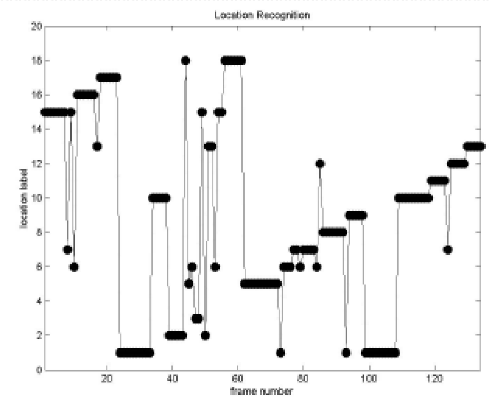
- Problem of previous scheme
  - Vulnerable to dynamic changes of environment

- Model spatial relationship with HMM
  - $P(L_t = l_i | o_{1\ldots t}) \propto P(o_t | L_t = l_i) P(L_t = l_i | o_{1:t-1})$
  - $p(o_t | L_t = l_i) = \dfrac{C(i)}{\sum_j C(j)}$
  - $P(L_t = l_i | o_{1:t-1}) = \sum^{N} A(i, j) P(L_{t-1} = l_j | o_{1:t-1})$
  - where $A(i, j) = P(L_t = l_i | L_t = l_j)$

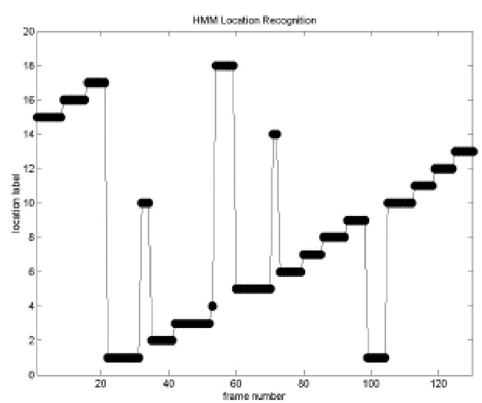
Result with Spatial HMM



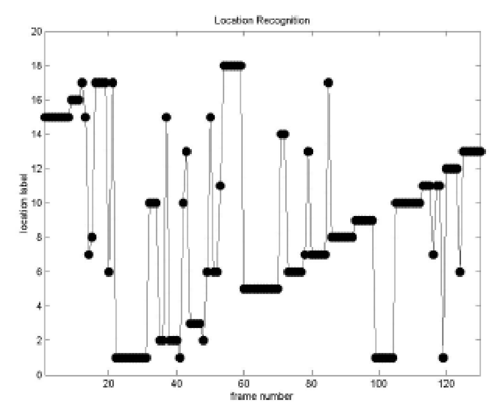
a) Sequence 2 with HMM



b) Sequence 2 without HMM



c) Sequence 3 with HMM



d) Sequence 3 without HMM

Summary

- Simple appearance-based location recognition and global localization
- Simple discrimination technique
  - Compare with χ^2 distance measure with gradient orientation histogram
  - Compare scale-invariant SIFT features
- Infer topological model of indoor environment
- Exploit spatial relationship model by HMM

Vision-based SLAM and Global Localization

- Objective
  - Simultaneous localization and map building using only visual appearances
  - Global localization without any prior location estimate

- Outline
  - Simultaneous localization and mapping
  - Global localization
  - Submap alignment
  - Closing the loop

Vision-based SLAM and Global Localization

- Reference papers
  - Vision-based Mobile Robot Localization and Mapping Using Scale-Invariant Features, Se et al. ICRA 2001.
  - Vision-based Global Localization and Mapping for Mobile Robots, Se et al. IEEE Transactions on Robotics, 2005.

Background: SLAM

- Simultaneous Localization And Mapping

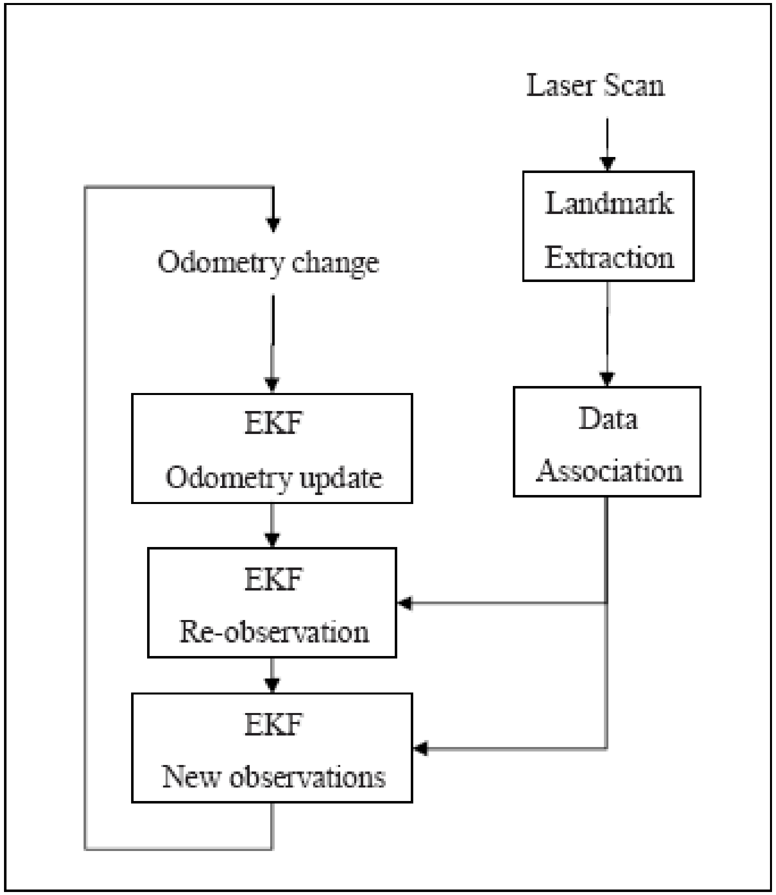
"**SLAM** is concerned with the problem of:

- building a map of an unknown environment by a mobile robot while at the same time
- navigating the environment using the map."

