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
  - Indoor / underground / dense urban areas
- Explore and retrace for human users
  - Soldiers in the field / Visually impaired / Disabled people



Related work

Visual SLAM

- Davison et al., MonoSLAM: Real-Time Single Camera SLAM, PAMI ’07
- J. Neira et al., Data association in O(n) for Divide and Conquer SLAM, RSS ’07
- Wolf et al., Robust Vision-Based Localization by Combining an Image Retrieval System with Monte Carlo Localization, IEEE Transactions Robotics ’05
- Konolige, Agrawal et al., . Mapping, Navigation and Learning for Off-road Traversal, Journal of Field Robotics ‘08

Related work

Metric and topological localization

- Zhang & Kosecka, Hierarchical Building Recognition, Image and Vision Computing ‘07
- B. Kuipers, Using the topological skeleton for scalable global metrical map-building, IROS ’04

Appearance-based

- Cummins & Newman, Probabilistic Appearance Based Navigation and Loop Closing, ICRA ‘07

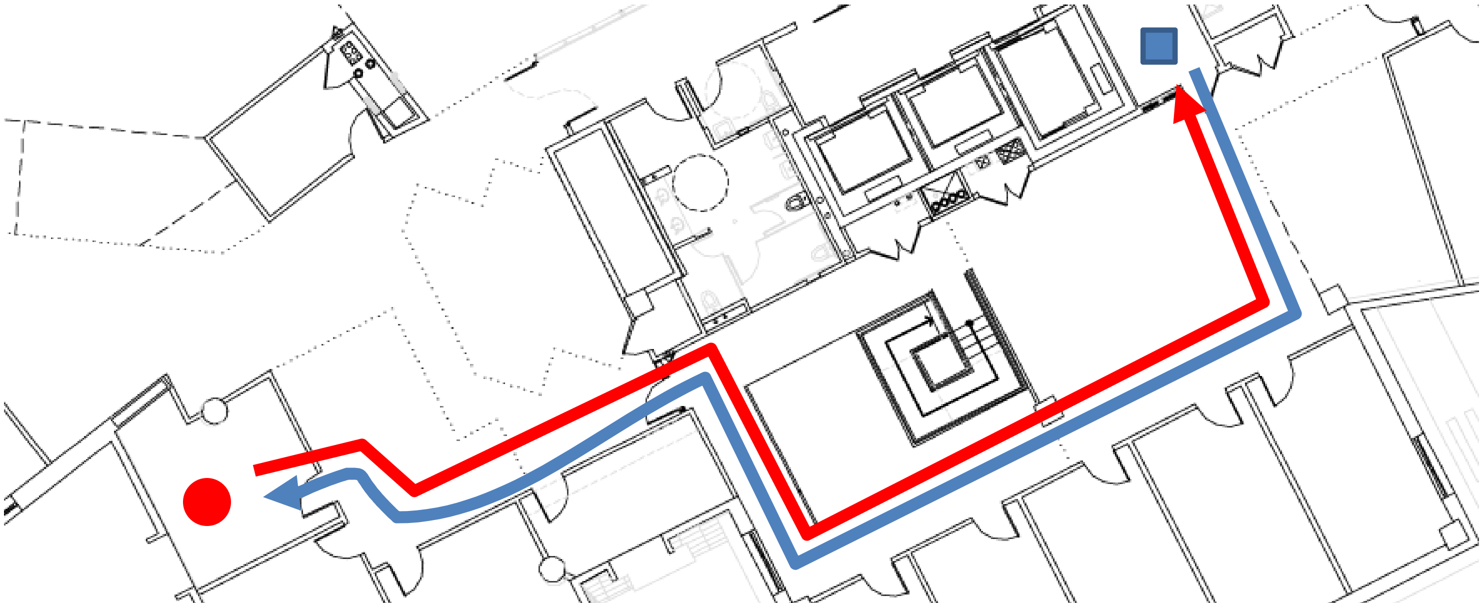
Problem statement

