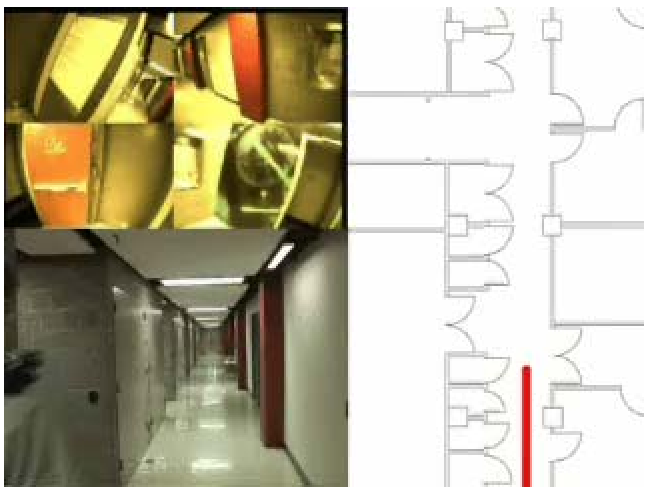
A Self-Calibrating, Vision-based Navigation Assistant



Olivier Koch
koch@csail.mit.edu

Seth Teller
teller@csail.mit.edu

Massachusetts Institute of Technology
Computer Science and Artificial Intelligence Laboratory (CSAIL)

Motivation

- Navigation in GPS-denied environments
  - Indoors / underground / urban areas with limited sky visibility
- Exploration and path retracing for human users
  - Soldiers in the field / Visually impaired / Disabled people

Related work

Vision-based *S*imultaneous Localization and Mapping

- Wolf et al., Robust Vision-Based Localization by Combining an Image Retrieval System with Monte Carlo Localization, IEEE Transactions Robotics 2005
- Davison et al., MonoSLAM: Real-Time Single Camera SLAM, PAMI 2007
- J. Neira et al., Data association in *O(n)* for Divide and Conquer SLAM, RSS 2007
- Konolige, Agrawal et al., Mapping, Navigation and Learning for Off-road Traversal, Journal of Field Robotics 2008

Related work

Hybrid metrical-topological localization

- Bosse et al., SLAM in Large-Scale Cyclic Environments Using the Atlas Framework, IJRR 2004
- B. Kuipers, Using the topological skeleton for scalable global metrical map-building, IROS 2004
- Zhang & Kosecka, Hierarchical Building Recognition, Image and Vision Computing 2007

Appearance-based navigation

- Cummins & Newman, Probabilistic Appearance-Based Navigation and Loop Closing, ICRA 2007

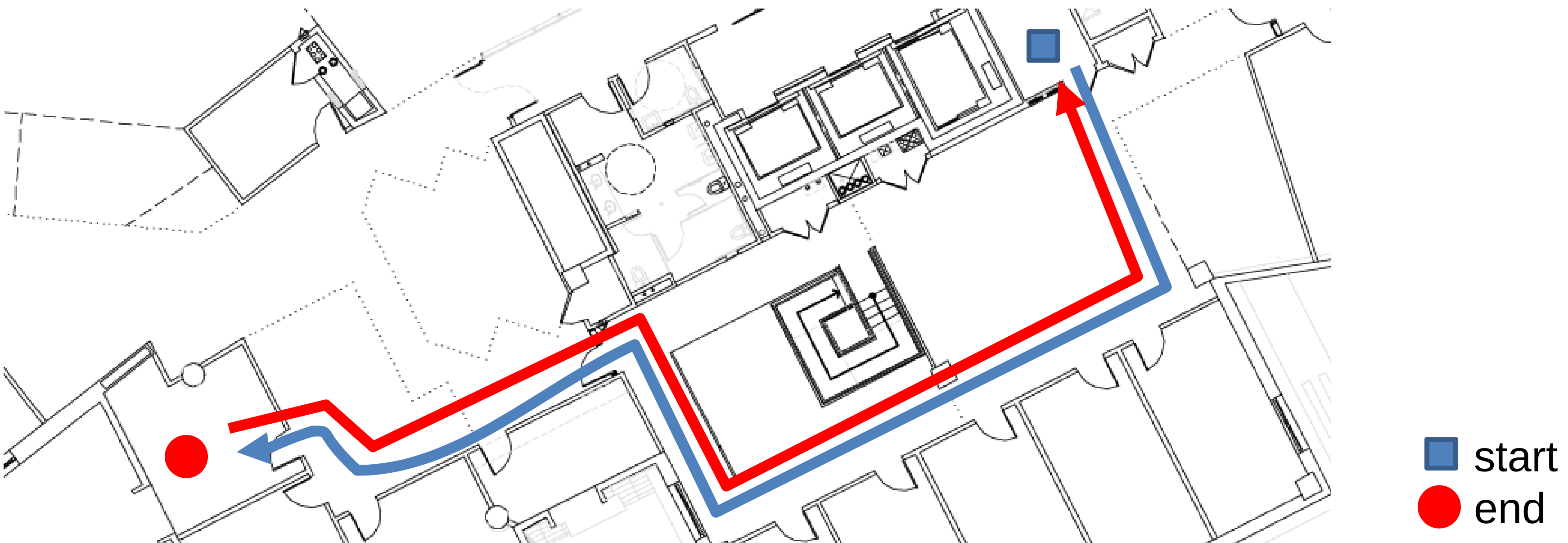
Problem statement

Inputs:

- Training sequence
- Live video sequence

Outputs:

- Live walking guidance
- Helps user retrace path



Capture Rig & User Interface

Four IEEE1394 PointGrey Firefly Cameras
4 x 360 x 240 SIFT detection, tracking @ 4Hz
FOV: 360° (horiz.) x 90° (vert.)

Tablet PC Interface
With earphone

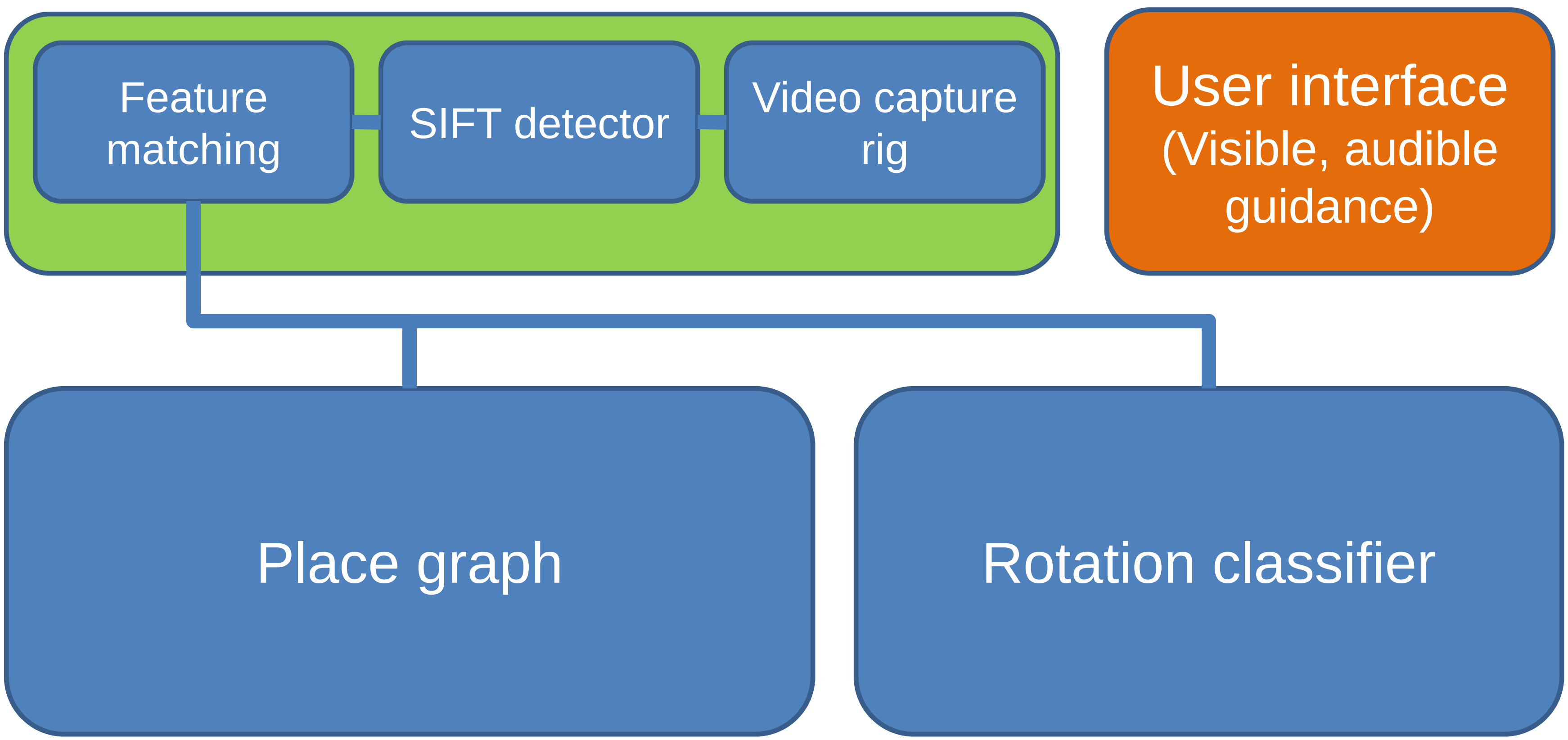
Novelty

- Interface:
  - Provides non-metrical walking guidance to humans
  - Guidance is “user-centered,” i.e., body-relative
- Purely vision-based:
  - Requires no camera calibration
  - Does not constrain the number of cameras or their relative positions on the capture rig
  - Novel method for correlating user motion with image feature motion