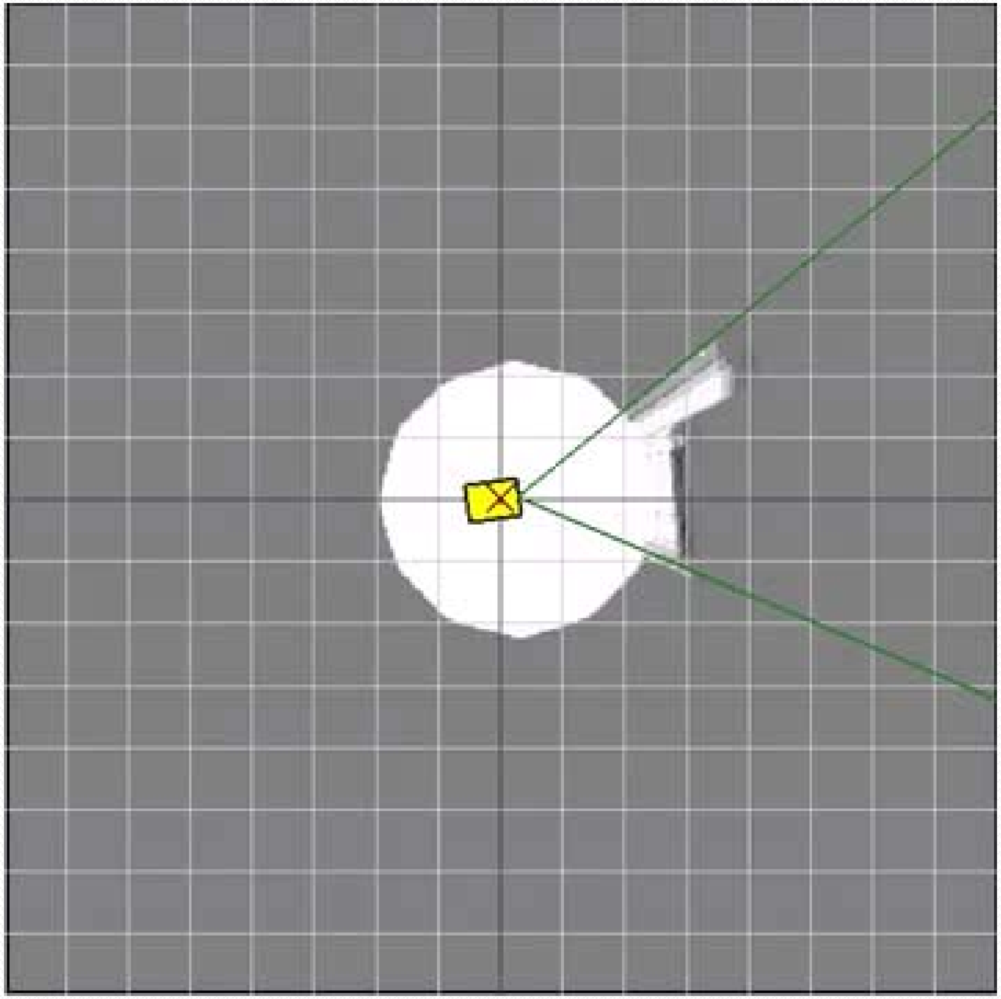
Background: SLAM

- Landmark Extraction
- Data Association
- State Estimation
- State Update & Landmark Update

Kalman Filter



Video: SLAM



Overview of SLAM Process

- SLAM process

Extract SIFT Features

Stereo Vision
- Extract 3D location for each feature

Predict
- Track features using odometry

Update
- Localize using least-squares

SIFT Features

Top Camera (193 Features)

3 images at one time frame

Size of square – Scale

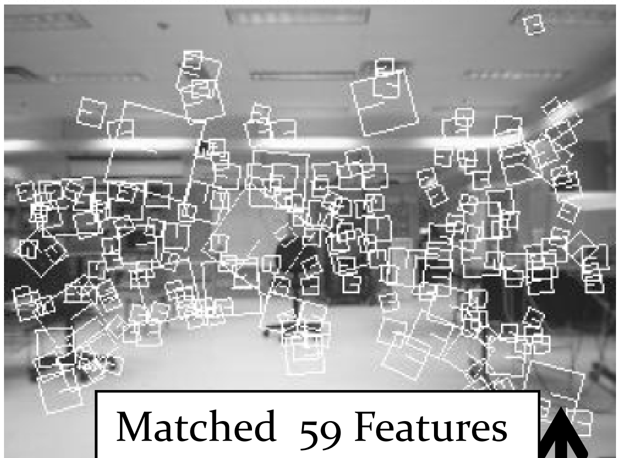
Line in square – Orientation

Bottom Left Camera (166 Features)

Bottom Right Camera (189 Features)