Input

- Video sequence
- Calibration sequence

Output

- Backtrack along linear path
- Loose guidance in 2D



start
end

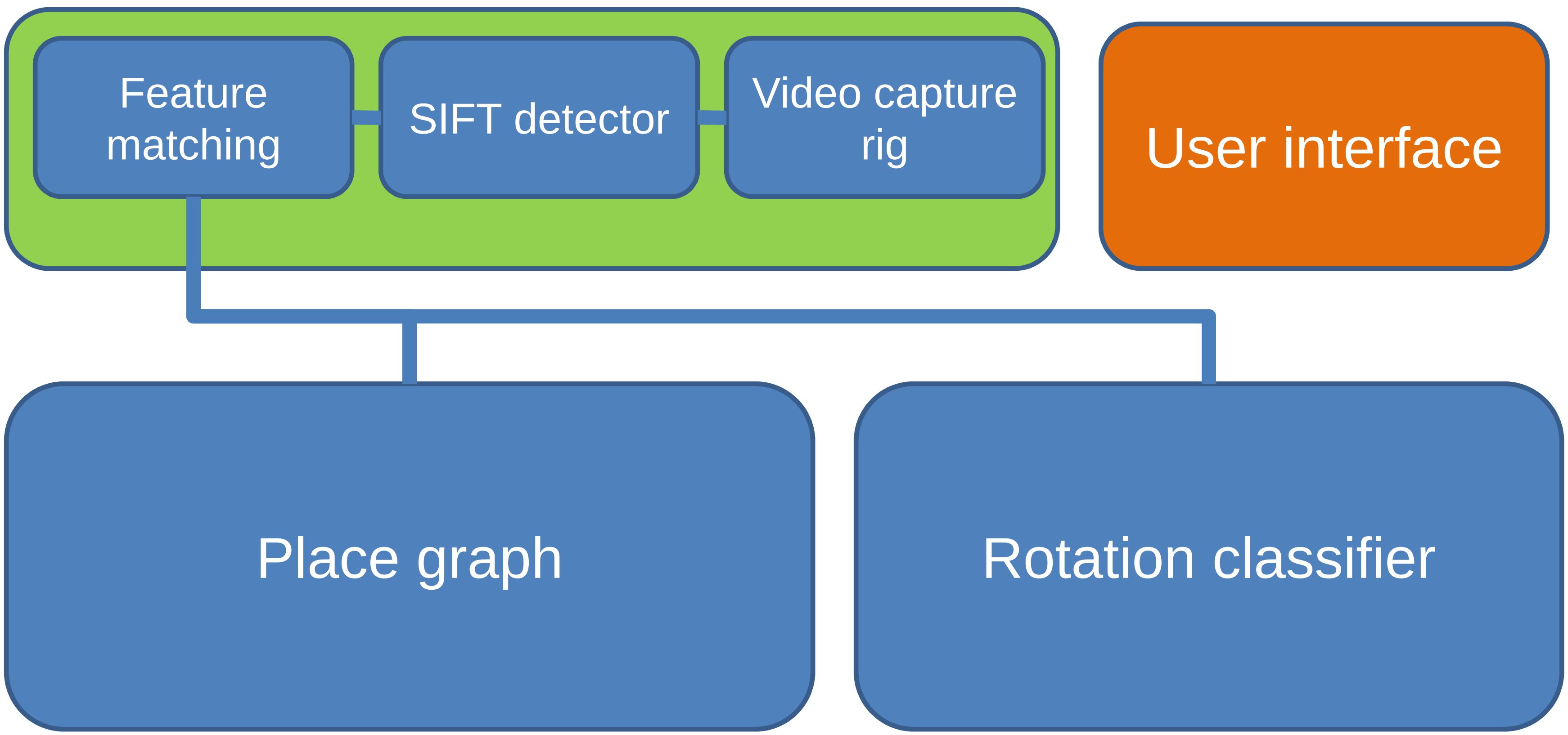
Novelty

- Provides non-metric, loose guidance to humans
- Purely vision-based
- Requires no camera calibration
- Does not constraint the number of cameras or their relative position on the rig
- Uses a new way of correlating user to image motion

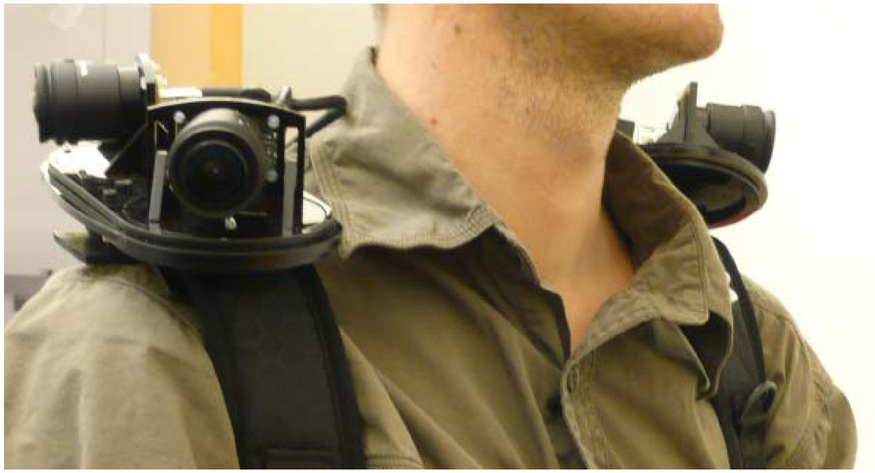
Assumptions

- The motion of the user is continuous
- The rigid-body transformation between cameras is fixed but can change slightly
- The environment is mostly static and contains descriptive visual features
- The user evolves in a flat 2D world (although our method extends to 3D motion)
- The user needs to train the system for a few minutes (in any environment, once and for all)

System Overview



Capture Rig & User Interface



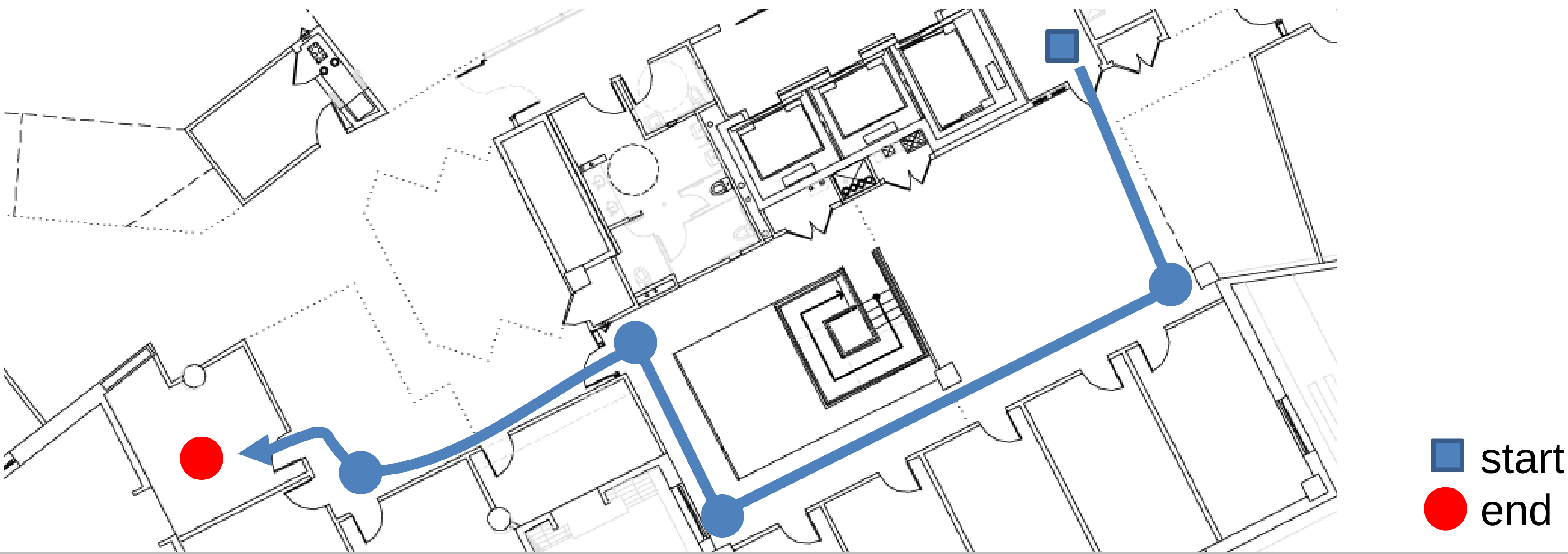
Four IEEE1394 PointGrey Firefly Cameras
4 x 360 x 240 SIFT detection & tracking at 4Hz
FOV: 360° (h) x 90° (v)

