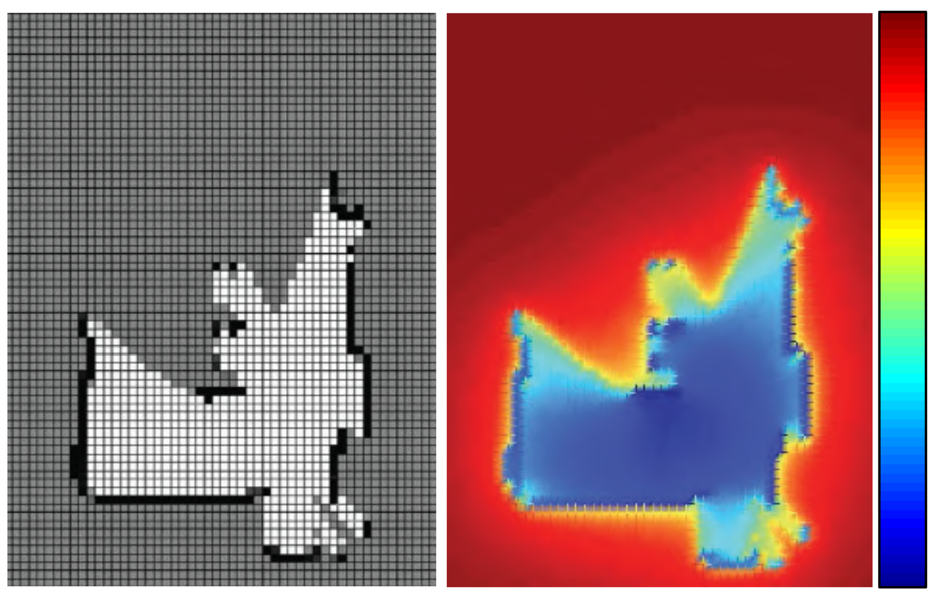
FSMI: Fast computation of Shannon Mutual Information for information theoretic mapping

Zhengdong Zhang, Trevor Henderson, Vivienne Sze, Sertac Karaman
Massachusetts Institute of Technology

Problem Statement

Robotic Exploration Problem: Where should the robot move next to learn most about the map?



Occupancy Grid   Mutual Information

Theoretically Proven Approach: Move to the location that maximizes the expected information gain (mutual information) between prospective range measurements and the map.

$$H(M|Z) = H(M) - I(M;Z)$$

Prospective updated map entropy — Current map entropy — Mutual information

Challenge: Existing algorithms for **computing the mutual information $I(M;Z)$** between prospective range measurements Z and the occupancy map M have **high computational complexity**.

Assumptions and Definitions

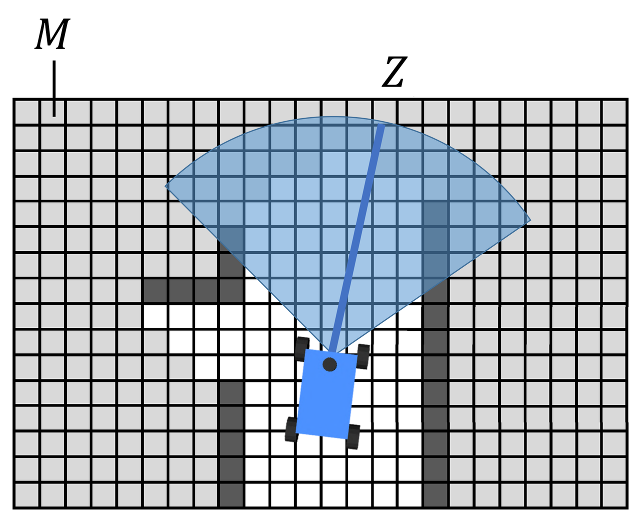
- The map M is an **occupancy grid** consisting of cells with occupancy probabilities o_i, updated using a **Bayesian filter**. The occupancy probabilities are assumed to be **independent**.
- The range measurement Z has **Gaussian noise**.
- The inverse sensor model: $\delta_i(z) = \begin{cases} \delta_e < 1 & \text{z indicates cell i empty} \\ \delta_o > 1 & \text{z indicates cell } i \text{ occupied} \\ 1 & \text{otherwise} \end{cases}$

The mutual Information $I(M;Z)$ between range measurement Z and map M

[Julian et al., IJRR 2014]

$$I(M;Z) = \sum_{i=1}^{n} \int P(Z=z) f(\delta_i(z), o_i) dz$$

$$\approx \sum_{i=1}^{n} \sum_{z} P(Z=z) f(\delta_i(z), o_i) \lambda_z^{-1}$$



The integral has no closed form solution. Requires expensive **numerical integration at resolution λ_z**

The existing algorithm for computing the mutual information $I(M;Z)$ of a range measurement of length n runs in time $O(n^2\lambda_z)$

