Vol-oriented Inference and Planning for Mapping and Navigation

Beipeng Mu and Jonathan P. How

Massachusetts Institute of Technology
**Problem:** Allocate heterogeneous sensors to search for targets of various importance. How to gather *most valuable information* given resource constraints?

- **Global Adaptive:** flexible in amount and location of resource allocation
- **Global Uniform:** allocate resource uniformly to the whole scene
- **Local Adaptive:** each sensor allocated to one location

**Approach:** Adaptive resource allocation using prior observations [1]
Outline

1. Motivation

2. Vol-based Inference for Navigation
   - Two-stage selection
   - Experiments

3. Vol-oriented Planning
   - Simultaneous Planning, Localization and Mapping
   - Topological Feature Graph
   - Planning for Information Gathering
   - Experiments

4. Ongoing Work
Trends in Robotics

- New sensing technologies yield “big data”
  - Cameras, cheaper laser/Velodyne
- Demands on robot capabilities
  - Complicated scenarios
    - self-driving, delivery, home-company, education
  - Environment features
    - nonlinear, discontinuous, high-dimension, non-stationary
  - Resource constraints
    - Computation, memory, communication, battery

- **Gap**: how covert large volume of sensor data to information of high value for real-time, resource-constrained systems
Data to Information

Importance of VoI:
- Data ≠ Information
  - Data often intractable, information tractable
- For real-time system need
  - Fast inference, fast planning, save resources

Problem: extract valuable information from big data to do real-time inference and planning for resource-constrained systems
1. Identify important variables
2. Quantify information in data
3. Build inference algorithms
4. Exploit information for planning

Contribution
- Two stage focused inference for mapping [3]
- Simultaneous planning, localization and mapping [4]
Data to Information

▶ Importance of VoI:
- Data $\neq$ Information
  - Data often intractable, information tractable
- For real-time system need
  - Fast inference, fast planning, save resources

▶ Problem: extract valuable information from big data to do real-time inference and planning for resource-constrained systems
- Identify important variables
- Quantify information in data
- Build inference algorithms
- Exploit information for planning

▶ Contribution
- Two stage focused inference for mapping [3]
- Simultaneous planning, localization and mapping [4] “SPLAM”
Vol-based Inference for Navigation

- Long-term navigation in unknown environment:
  - Mapped area grows $\Rightarrow$ feature space grows $\Rightarrow$ memory requirement grows
  - Temporal scale increases $\Rightarrow$ data grows $\Rightarrow$ computation complexity grows

Questions: What are important feature/info to keep?
- Depend on how much it can help mission

Approach: identify valuable features to tasks first then focus resources only on valuable features
- E.g. select focused features (landmarks for nav) that minimize $\text{Prob}_{\text{collision}}$
- Select data that maximize value of info on focused features
  - Much sparser model than prior work, but still good task performance
Vol-based Inference for Navigation

- Long-term navigation in unknown environment:
  - Mapped area grows $\rightarrow$ feature space grows $\rightarrow$ memory requirement grows
  - Temporal scale increases $\rightarrow$ data grows $\rightarrow$ computation complexity grows

- **Questions** What are important feature/info to keep?
  - Depend on how much it can help mission

- **Approach**: identify *valuable* features to tasks first then focus resources *only* on valuable features
  - E.g. select focused features (landmarks for nav) that min $\text{Prob}_{\text{collision}}$
  - Select data that maximize *value of info* on focused features
    $\Rightarrow$ Much sparser model than prior work, but still good task performance
Stage 1: Select focused features to minimize collision probability

- **Problem:** computer vision techniques give thousands of features, what are the valuable ones for collision-free navigation?
  - Exact collision probability in unmapped environments hard to known
  - Sampling methods computationally expensive
  - Approximate \( P_{\text{collision}} \) by robot pose uncertainty + distance to obstacle
- Robot pose uncertainty:
  - Pose covariance \( \Sigma(x) \) from steady state Riccati Eq.
  - Closest obstacle point \( x_{\text{obs}} \)
- Approximate collision probability
  - Mahalanobis dist: \( P(x) \sim \exp \left\{- (x - x_{\text{obs}})^T \Sigma^{-1}(x)(x - x_{\text{obs}})\right\} \)
- Select features that reduce maximal collision probability

\[
L_f = \min_{l \subset L} \left( \max_x P(x) \right)
\]

- \( L \) set of all features, \( L_f \) set of focused features
Stage 2: Select data to maximize info gain

▶ **Graphical model**: compact way of representing high-dimension model of robot poses $x$ and features $l$

\[ P(x, l|z^R) \sim \exp \left\{ \sum_c \phi_c(x_c, l_c) \right\} \]

- Factors $\phi_c(x_c, l_c)$: meas between robot poses, or pose and features
- Laplacian approximation \( P(x, l|z^R) \approx \mathcal{N}(\zeta, \Lambda_{z^R}) \)

▶ Select measurements $z^R$ out of all measurements $z$ to max info on focused features $L_f$

\[
\max_{z^R \subset z} H(L_f | z^R) \quad \text{s.t.} \quad g(z^R) \leq C
\]

- $H(\bullet)$ entropy, $g(\bullet) \leq C$ resource constraint
- **Closed-form** solution of $H(\bullet)$ exists for Laplacian approximation
VoI-focused inference: Simulation

Simulated, compared with select either feature or measurements only

- Green circles: selected features with their size representing uncertainty
- Blue lines: robot nominal trajectories
- Red circles: robot pose uncertainty.