Embedded PC cluster
Three 1.8Ghz Intel Core 2 Duo
3h untethered operation

Tablet PC Interface
Microphone/earphone

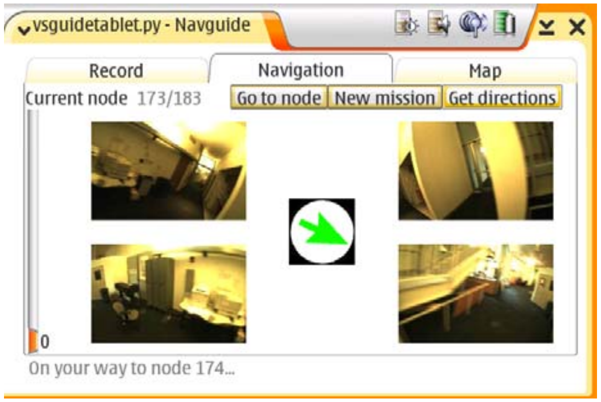
Place Graph

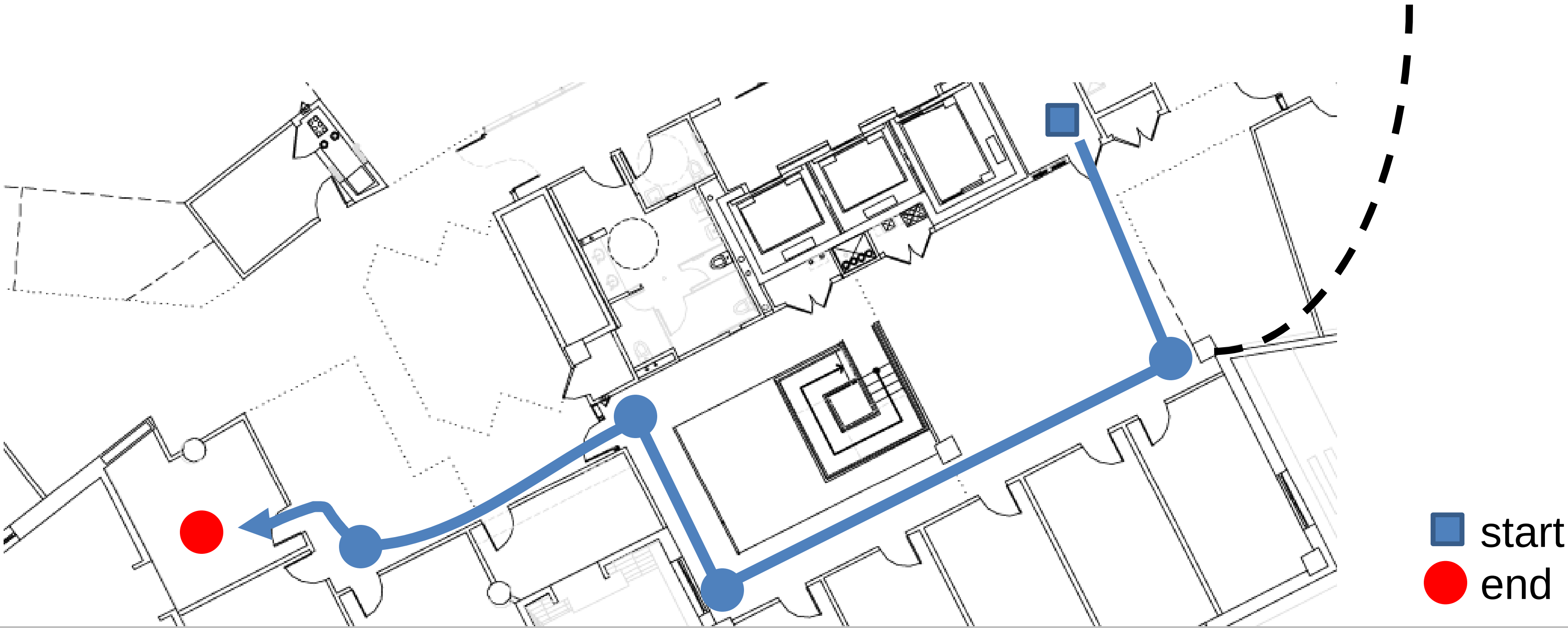
- Nodes: places of strategic interest for navigation
- Edges: physical path between nodes
- Built online and automatically during exploration