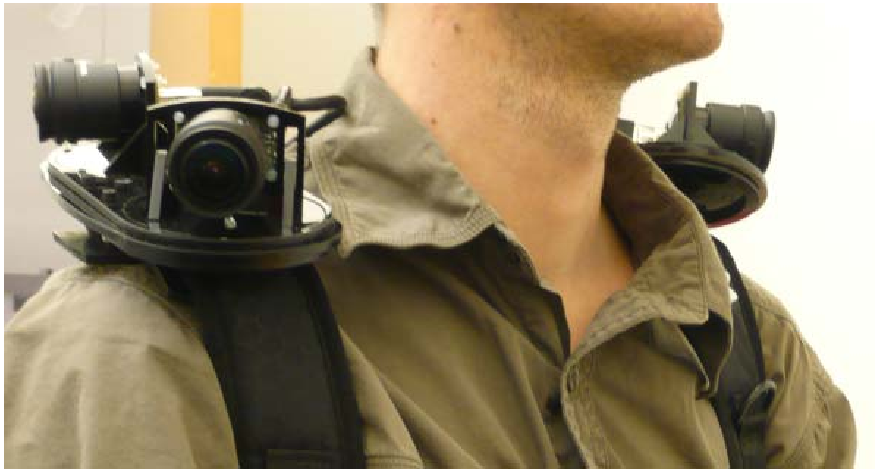
Assumptions

- User motion is smooth
- Rigid-body transformation between cameras is fixed but can change slightly over time
- System requires a brief sequence with known user motion
  - Turning slowly in place for two revolutions
  - Required only once for any given camera configuration
- Environment is 2D, mostly static and contains distinctive visual features

System Overview



Capture Rig & User Interface



Four IEEE1394 PointGrey Firefly Cameras
4 x 360 x 240 SIFT detection, tracking @ 4Hz
FOV: 360° (horiz.) x 90° (vert.)

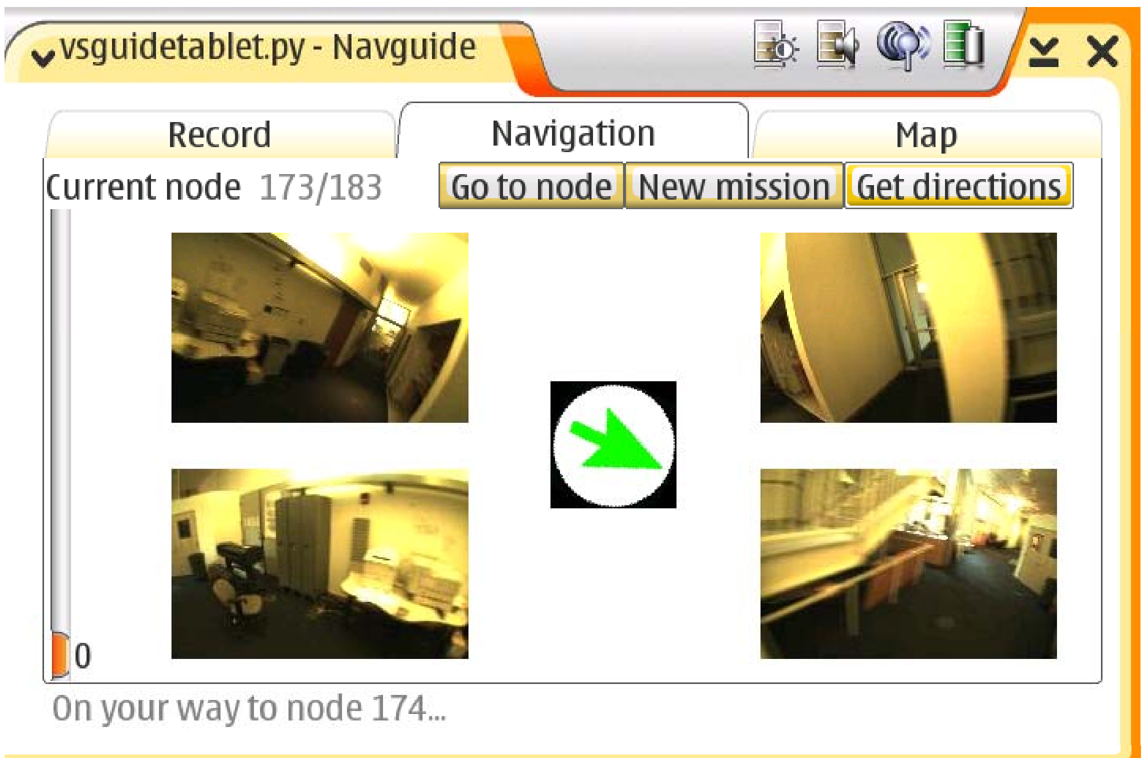
Wearable embedded PC cluster
3 x 1.8Ghz Intel Core 2 Duo CPUs
3 hours untethered operation