Two-stage has less uncertainty in narrow passages and lower $\text{Prob}_{\text{collision}}$ compared to selecting feature or measurements separately.
Vol-focused inference: Simulation

- Simulated, compared with select either feature or measurements only

- Green circles: selected features with their size representing uncertainty
- Blue lines: robot nominal trajectories
- Red circles: robot pose uncertainty.
- Two-stage has less uncertainty in narrow passages and lower $\text{Prob}_{\text{collision}}$ compared to selecting feature or measurements separately

MC comparison of collision prob
Vol-oriented inference: Hardware

- Environment and robot

office environment, red(narrow), blue(wide)

Results

- only stage 2
- only stage 1
- Both stages
Simultaneous Planning, Localization and Mapping (SPLAM)

- **Occupancy grid map [5]**
  - State-of-art model for SPLAM
  - Easy to check path feasibility
  - Does not scale well, not robust, computation expensive, heavily relies on parameter tuning, depends on expensive sensor

- **Factor graphs [6]**
  - Scales well, robust to long-term drift
  - Easy to incorporate virtual data, semantic labels, sensor cheap
  - Hard to check path feasibility, thus typically not used for path planning

- **Gap**: Possible to maintain advantages of factor graphs, but how enable fast path feasibility check?
Simultaneous Planning, Localization and Mapping (SPLAM)

- **Occupancy grid map** [5]
  - State-of-art model for SPLAM
  - Easy to check path feasibility
  - Does not scale well, not robust, computation expensive, heavily relies on parameter tuning, depends on expensive sensor

- **Factor graphs** [6]
  - Scales well, robust to long-term drift
  - Easy to incorporate virtual data, semantic labels, sensor cheap
  - Hard to check path feasibility, thus typically not used for path planning

- **Gap**: Possible to maintain advantages of factor graphs, but how enable fast path feasibility check?
Simultaneous Planning, Localization and Mapping (SPLAM)

- **Occupancy grid map** [5]
  - State-of-art model for SPLAM
  - Easy to check path feasibility
  - Does not scale well, not robust, computation expensive, heavily relies on parameter tuning, depends on expensive sensor

- **Factor graphs** [6]
  - Scales well, robust to long-term drift
  - Easy to incorporate virtual data, semantic labels, sensor cheap
  - Hard to check path feasibility, thus typically not used for path planning

- **Gap**: Possible to maintain advantages of factor graphs, but how enable fast path feasibility check?
Topological Feature Graph (TFG)

- **Observations** [4]
  - Features themselves lack geometric information about obstacles ⇒ cannot check path feasibility
  - Features usually lie on obstacle surfaces
  - Can connect features to represent geometry information

- **Topological Feature graphs**

  \[ G(V, E) \]

  - \( V \), vertices, represent features
  - \( E \), edges, represent obstacle surfaces

- **Approach**
  - RGB image ⇒ features
  - Depth image ⇒ obstacle surfaces ⇒ edges between two features
Path Planning for Information Gathering

**Problem:** given a partial map, where should the robot go next to gather more information?

- Exploration: visit frontiers for new features
- Exploitation: revisit mapped places to improve existing feature accuracy

---

Theorem (Mu’15 [4])

*Given factor graph model, assume prior variance of unknown features significantly larger than that for known features, then total information gain (entropy reduction) is sum of exploration gain and exploitation gain*

---

**Quantify information gains**

- Given a partial map, how much info can robot gain by visiting particular place?
- Sample free-space, compute observable features, entropy reduction

---

Sample free space and compute observable features
Path Planning for Information Gathering

- Exploration and exploitation info gain of samples in free space

- Path Planning
  - Goal point: maximal total info gain
  - Probabilistic Roadmap (PRM)
Hardware Experiments – Indoors

Result comparison

- Grid Map
- Human Operator
- Topological Feature Graph

- Human operator has no good metrics for map quality, missed corners
- Grid map, order of magnitude more resources, not robust to disturbance
Hardware Experiments

Information-based Active SLAM via Topological Feature Graphs

{mubp, lpaull, aliagha, jhow, jleonard}@mit.edu

Laboratory for Information and Decision Systems
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Ongoing Work

**Goal:** demonstrate in real environment with natural objects not artificial tags

Robot work with natural objects as features

**Challenges**

- **Vision** *Real-time* object detection, subject to lighting, occlusion, perspective variance etc.
- **Modeling** inference algorithms on graphs with noisy node/edge identification, unreliable data association, multi-modal hypothesis
- **Planning** explicitly account for multi-modal noise, data association inaccuracies