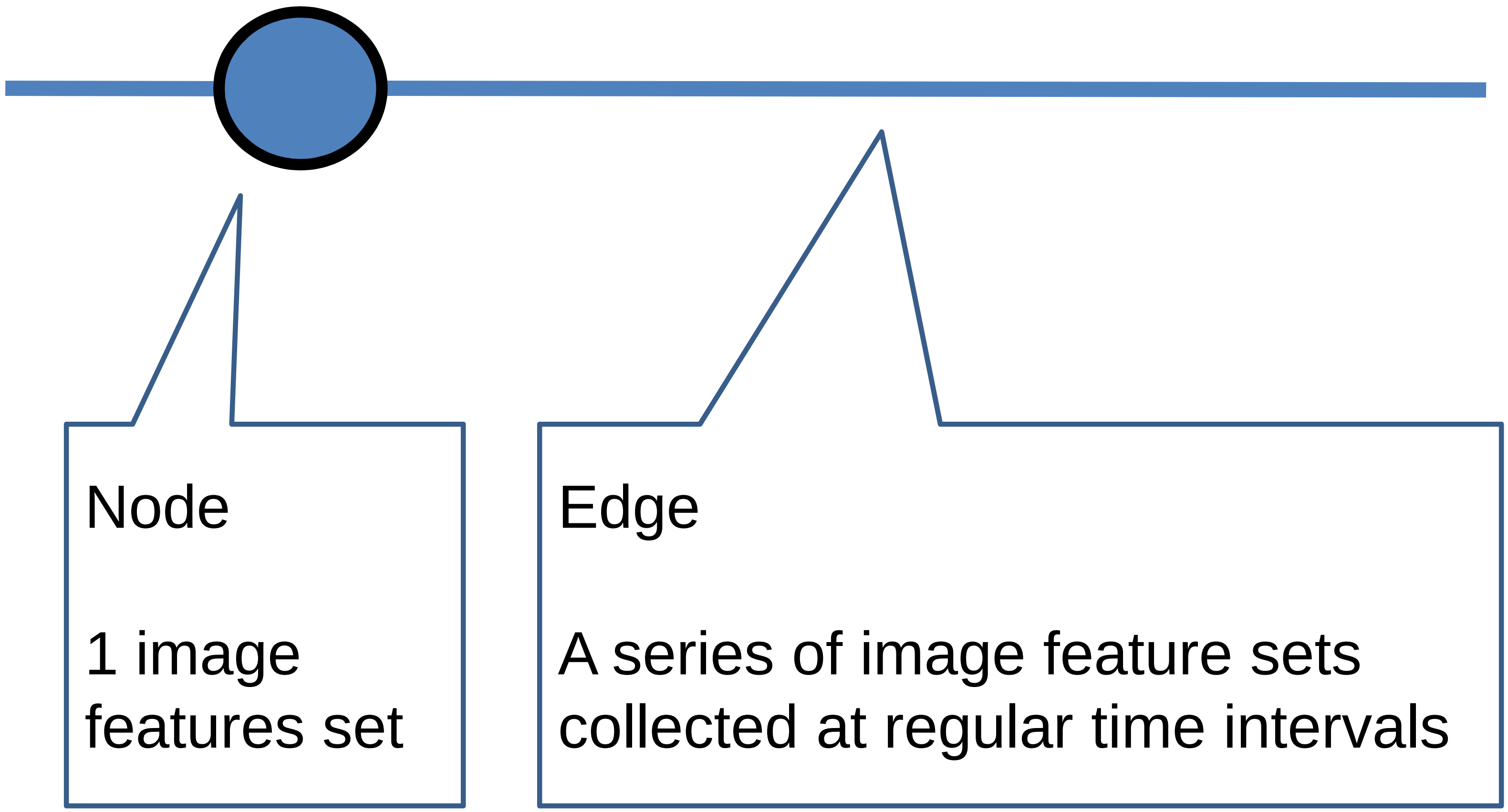
Navigation Strategy

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



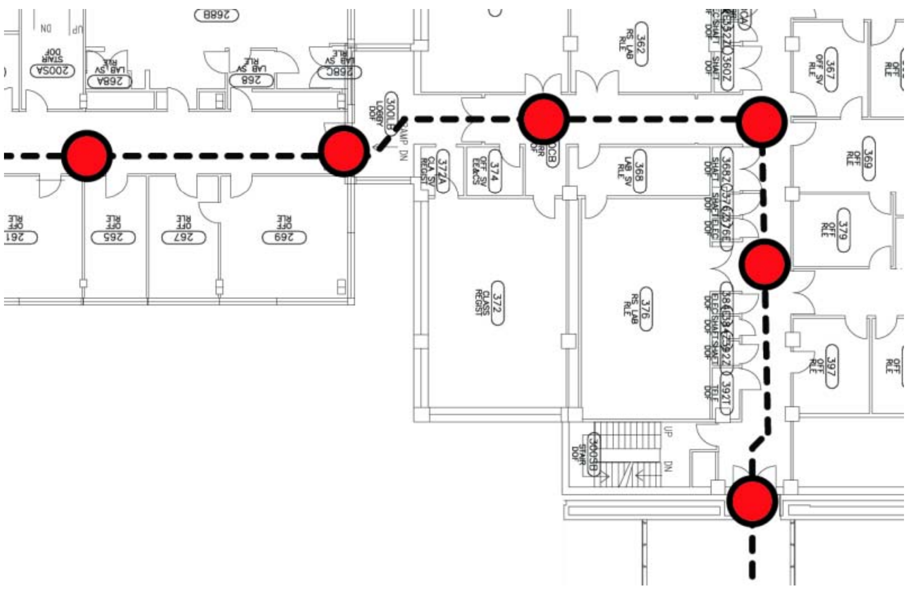


Place graph data structure



Place Graph

- Nodes in the map are created online and automatically:
  - At places of high rotation rate
  - At places of drastic change in the scene appearance



Subset of Place Graph (INDOOR dataset)
Nodes overlaid on 2D map manually.

FRONT LEFT FRONT RIGHT

REAR LEFT REAR RIGHT

Sample node (INDOOR dataset)