Tablet PC Interface
With earphone

Capture Rig & User Interface

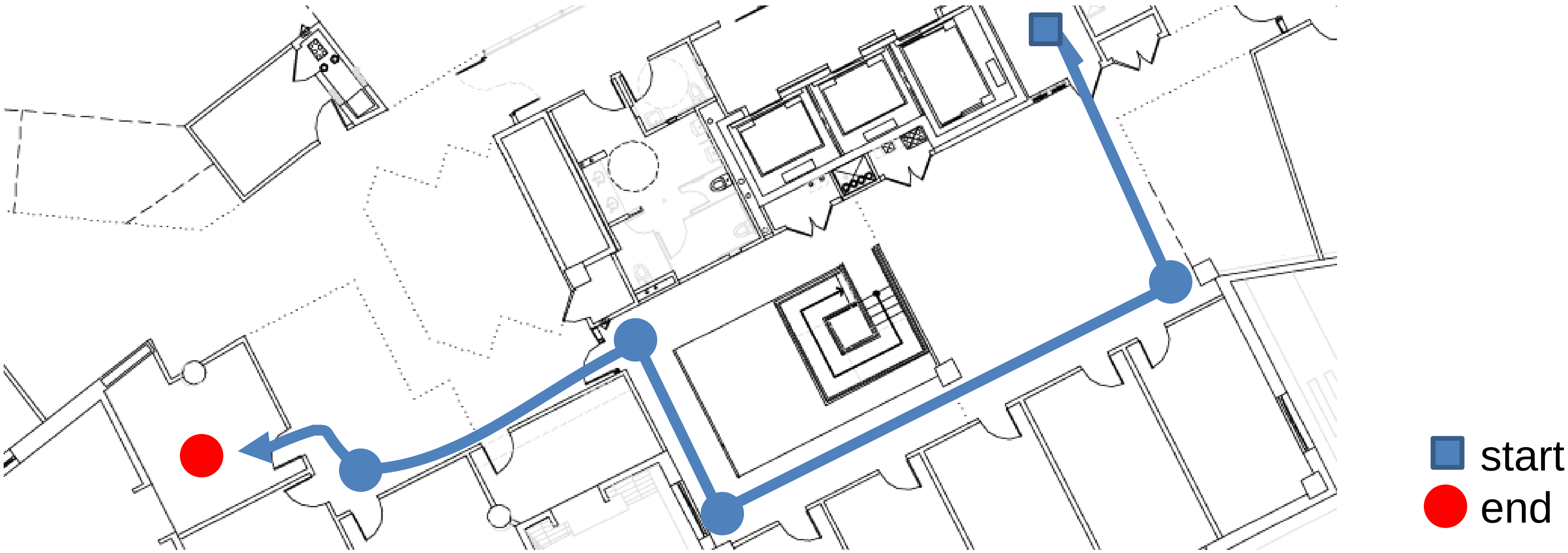
Qualitative guidance

- Our system provides human-understandable guidance



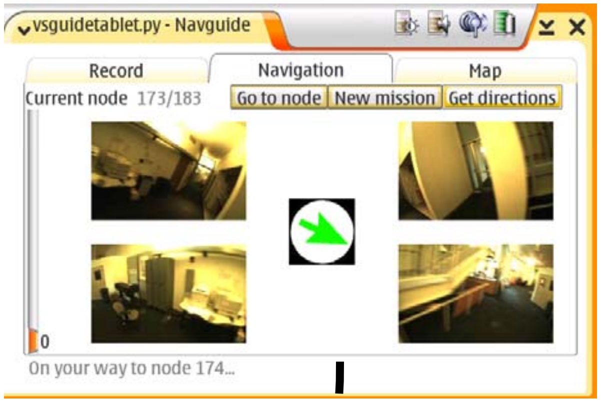
Place Graph

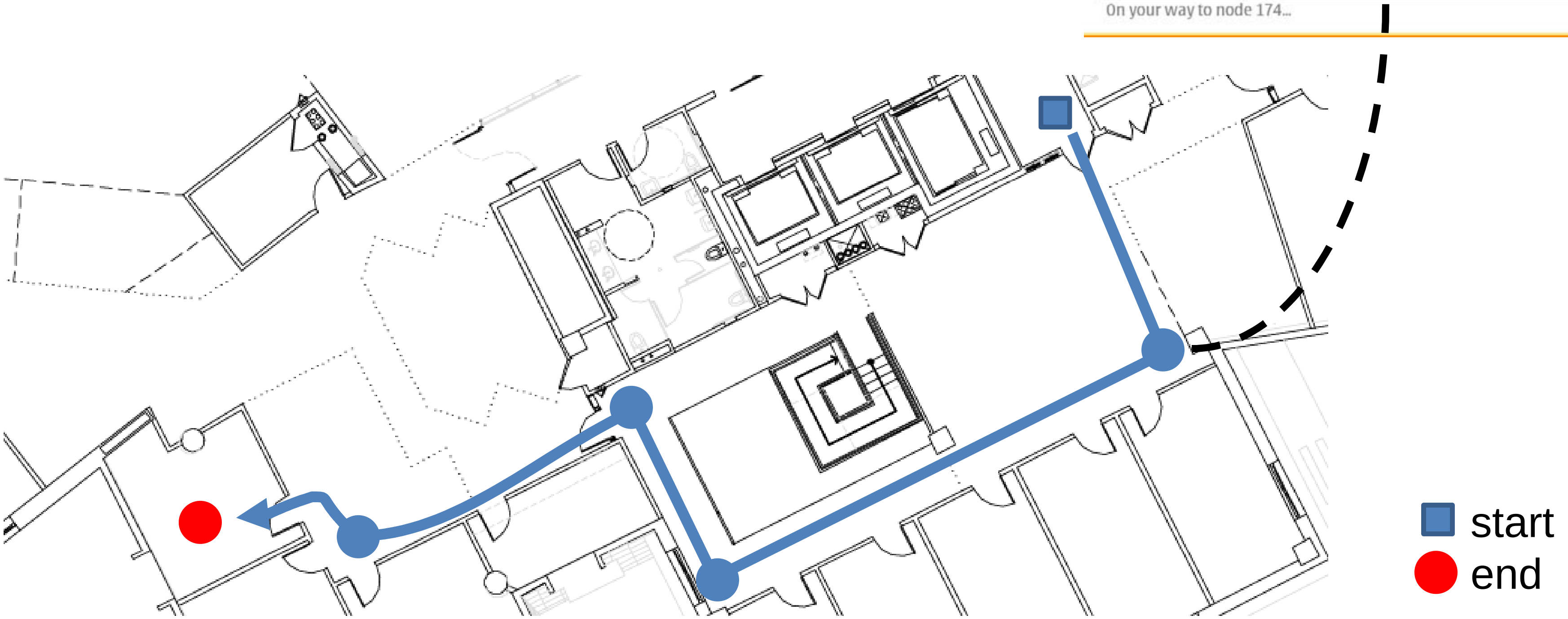
- Node: place of strategic interest for navigation
- Edge: a direct physical path between nodes
- Graph built online and automatically during exploration



Navigation Guidance

- Provide *rotation guidance* at nodes
- Provide *relative progress* along edges
- In a human-understandable fashion





Place graph data structure

Node

• One image, feature set

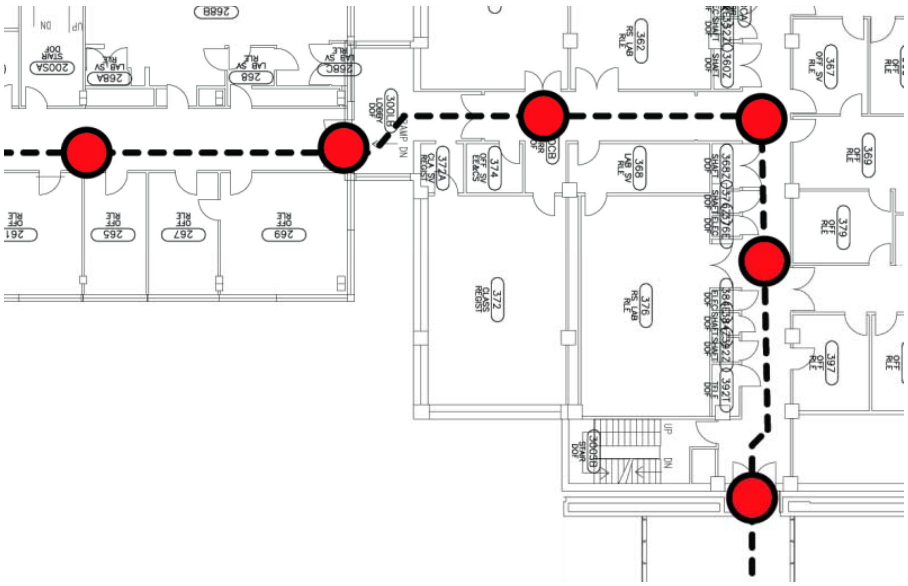
Edge

• A series of image feature sets collected at regular time intervals
• Stored in an *incremental visual dictionary* [1]

[1] D. Filliat, **A visual bag of words method for interactive qualitative localization and mapping.** *Proceedings of the International Conference on Robotics and Automation (ICRA)*. 2007.

Place Graph

- Graph nodes are created online and automatically:
  - Where rotation rate is high (i.e., user is turning)
  - Where scene appearance changes drastically (e.g., user exits room)