Stereo Vision

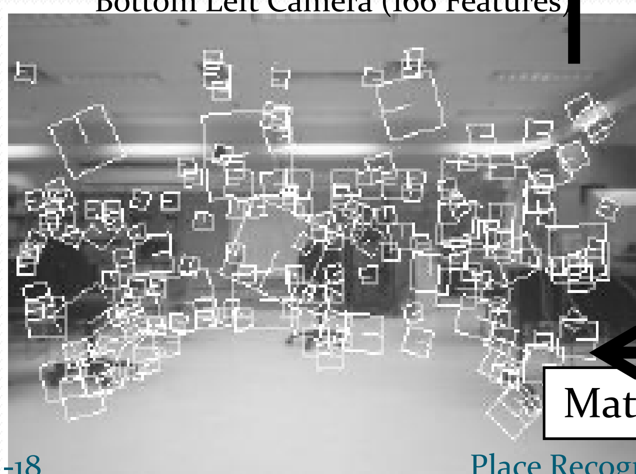
Top Camera (193 Features)



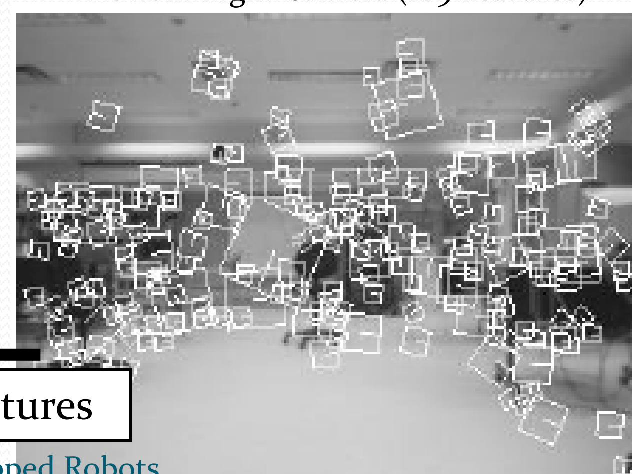
Find Disparity of SIFT features only

Use 3rd camera for verification (noise reduction)

Bottom Left Camera (166 Features)

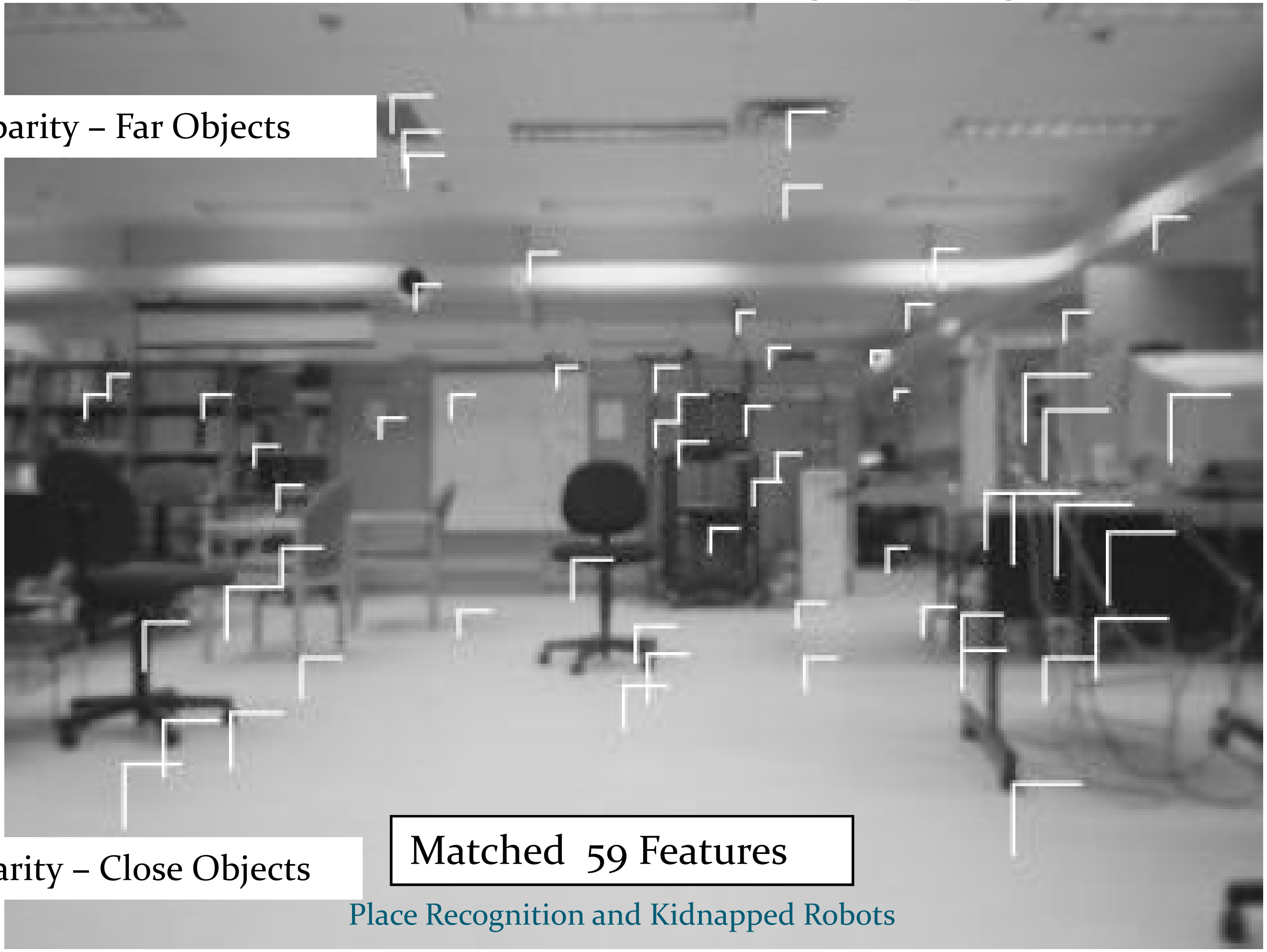


Bottom Right Camera (189 Features)