Rotation classifier

TRAINING

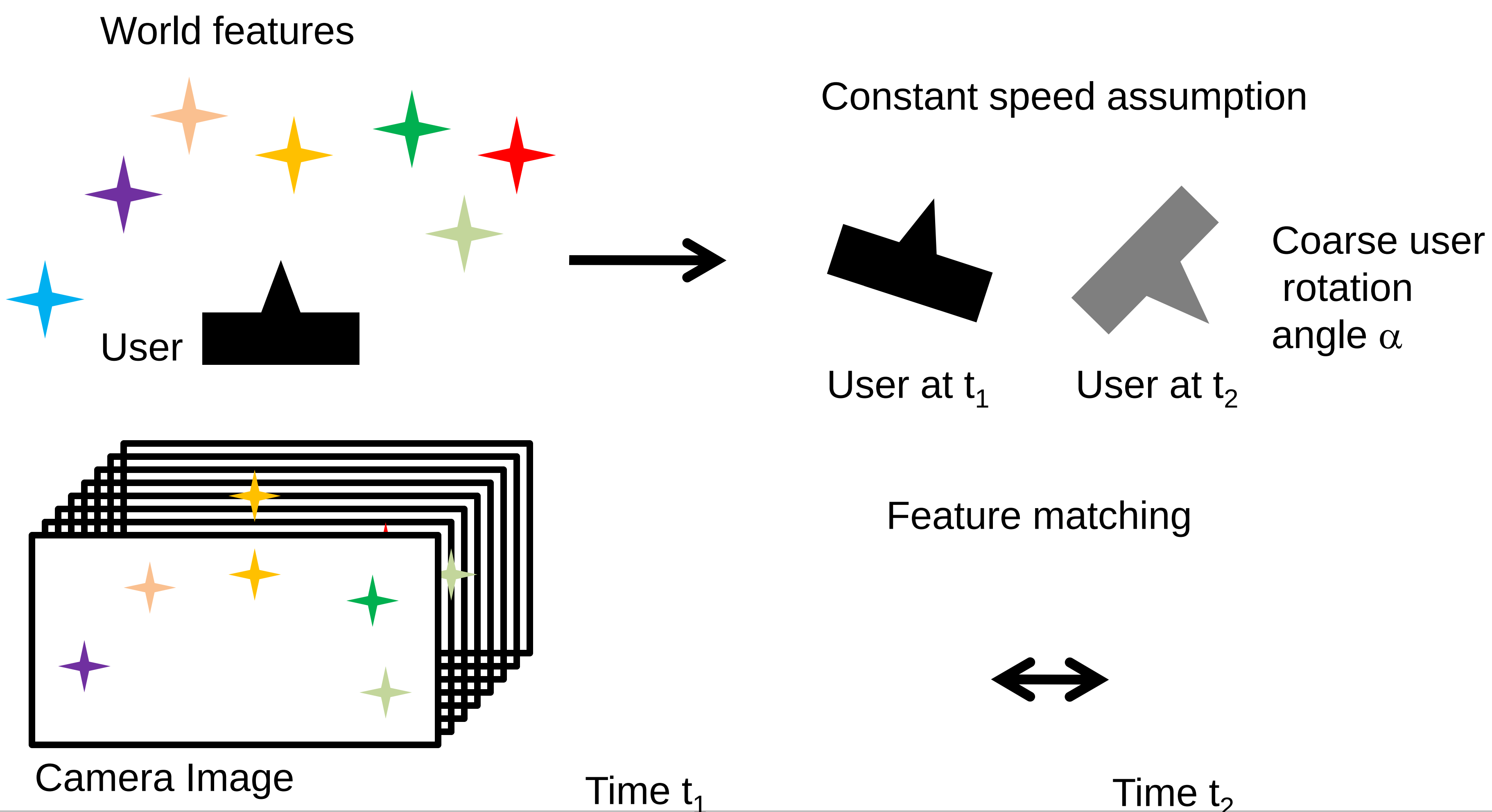
Input		**Output**
Calibration Video Sequence Coarse user motion	→	Classifier table

QUERY

Input		**Output**
Two features sets	→	Optimal user rotation bringing the two sets in alignment