Main Result: The FSMI Algorithm

$$I(M;Z) = \sum_{j=1}^{n} \sum_{k=1}^{n} P(e_j) C_k G_{k,j}$$

Under this formulation, $I(M;Z)$ can be computed exactly in $O(n^2)$

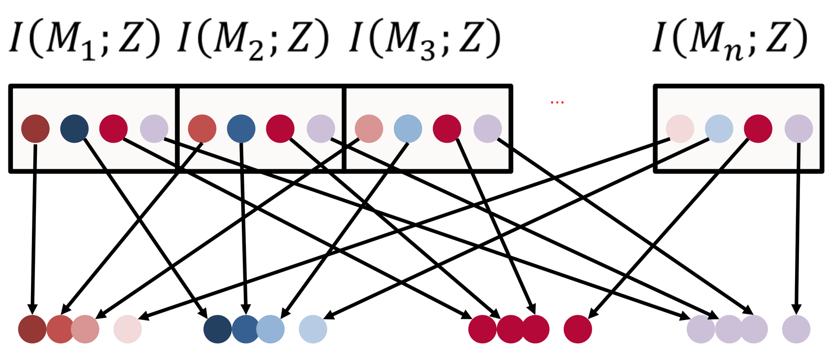
where

$$P(e_i) = o_i \prod_{j<i} (1-o_j) \quad i \in [1,n] \qquad O(n)$$

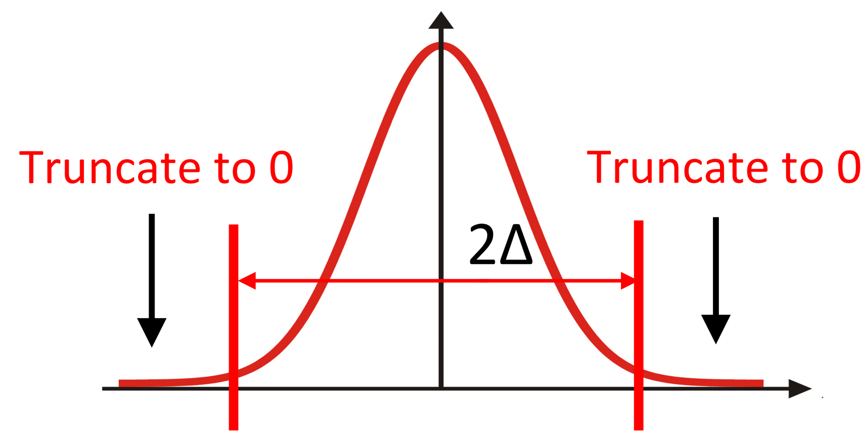
$$C_k = f(\delta_0, o_k) + \sum_{i<k} f(\delta_e, o_i) \quad k \in [1,n] \qquad O(n)$$

$$G_{k,j} = \int_{l_k}^{l_{k+1}} P(z|e_j) dz = \Phi_j(l_{k+1}) - \Phi_j(l_k) \qquad O(1)$$

Key Insight: The mutual information is evaluated along an entire beam rather than summed over individual cells.



Approximation Techniques



Truncate the Normal Distribution

With truncation*, $I(M;Z)$ can be computed in $O(n)$

*also explored in Charrow et al., ICRA 2015

$$I(M;Z) = \sum_{j=1}^{n} \sum_{k=j-\Delta}^{j+\Delta} P(e_j) C_k G_{k,j}$$

Δ can be as small as 3 or 5

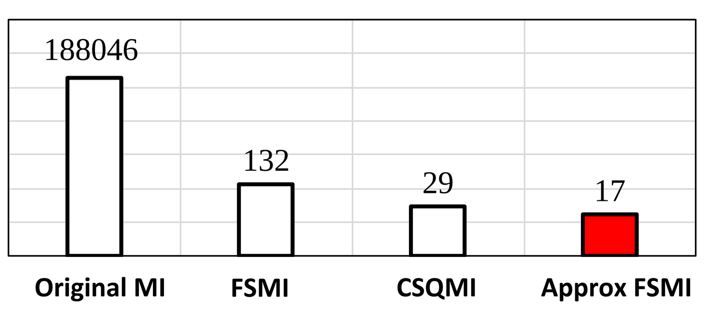
If the sensor noise is uniform, $I(M;Z)$ can be computed *exactly* in $O(n)$.

Experimental Results

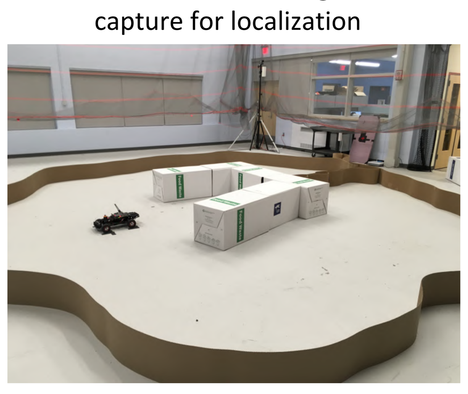
Original MI[1]	FSMI	CSQMI[2]	Approx FSMI
$O(n^2\lambda_z)$	$O(n^2)$	$O(n)$	$O(n)$