Stereo Vision

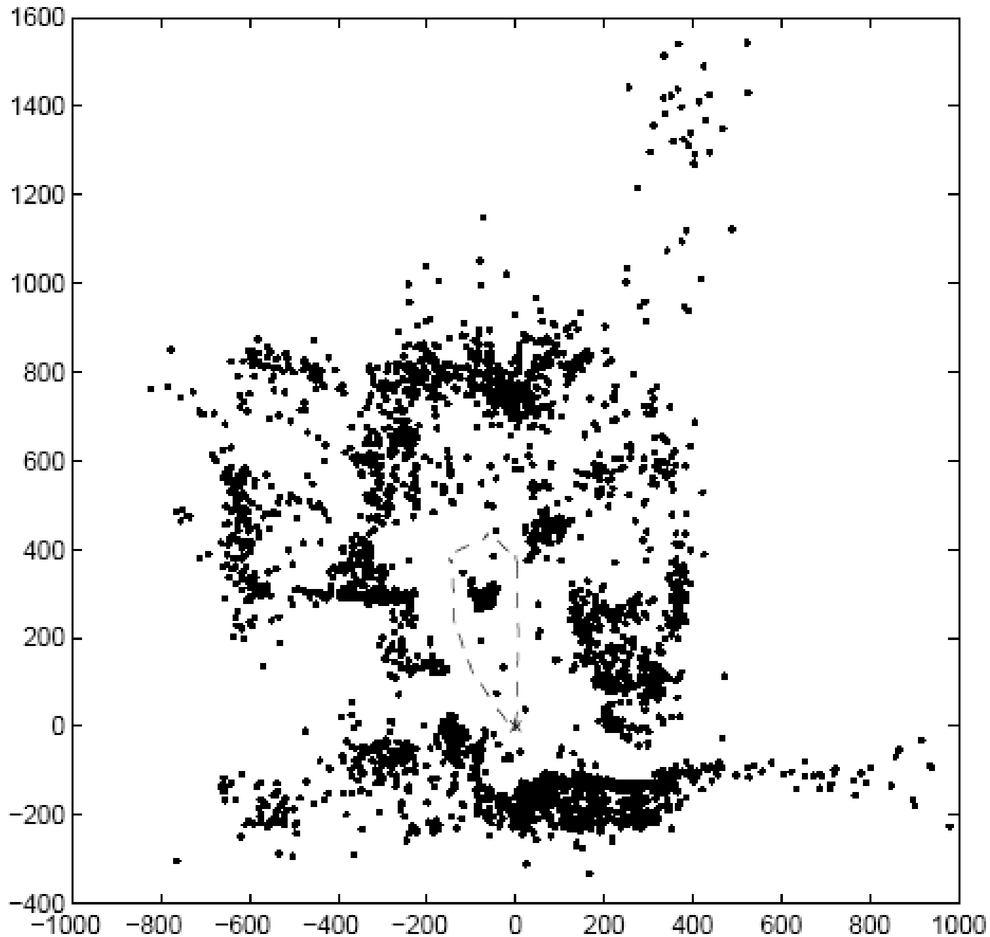
3D locations of each feature by Disparity



Map Building

- Match consecutive frames to predict robot motion
  - Use odometry to narrow down the search area
- Get more accurate matches using least-squares
- Track SIFT landmarks
- Build 3D map

Map Building Result



249 Frames

3590 Landmarks

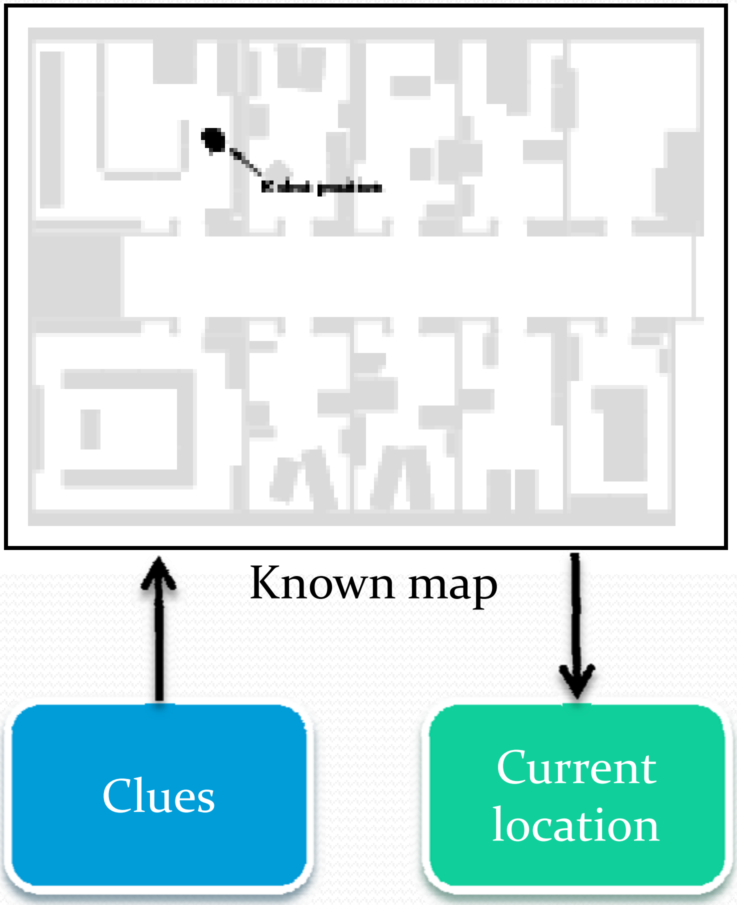
4m trajectory around room

Max Speeds:

- 40cm/sec = 0.89 mi/hr
- 10°/sec

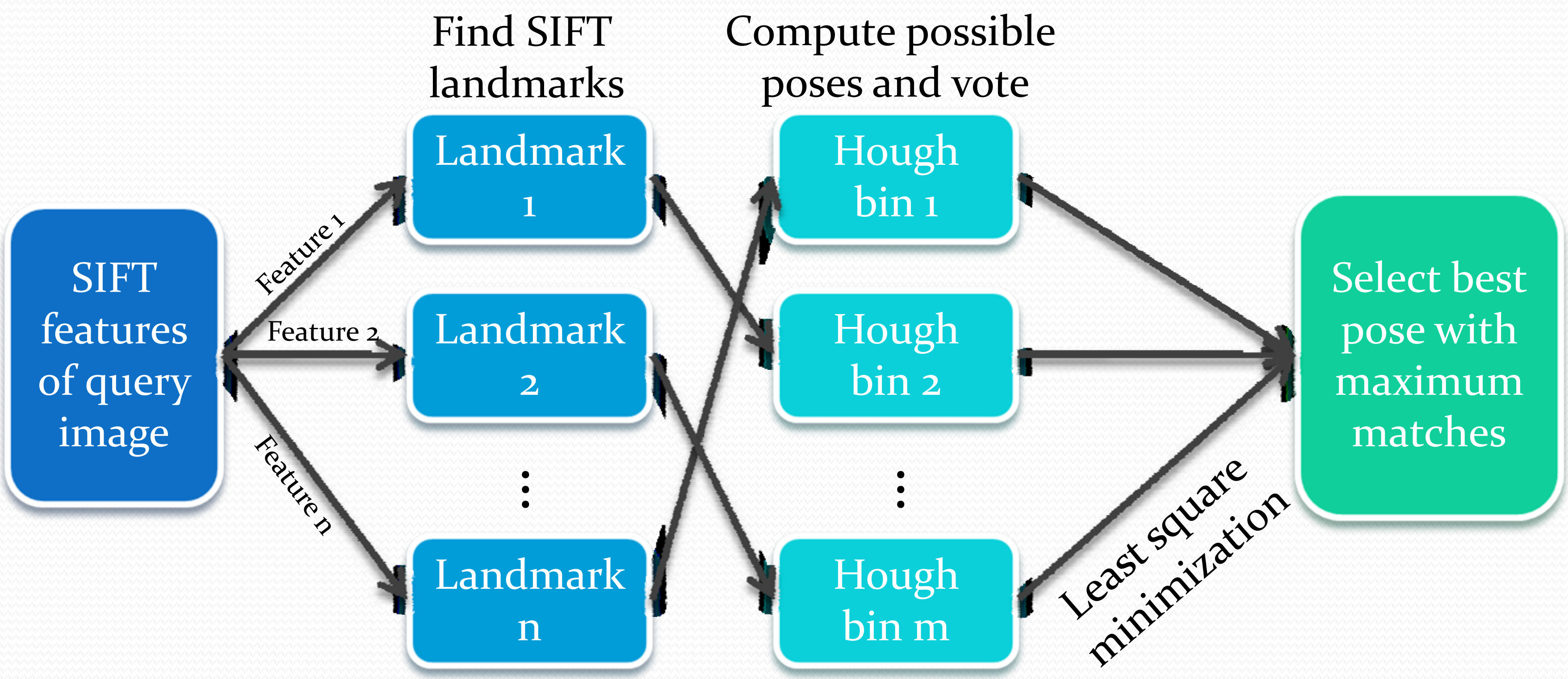
Global Localization

- Given known environment and the current view, find robot's location in the environment
- Two approaches of finding best matching location
  - Hough transform
  - RANSAC



Hough Transform Approach

- Find best 3D transformation (X, Z, θ)



RANSAC Approach

- Tentative matches
  - Compare each feature with landmarks in database
- Computing the alignment
  - Find align parameter (X, Z, θ)

$$\theta = \tan^{-1} \frac{BC - AD}{AC + BD}$$

where $A = X'_i - X'_j, B = Z'_i - Z'_j, C = X_i - X_j, D = Z_i - Z_j$

  - (X_i, Y_i, Z_i) : landmark position
  - (X_i', Y_i', Z_i') : feature position of current frame