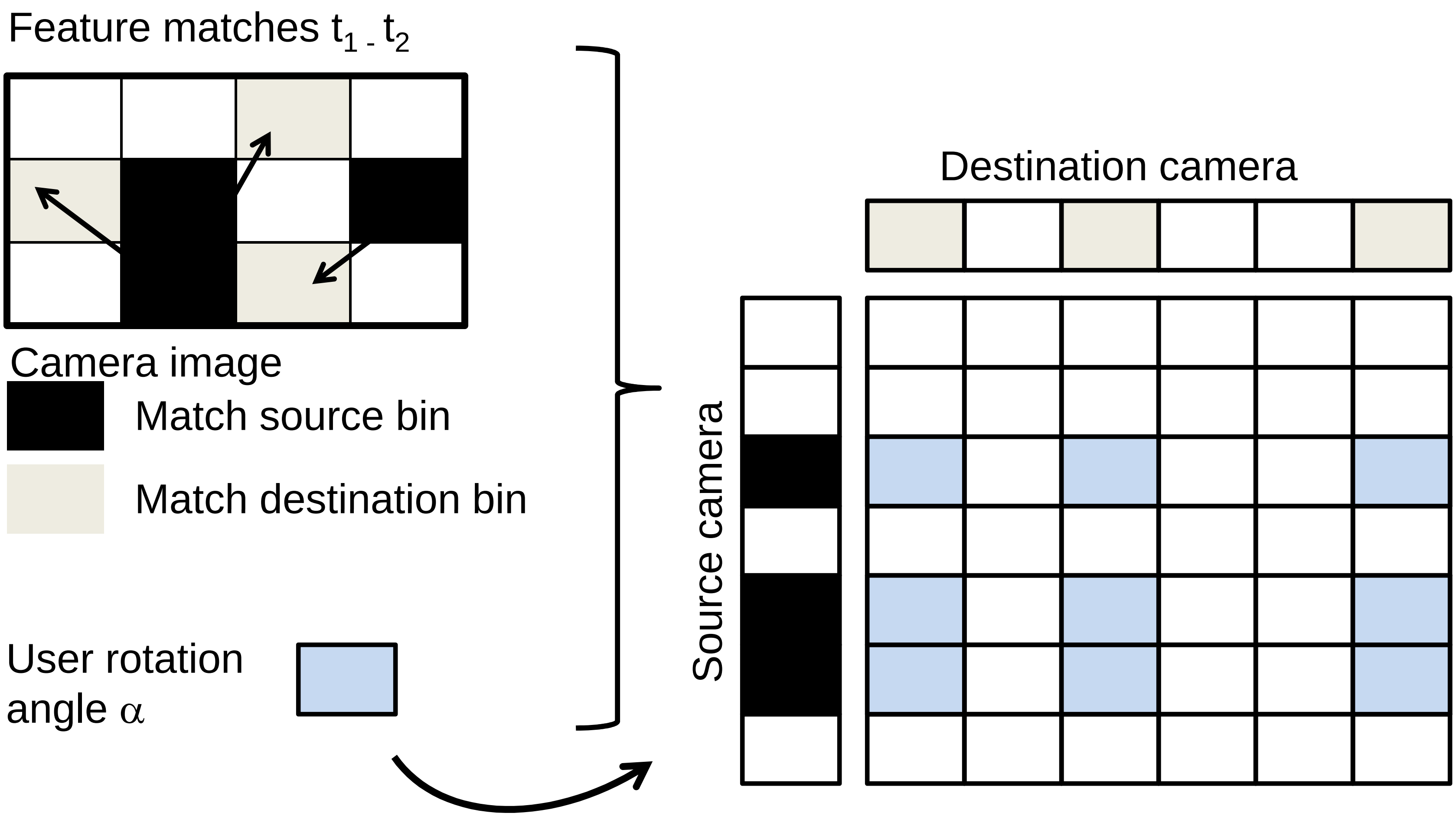
Rotation classifier

TRAINING



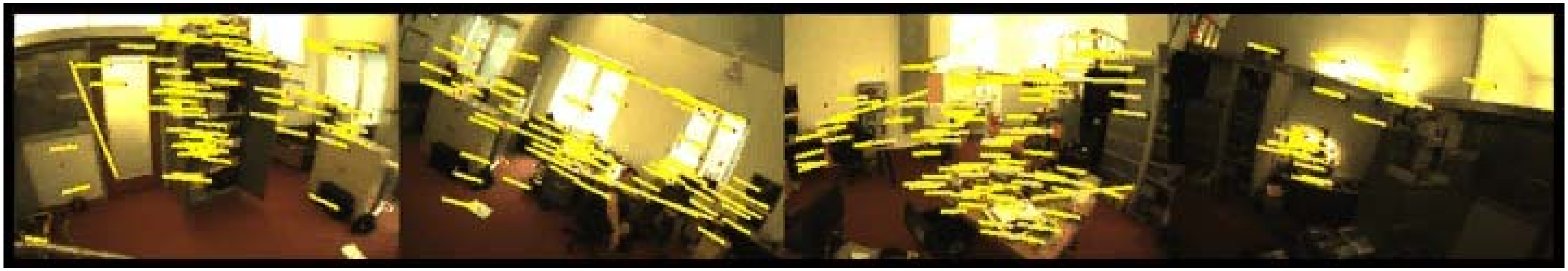
Rotation classifier

TRAINING

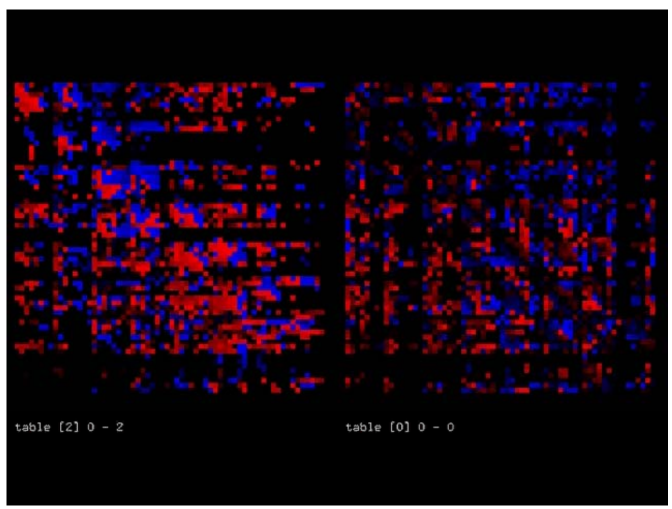


Rotation classifier

TRAINING



Calibration sequence - user rotates in place – 1 minute – 4 Hz – 240 frames



Red : angle > 0

Blue : angle < 0

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

Rotation classifier

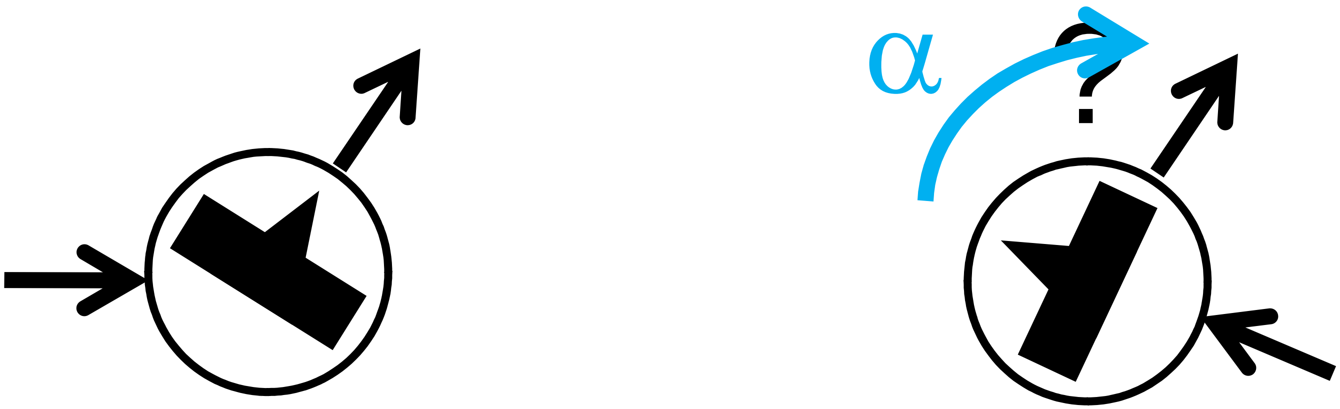
TRAINING

Input		**Output**
Calibration Video Sequence Coarse user motion	→	Classifier table

QUERY

Input		**Output**
Two features sets	→	Optimal user rotation bringing the two sets in 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 exit the node

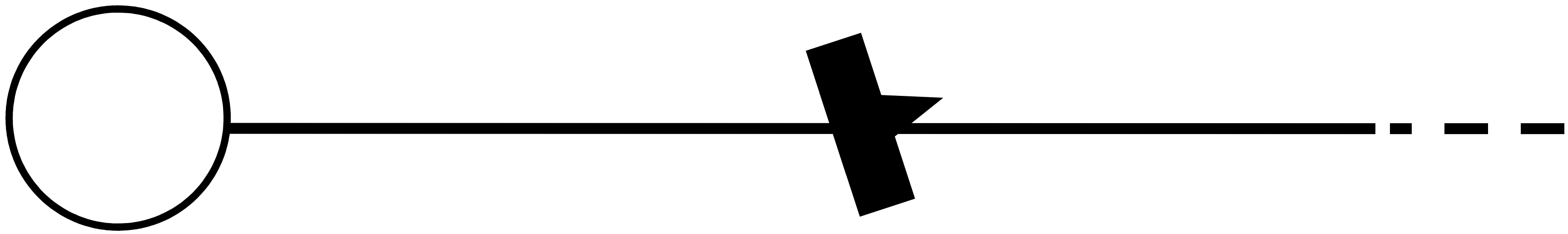
Method

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

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, 0 otherwise.