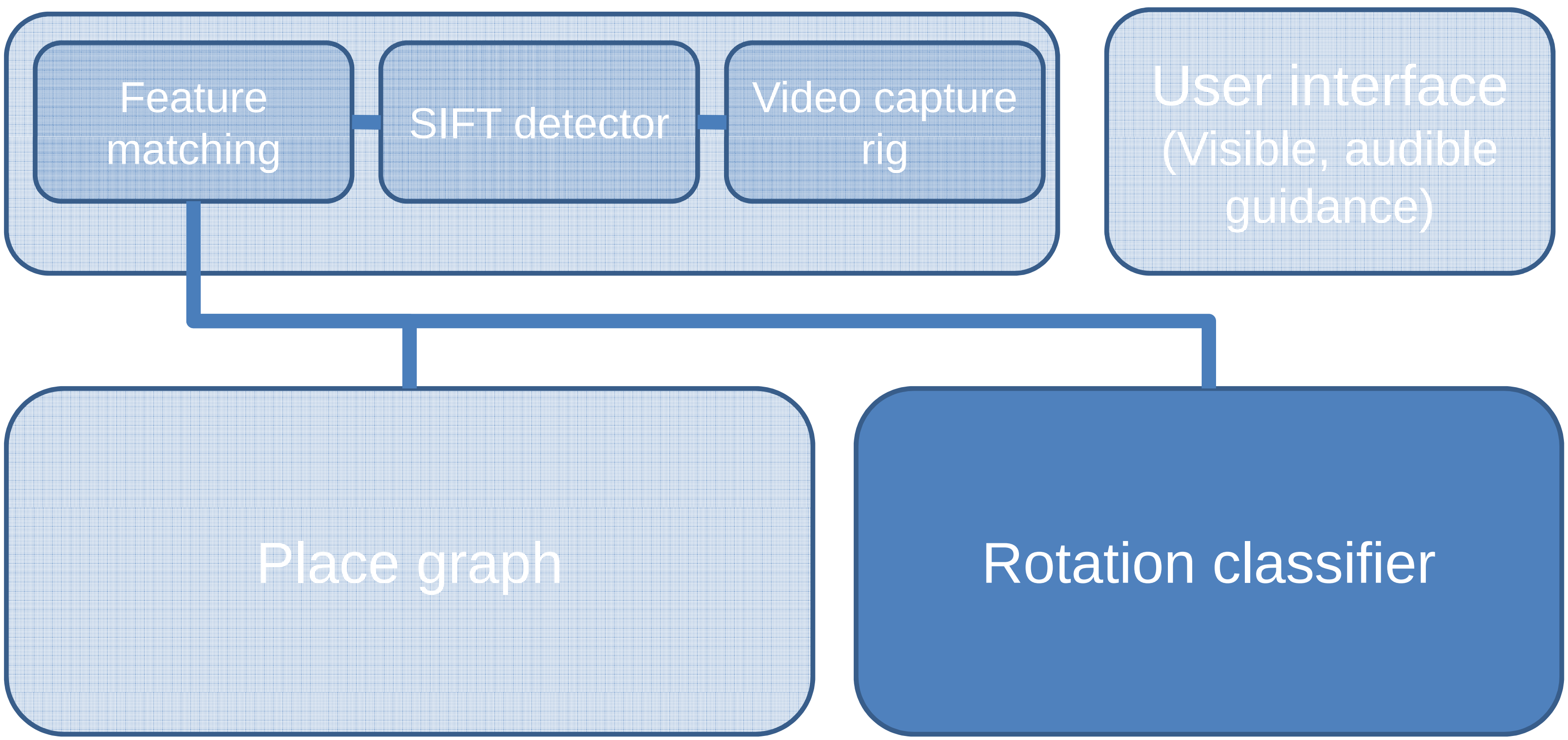
Subset of Place Graph (INDOOR dataset), with nodes overlaid manually for visualization

FRONT LEFT FRONT RIGHT

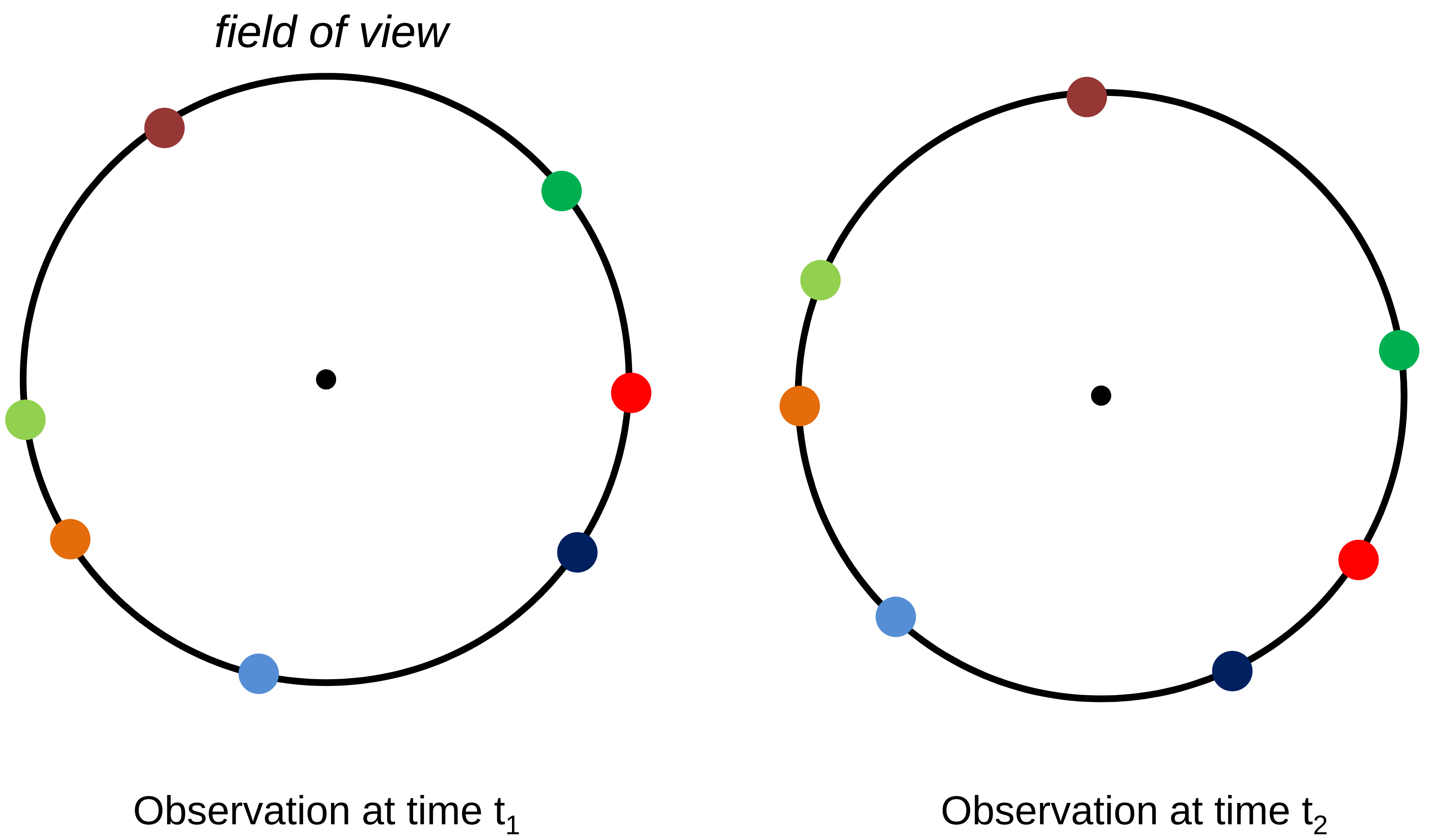
REAR LEFT REAR RIGHT

Example node (INDOOR dataset)