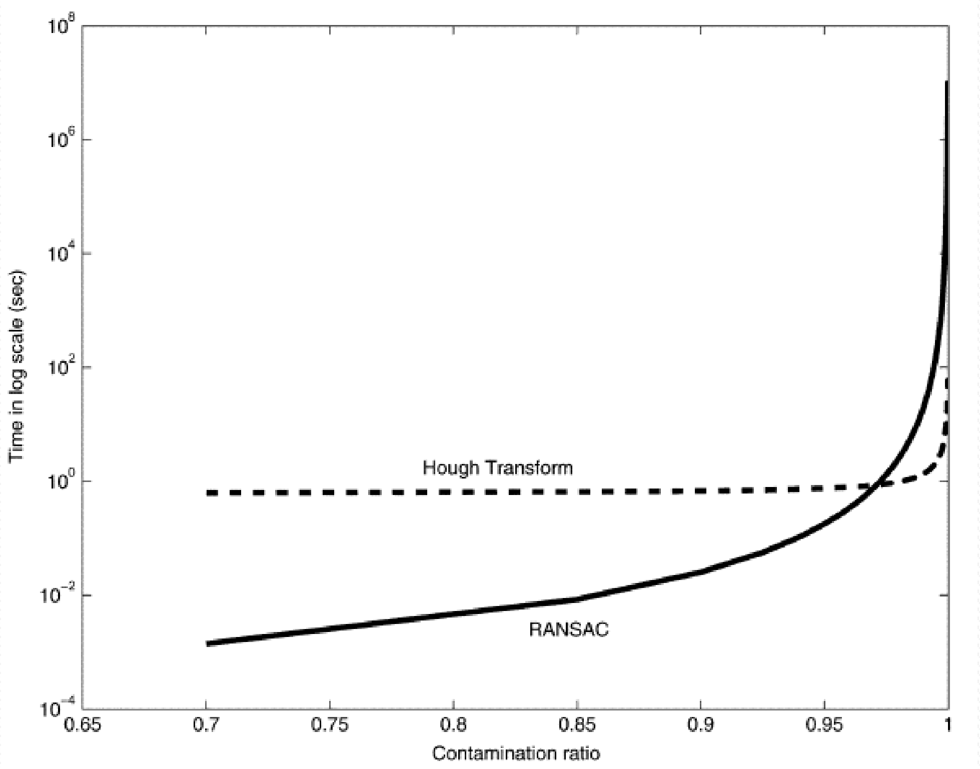
RANSAC Approach

- Seeking support
  - Check all tentative matches which support the particular pose (X, Z, θ)

- Find best hypothesis
  - Previous steps repeated m times
  - Find the hypothesis with the most support
  - Iterate least-squares minimization to find the most accurate pose estimate

Result

- Execution efficiecy
  - With SIFT features
    - RANSAC > Hough transform
  - With nonspecific features
    - RANSAC < Hough transform

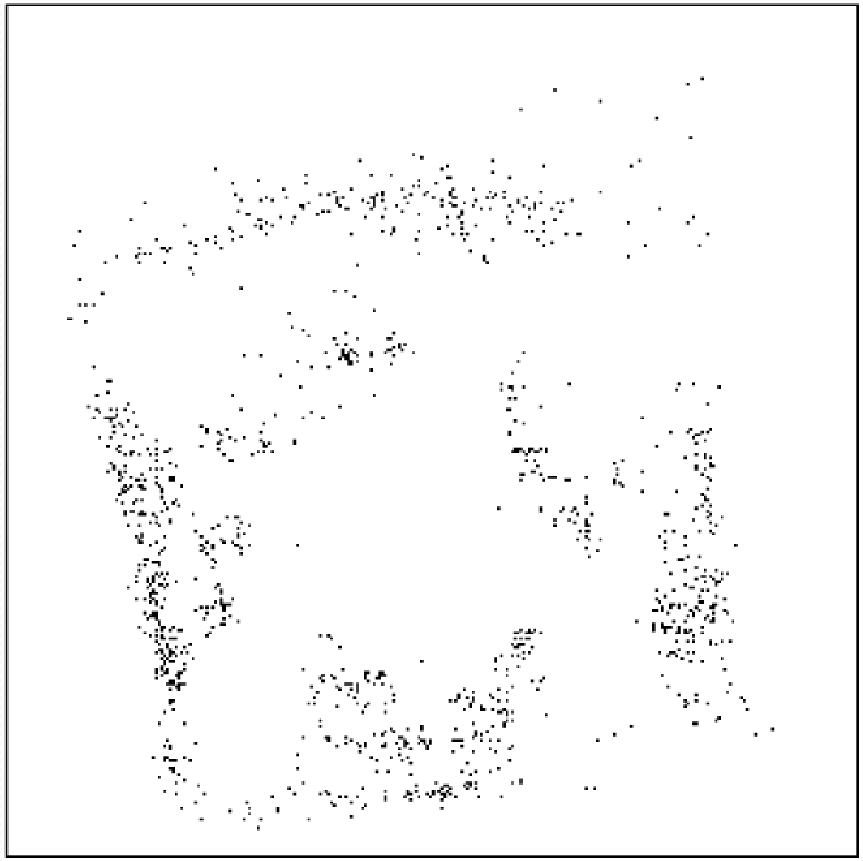


Map Alignment

- Just one frame might not be enough to localize
- Build small submap and match with global map already generated
- Using RANSAC to match SIFT features from both maps