Navigation along edges

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

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

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

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

- **Observation update**

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

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

OUTDOOR Dataset
Kendall Square, Cambridge MA

INDOOR Dataset
MIT Underground

Navigation at node

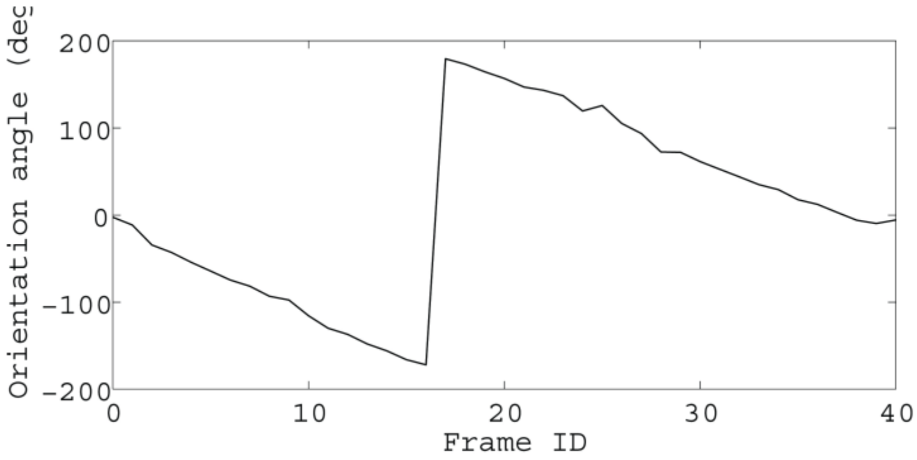


Fig. 1 - Rotation guidance output while user rotates in place in a new environment

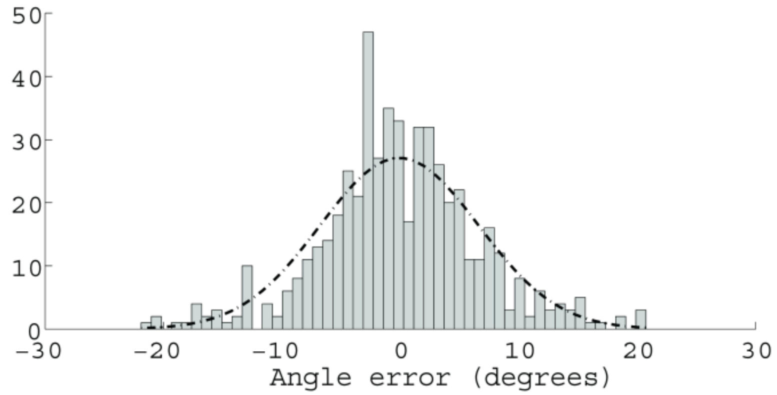


Fig. 2 – Error distribution against IMU-ground truth. **Standard deviation = 12 deg.**

Navigation along edges

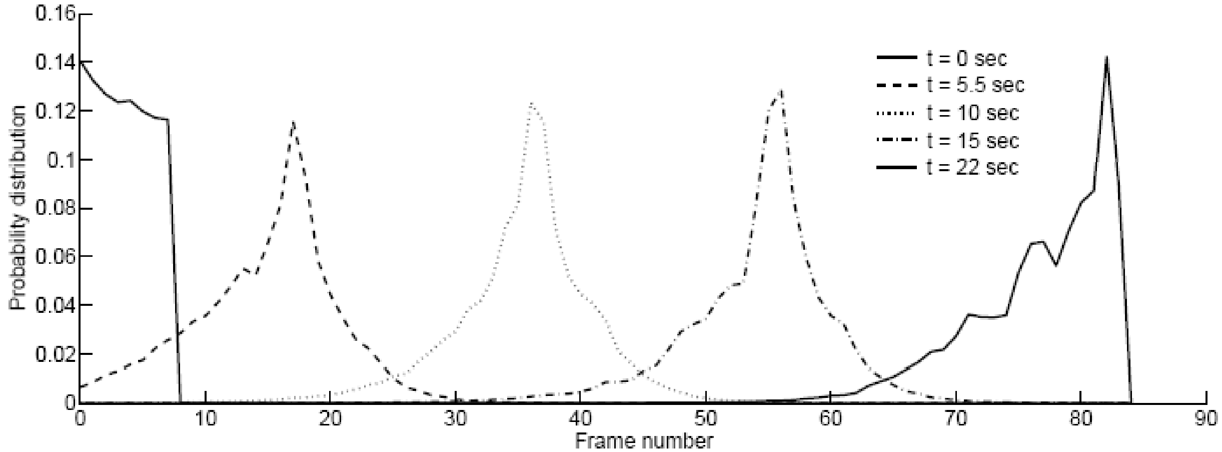


Fig 3 – belief state propagation while user walks along an edge (INDOOR dataset)