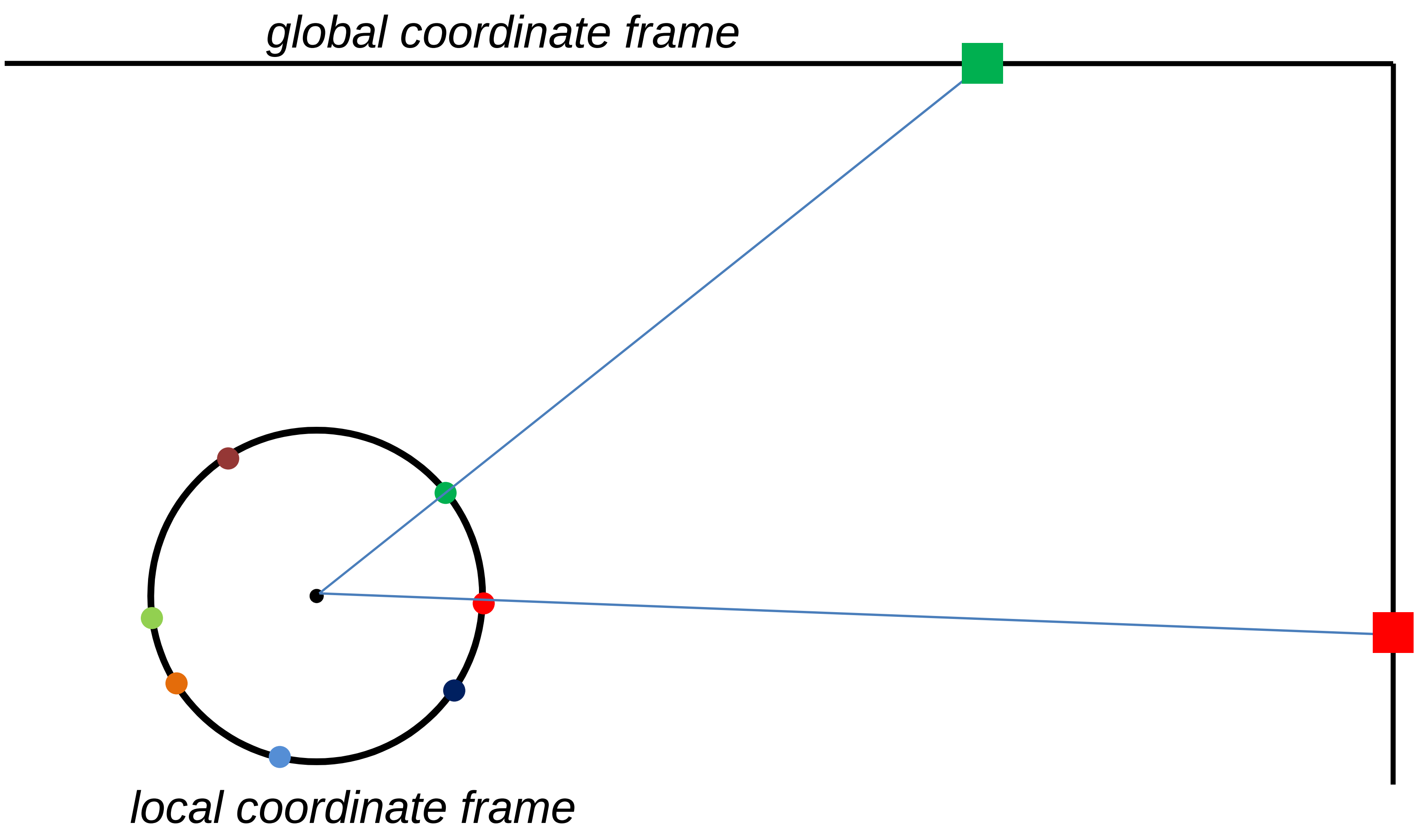
System Overview



Rotation classifier

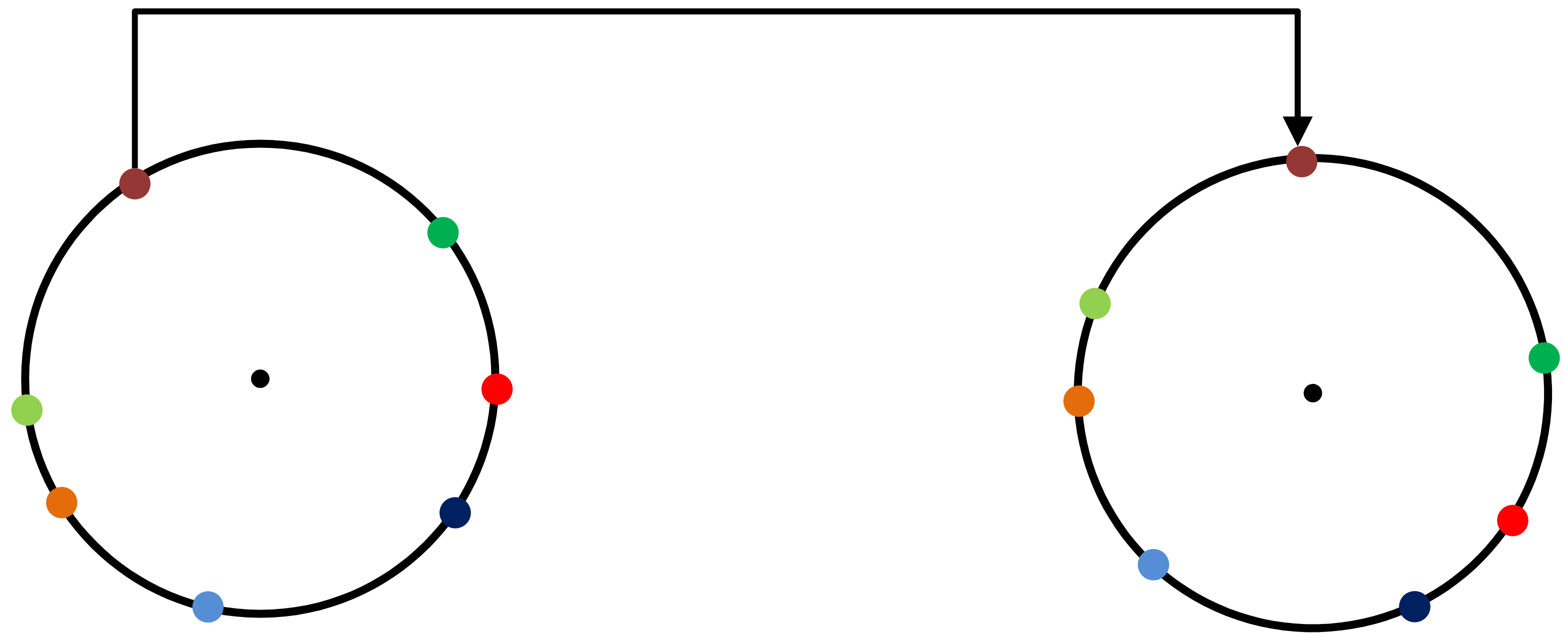


Rotation classifier



Rotation classifier

Can we learn which rotation of the user would bring these two features in alignment with no camera calibration?