[1] Julian et al., IJRR 2014; [2] Charrow et al., ICRA 2015

Measured run time per beam (μsec) on an **Intel Xeon core (desktop)**



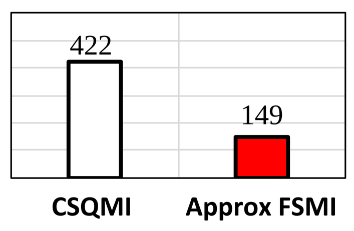
Exploration using FSMI demonstrated with a mini race car using motion capture for localization

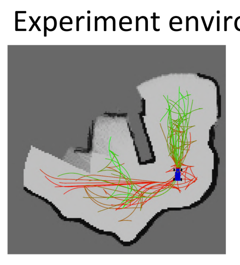


Experiment environment

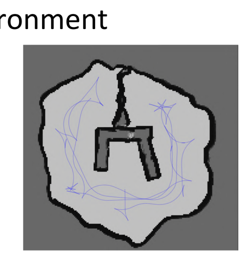
Approx-FSMI is **over 10,000x faster** than the original algorithm and **1.7 – 2.8x faster** than CSQMI

Measured run time per beam (μsec) on an **ARM Cortex-A57 core (embedded)**





Paths with high MI/meter in green

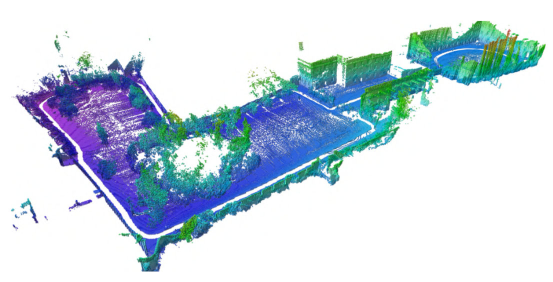


Complete map trajectory

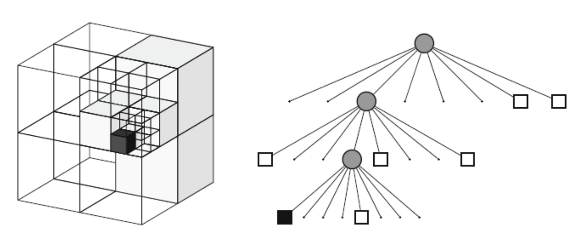
FSMI on a chip: An FPGA implementation of FSMI can compute the per pixel mutual information of an **entire 200 x 200 pixel map** in **under a second** while consuming **under 2W (to appear at RSS 2019[3])**

Extending FSMI to 3D Environments

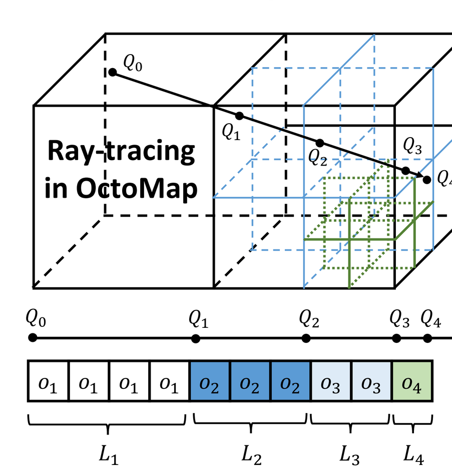
Computing MI on a **3D map** requires a significant amount of storage and computation



Compress the 3D map with OctoMap
[Hornung, et al., Autonomous Robots, 2013]



Compute FSMI on the Compressed 3D Map



Uncompressed input format

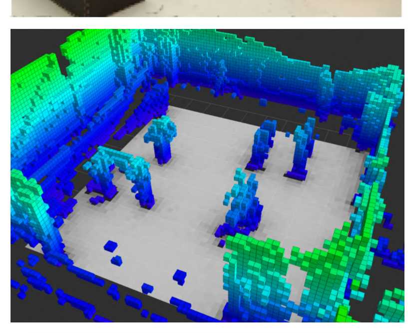
$$\underbrace{o_1, o_1, \dots, o_1}_{L_1}, \underbrace{o_2, \dots, o_2}_{L_2}, \underbrace{o_3, \dots, o_3}_{L_3}, \dots, \underbrace{o_{n_r}, \dots, o_{n_r}}_{L_{n_r}} \qquad L_1 + L_2 + L_3 + L_{n_r} = n$$

Compressed format (Run Length Encoding)

$$\underbrace{(o_1, L_1), (o_2, L_2), (o_3, L_3), \dots, (o_{n_r}, L_{n_r})}_{n_r}$$

$n_r \ll n$, significant reduction if the constants are comparable $O(n) \rightarrow O(n_r)$

A 3D environment featuring an arch, a giant cat, a box and a tree



3D OctoMap where color indicates the height of a voxel

Computing FSMI directly on a 3D OctoMap achieves an **acceleration ratio of 8×**

Z. Zhang et al., FSMI: Fast computation of Shannon Mutual Information for information-theoretic mapping, *arXiv 2019*
http://arxiv.org/abs/1905.02238

[1] Julian et al., On mutual information-based control of range sensing robots for mapping applications, IJRR 2014
[2] Charrow et al., Information-Theoretic Planning with Trajectory Optimization for Dense 3D Mapping, RSS 2015
[3] Li et al., High-Throughput Computation of Shannon Mutual Information on Chip, RSS 2019