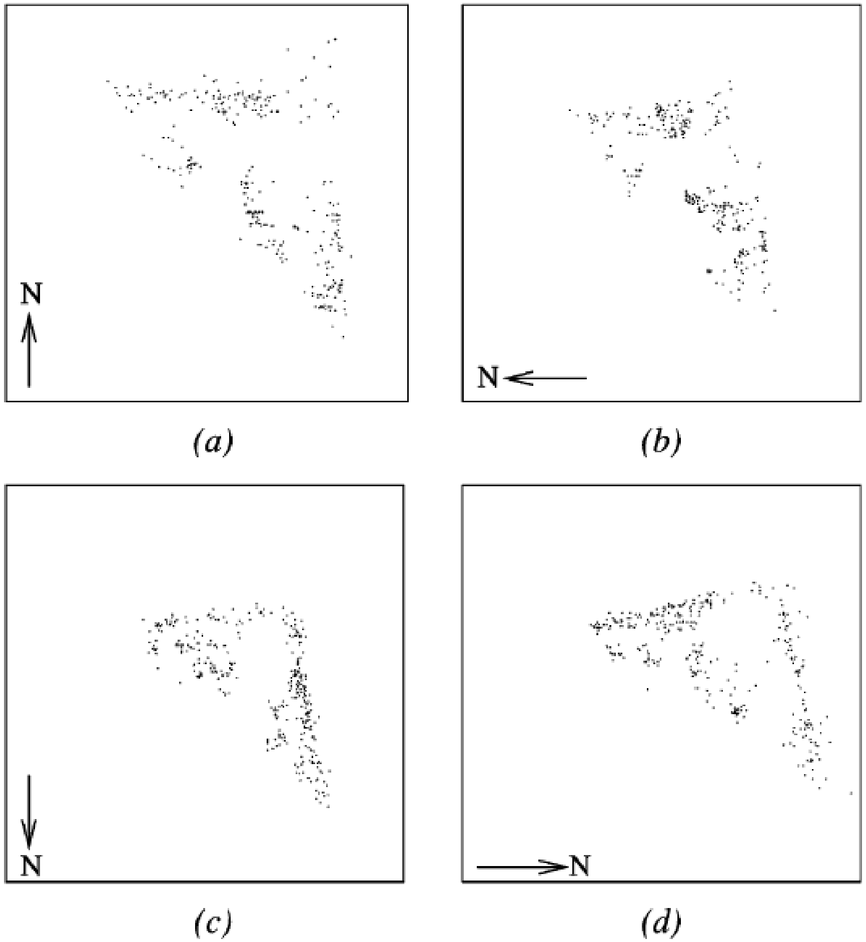
Problem on Map Construction

- Problem on large map construction over time
  - Due to occlusion and clutters, it often leads to significant errors



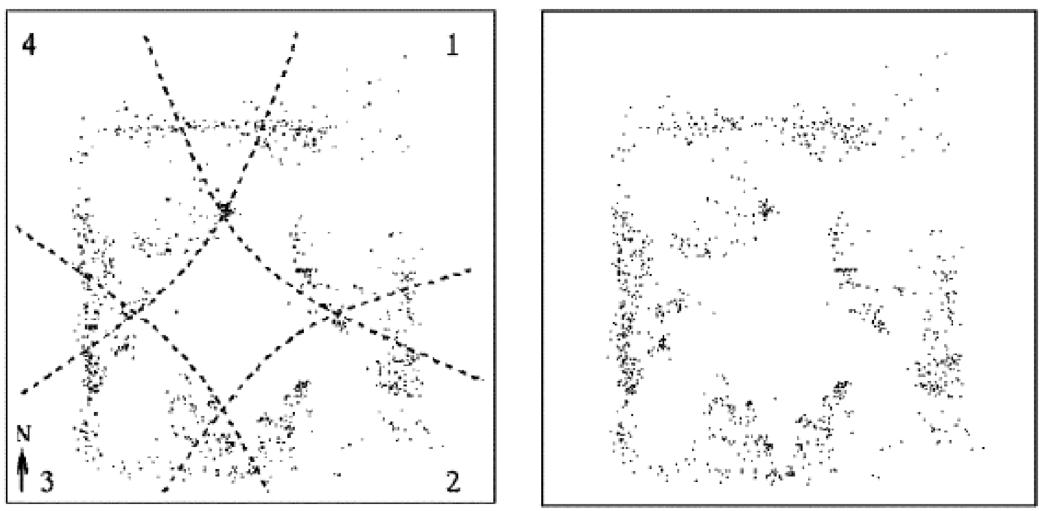
Building Large Map from Submaps

- Use submaps:
  1) Divide image sequence when discontinuity occurs
  2) Build submaps for each divided sequence
  3) Merge submaps by map alignment



Building Large Map from Submaps

- Alignment techniques



(a) (b)

Pairwise alignment

$$\begin{pmatrix} 1 \leftrightarrow 2 \\ 2 \leftrightarrow 3 \\ 3 \leftrightarrow 4 \end{pmatrix}$$

Incremental alignment

$$\begin{pmatrix} 1 \leftrightarrow 2 \\ 1,2 \leftrightarrow 3 \\ 1,2,3 \leftrightarrow 4 \end{pmatrix}$$

Closing the Loop

- Closing the loop means revisiting a previously observed scene.
- When image sequences form a loop, the method could still suffer from accumulated error
- Loop closing condition is a great clue to make the whole map accurate
- Does backward correction using global minimization

Global Minimization

- Backward correction using pairwise submap alignment
- For submaps 1, 2, …, n, and T_i is coordinate transformation of submap i to submap i+1

$$\mathbf{T}_1\mathbf{T}_2\ldots\mathbf{T}_{n-1}\mathbf{T}_n = \mathbf{I}$$

- Find correction vector c to minimize accumulated error:
  - Minimize $|\mathbf{J}\mathbf{c} - \mathbf{e}|^2$
- Adopt landmark uncertainty factor using weight matrix

Summary

- Build a 3D landmark map only with image sequences and raw odometry (SLAM)
- Solve global localization problem using image match via RANSAC and Hough Transform
  - RANSAC > Hough, with SIFT features
  - RANSAC < Hough, with nonspecific features
- Solve closing loop problem with:
  - Pairwise submap matching
  - Error correction with landmark uncertainty

Location and Orientation Prediction with Single Image

- Objective
  - Retrieve information about an urban scene using a single image from a mobile device
  - Locate correct position and orientation of user with image retrieval and comparison
- Reference paper
  - Image-Based Localisation, Cipolla et al. VSMM 2004.