Observation at time t_1 Observation at time t_2

Rotation classifier

TRAINING

Input

Known Motion Sequence

Output

Classifier table relating user motion, feature evolution in image

QUERY

Input

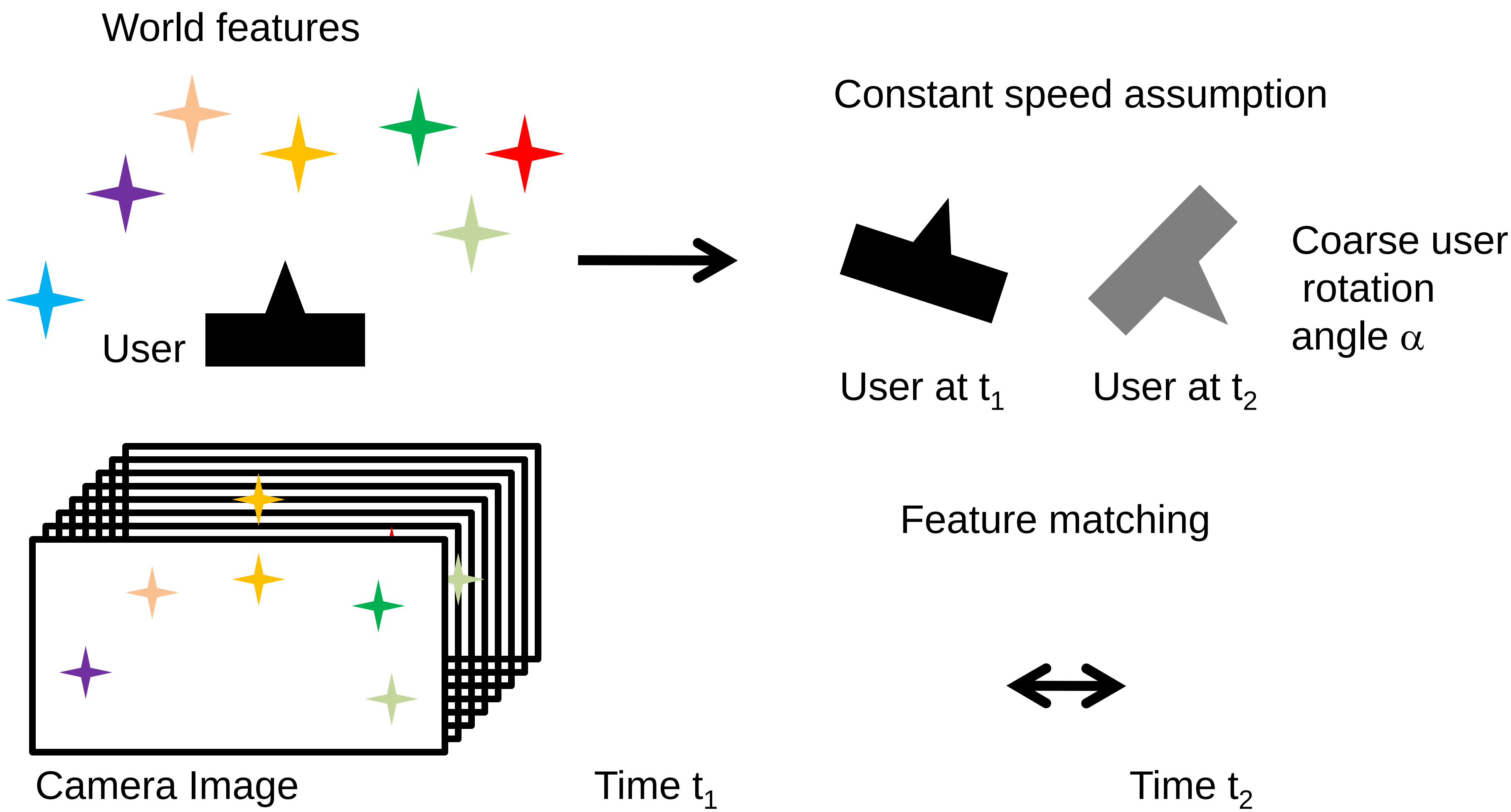
Two feature sets
Classifier table

Output

User rotation that brings the two feature sets into maximal alignment

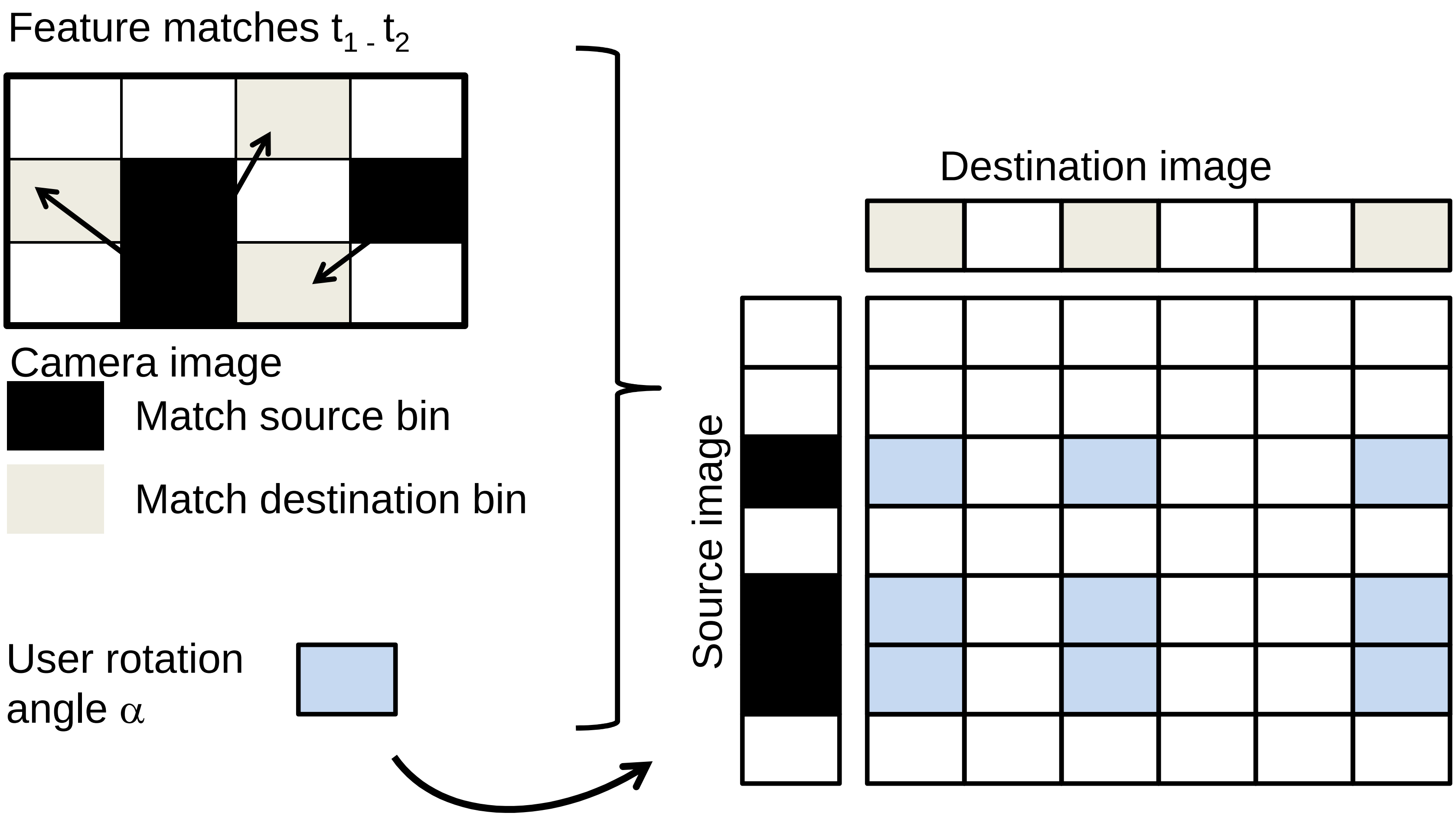
Rotation classifier

TRAINING



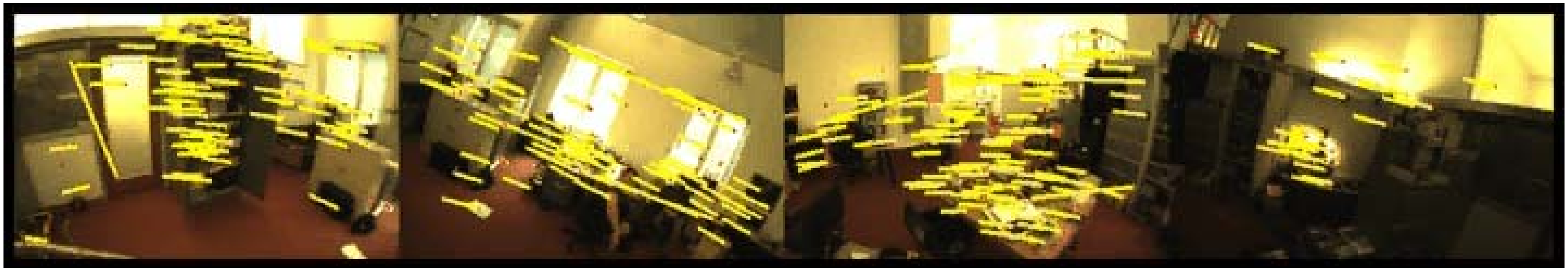
Rotation classifier

TRAINING



Rotation classifier

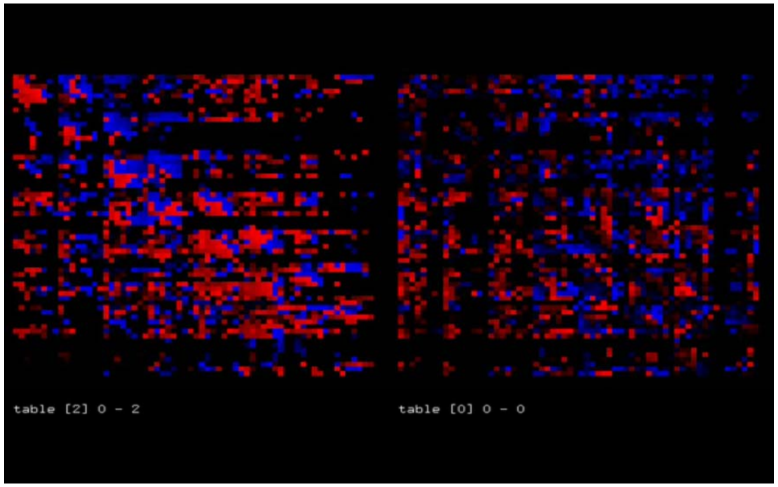
TRAINING



Known motion - user rotates in place 1 minute @ 4 Hz = 240 frames

Needs $O(n^2)$ storage:

$$n_{tables} = \binom{n}{2} + n$$



Red : angle > 0

Blue : angle < 0

Classifier tables
Left: camera 0 – 0
Right: camera 0 – 2

Rotation classifier

TRAINING

Input

Known Video Sequence
Coarse user motion

→

Output

Classifier table

QUERY

Input

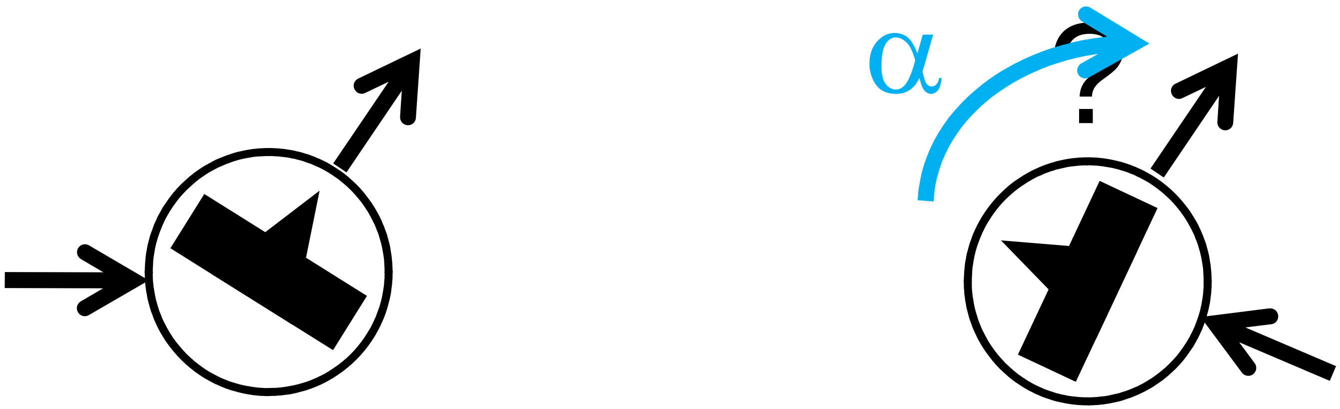
Two feature sets
Classifier table

→

Output

User rotation that brings the two feature sets into maximal alignment

Navigation at node



First visit ($t = t_1$)