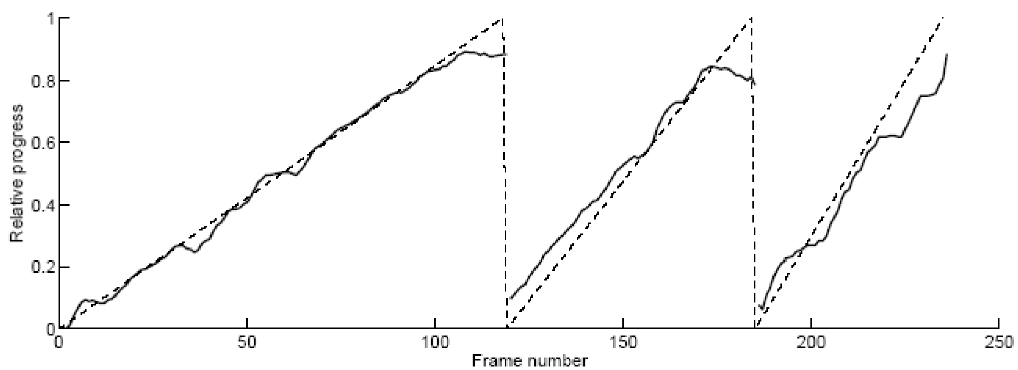


Fig 4 – Relative progress along several consecutive edges. Ground- truth estimated using constant speed assumption. **Standard deviation 3.3 frames (1 second, 1.5m)**

Topological map

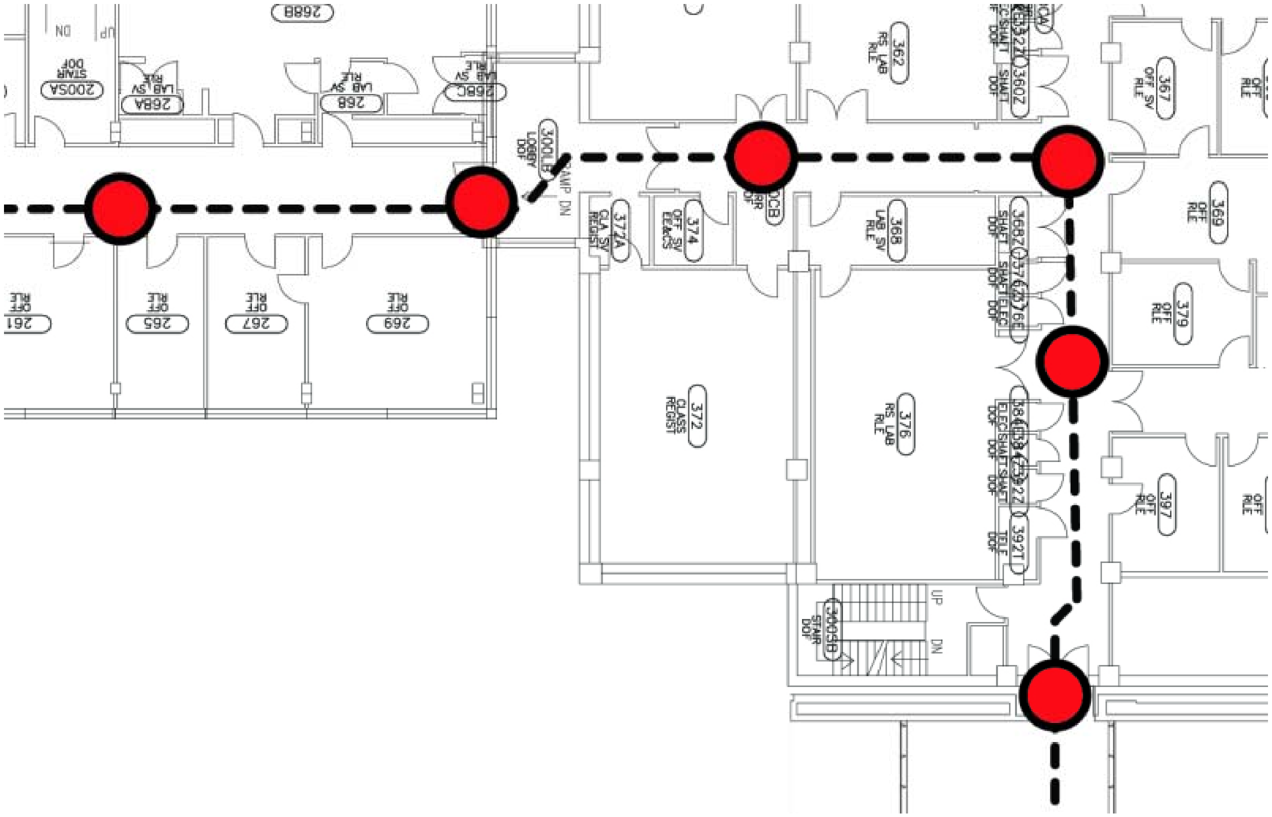


Fig 5 – Topological map automatically generated by the system (INDOOR dataset). Nodes overlaid on the map manually.

Summary

Input

- Video sequence
- Calibration sequence

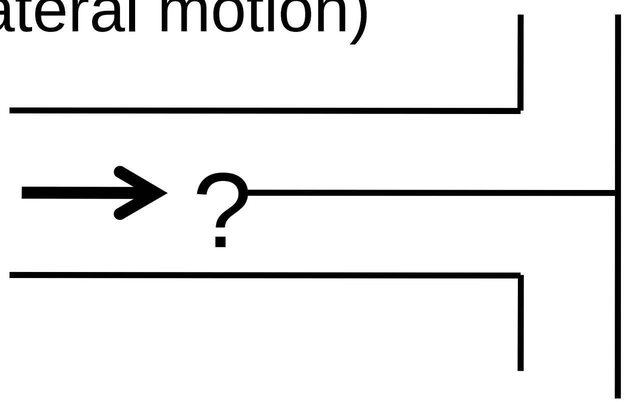
Output

- Backtrack along linear path
- Loose guidance in 2D

- No camera calibration
- No constraint on number of cameras or their relative position on the rig
- Requires rigid-body transformation between cameras to be fixed with some flexibility
- Provides loose guidance / imprecise directions
- New way of correlating user to image motion

Failure modes

- User leaving the exploration path
- Highly repetitive environments (tunnels)
- Significant change in lighting
- Dynamic scenes (crowd)
- Fast user motion (motion blur) or low lighting
- Ambiguous configurations (Y-shape)
- Handles only rotation along z-axis (lateral motion)



Future work

- Global localization
- User leaving the exploration path
- Path self-resection (non-linear graphs / loop closure)
- Augmented reality application (virtual tagging)