Approach Outline

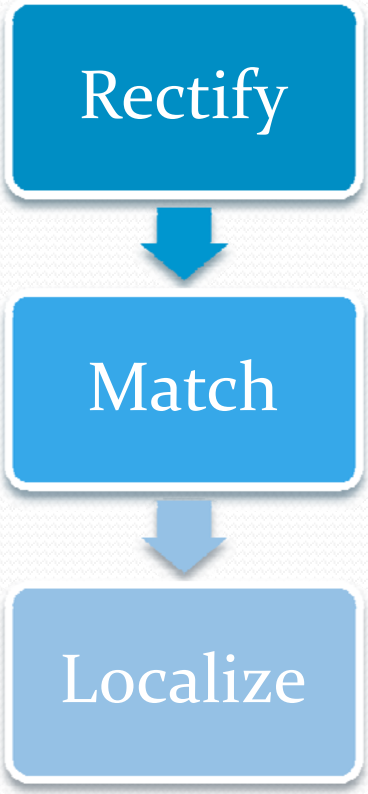


Image Rectification

- Find straight edge lines

- Find horizontal and vertical vanishing lines
- Find rectifying rotation matrix
- Get canonical image

Image Rectification

- Canonical images
  - Facades after rectification

Matching Two Canonical Views

- Match with simple isotropic scaling factor
  - Only horizontal line alignment is needed

$$\mathrm{p}'_{\perp} = \begin{bmatrix} \alpha & 0 & t_x \\ 0 & \alpha & h' - \alpha h \\ 0 & 0 & 1 \end{bmatrix} \mathrm{p}_{\perp} = \mathtt{H}_m \mathrm{p}_{\perp}$$

- Feature detection for canonical views
  - Harris-Stephens corner detector
  - Affine or perspective invariant is not needed
  - Features are characterized by a descriptor based on the surrounding image

Matching Two Canonical Views

- Matching by search
  - A range of scales for both views are compared

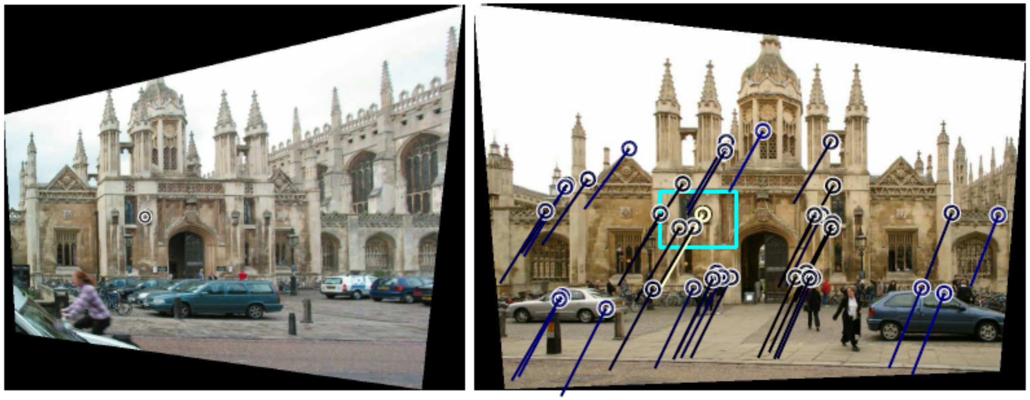
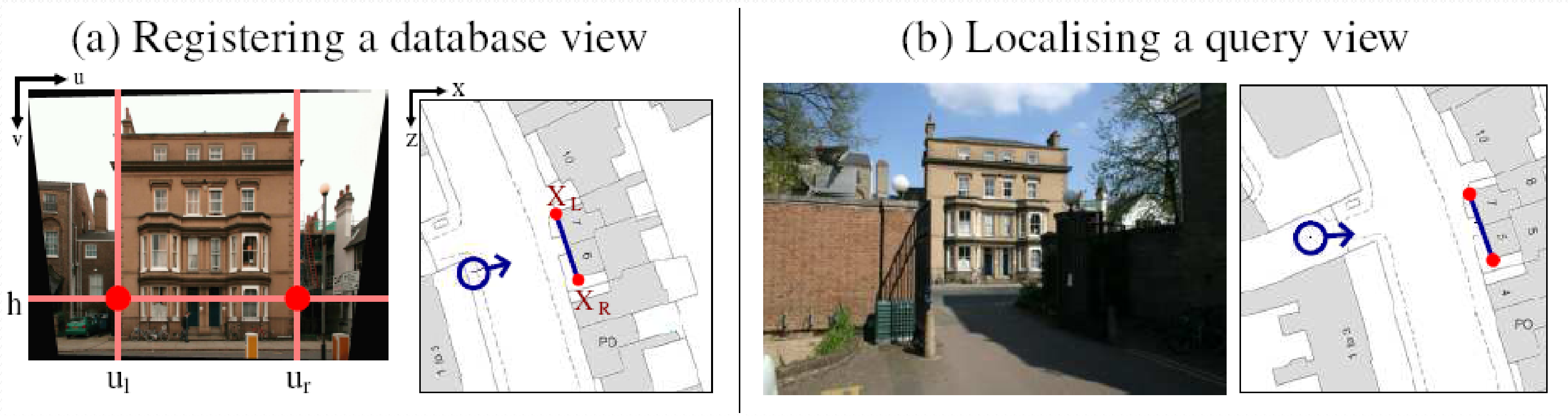


Figure 3. Features detected in two levels being compared. For the highlighted feature in image 1 (left), the search region and matching feature for a particular scale and translation are shown in image 2 (right).

Localization

- Localizing the user



Results

Summary

- Localization of position and orientation with a single image given image database
- Enable to navigate in an urban environment using a mobile device
- Registration of database images are needed with designating façades

- Limitation
  - Could fail if buildings are similar
  - Matching database view and query view could be slow

Conclusion

- Simple content-based image retrieval can be well used as location recognition system
- Only with appearances, localization of position and orientation are well-defined
- Visual images are powerful cues to solve loop-closing problem

Discussion Points

- Recognizing distance using cameras
  - Stereo vision
  - How to do with only one camera?
- What kinds of feature detectors and descriptors can pick the particular nature of location recognition domain?
  - SIFT descriptor
  - Gradient orientation histogram
- How to help standard SLAM problem with visual cues?
  - Detecting loop closing condition using still images

Thank you!