Revisit ($t = t_2$)

Input

Observation at time t_1 (visit) and time t_2 (revisit)

Output

Rotation guidance to align user with appropriate outgoing edge

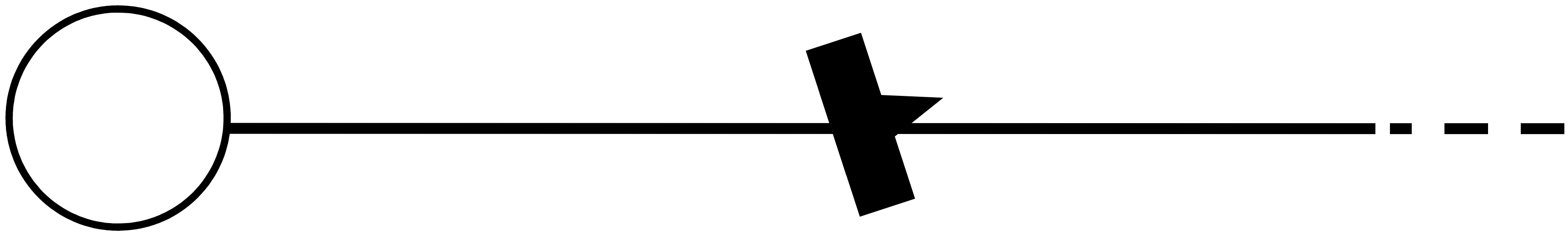
Method

Match features between t_1 and t_2

For each match, query the classifier and return a rotation angle

Run RANSAC voting to determine optimal rotation angle α

Navigation along edges



Input

A series of observations $S_0 = \{o^1, \dots\ o^n\}$ along edge (first visit)

Current observation o'^t

Output

Relative progress along the edge (normalized from 0 to 100%)

Navigation along edges

Method: recursive state estimator

State vector v.

v_i represents the probability of the user standing at location of observation o^i.

Initialization (user leaving node)

v_i = 1 if i=0 (i.e., user is at start of edge), 0 otherwise.

Navigation along edges

At each time step, given a new observation o'^{t}:

$$\mathcal{V}^{t} \xrightarrow{\textbf{transition}} \tilde{\mathcal{V}}^{t+1} \xrightarrow{\textbf{observation}} \mathcal{V}^{t+1}$$

- **Transition update** (motion continuity assumption)

$$\tilde{\mathcal{V}}^{t+1} = \mathcal{V}^{t} \otimes \text{Gaussian } (0,\sigma)$$

where σ is a function of frame rate and typical user motion speed

- **Observation update**

$$\mathcal{V}^{t+1}{}_{i} = \tilde{\mathcal{V}}^{t+1}{}_{i} \times \mathcal{P}\left(o^{i}, o'^{t}\right)$$

where $\mathcal{P}(a,b)$ is the probability that a and b are observed from the same location

Datasets

Name	Duration	Path length	Frame rate	# frames	# nodes
INDOOR	45 min	~2.5 km	4 Hz	11,000	280
OUTDOOR	12 min	~1 km	4 Hz	2,900	43

INDOOR Dataset
MIT Tunnel network

OUTDOOR Dataset
Kendall Square, Cambridge MA

Rotation guidance at nodes

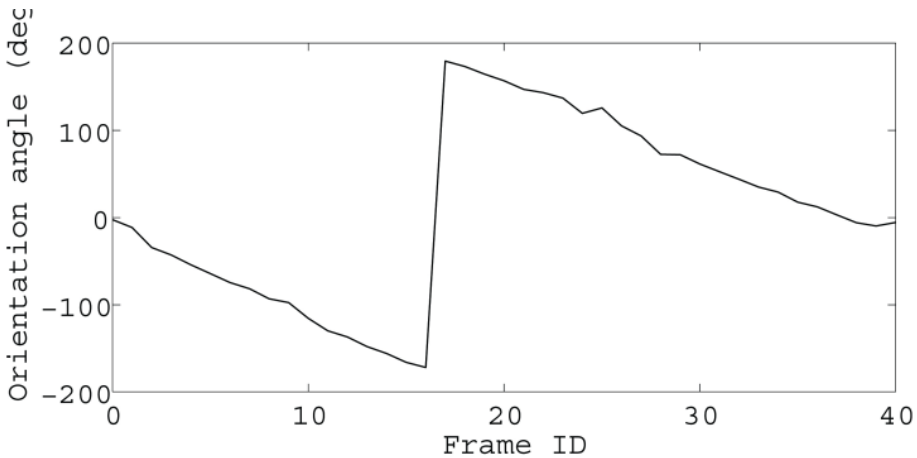


Fig. 1 - Rotation guidance output while user rotates in place in a new environment

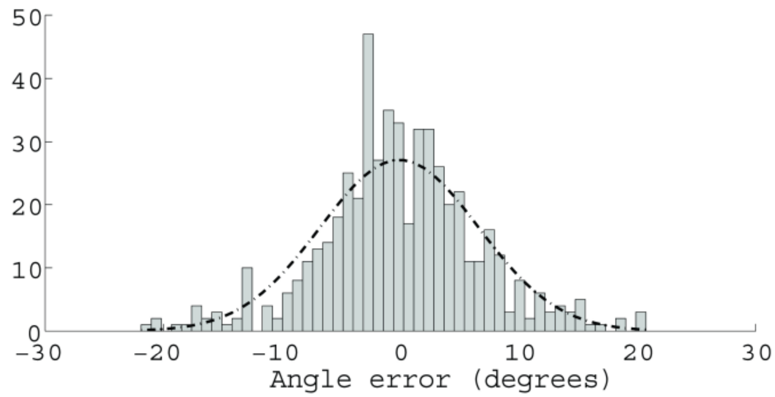
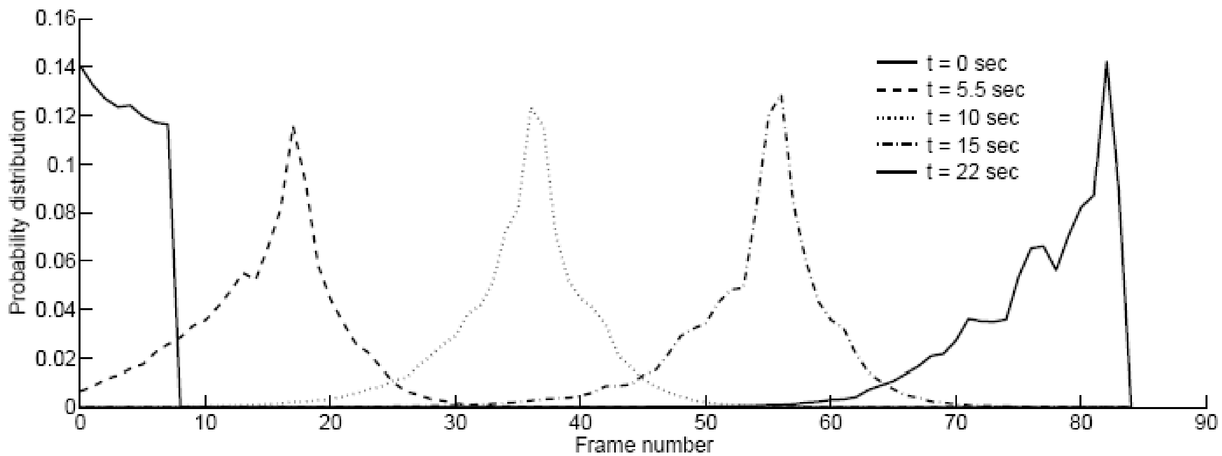
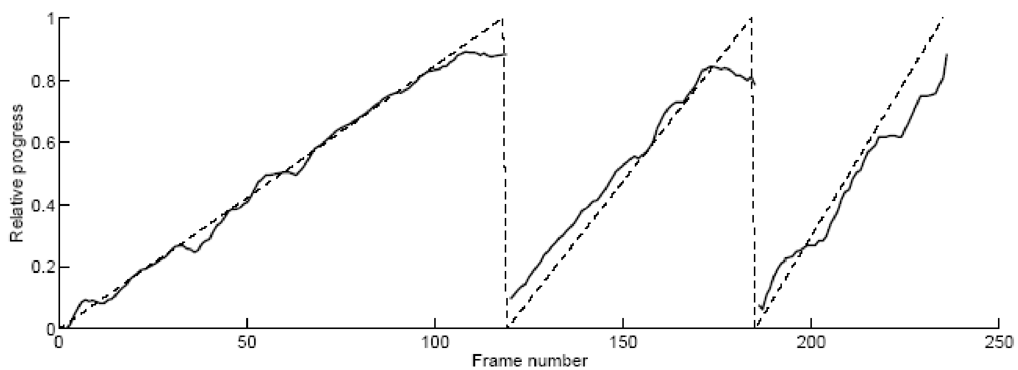


Fig. 2 – Error distribution against IMU-ground truth. **Standard deviation = 12 deg.**

Progress guidance along edges

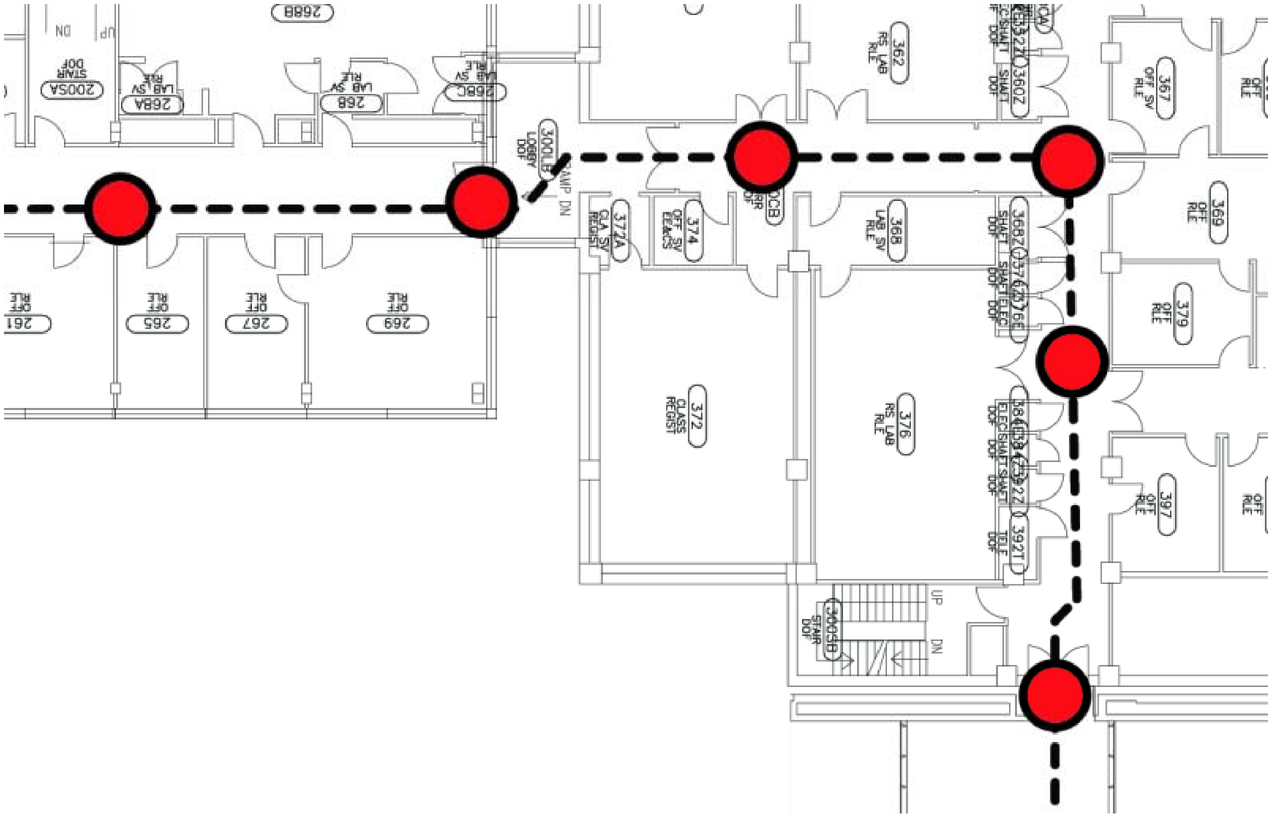


Belief state propagation while user walks along an edge (INDOOR dataset)



Relative progress along several consecutive edges. Ground truth estimated using constant speed assumption. **Std. dev. is 3.3 frames (1 second, ~1.5m)**

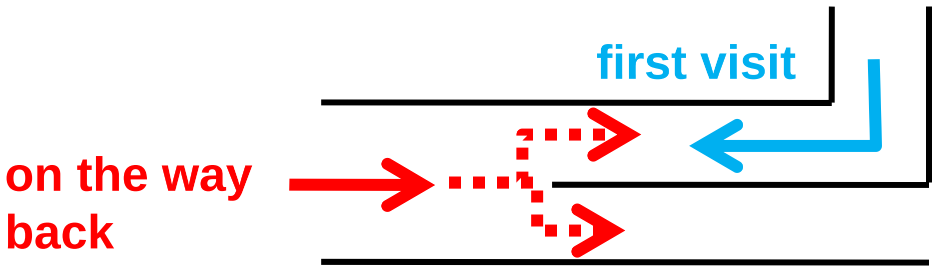
Topological map



Topological map automatically generated by the system (INDOOR dataset). Nodes manually overlaid on map for visualization.

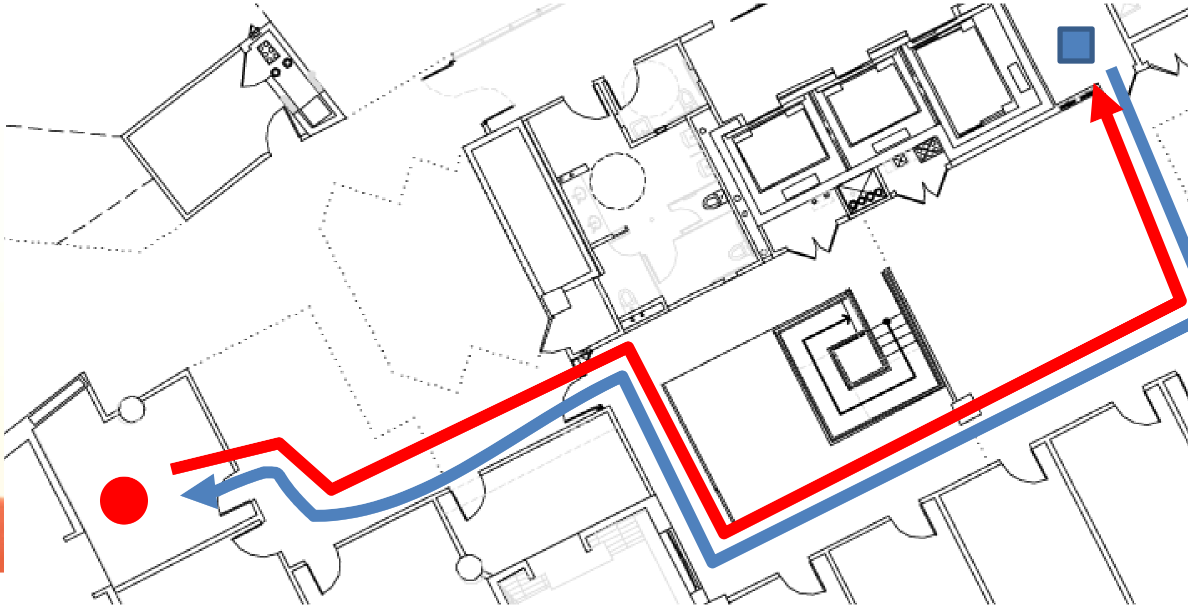
Failure modes

- Ambiguous configurations
- User leaves the exploration path
- Highly repetitive environments (featureless corridors)
- Significant change in lighting
- Dynamic scenes (e.g., crowds)
- Fast user motion (motion blur) or low lighting



Future work

- Global localization
- Extend to 3D motion
- Path self-intersection (non-linear graphs / loop closure)
- Augmented reality applications (e.g., in situ virtual tagging)



Summary

Inputs:

- Training sequence
- Video sequence

Outputs:

- Loose guidance in 2D
- Supports user retrace

- Requires no intrinsic or extrinsic camera calibration
- Generalizes to any number / configuration of cameras
  - Requires roughly fixed rigid-body transform between cameras
- New way of correlating user motion and image motion
- Provides loose guidance / directions to user

Discussion