Beyond Point Clouds: Fisher Information Field for Active Visual Localization

Zichao Zhang, Davide Scaramuzza

***Abstract*— For mobile robots to localize robustly, actively considering the perception requirement at the planning stage is essential. In this paper, we propose a novel representation for active visual localization. By formulating the Fisher information and sensor visibility carefully, we are able to summarize the localization information into a discrete grid, namely the *Fisher information field*. The information for arbitrary poses can then be computed from the field in *constant* time, without the need of costly iterating all the 3D landmarks. Experimental results on simulated and real-world data show the great potential of our method in efficient active localization and perception-aware planning. To benefit related research, we release our implementation of the information field to the public.**

Supplementary Material

Video: https://youtu.be/q3YqIyaFUVE
Code: https://github.com/uzh-rpg/rpg_information_field

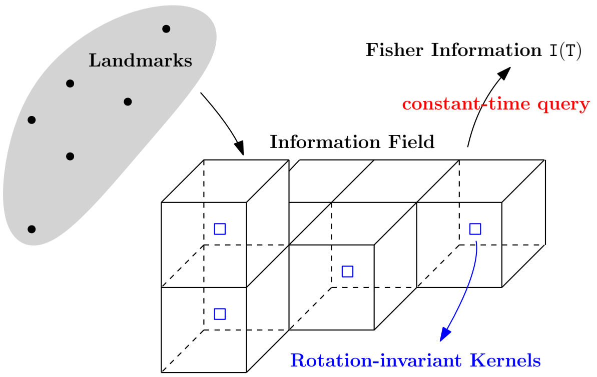


Fig. 1: Illustration of the proposed Fisher information field. The gray cloud denotes the 3D landmarks in the environment. For each voxel (black cubes), the building process summarizes the rotation-independent information kernels (19) or (21) (blue squares). Then the information of an arbitrary pose T can be computed in *constant time* without accessing the original 3D landmarks.

I. Introduction

On-board visual sensing and computing permits robots to operate autonomously, but brings additional constraints to motion planning algorithms. Specifically, the robot motion impacts the information that will be captured by the cameras and thus influences the performance of perception algorithms. Therefore, the requirement of visual perception has to be taken into consideration in motion planning. This is known as *active vision* [1]. In this paper, we are particularly interested in the problem of *active localization*, which aims to planning the sensor motion to maximize the localization accuracy with respect to a given map.

Active localization, or more generally *active simultaneous localization and mapping (SLAM)*, is still an open research problem, and one major paradigm is to plan the sensor motion based on the information/covariance that can be achieved. To analyze the attainable information for cameras, the environment is usually represented as a point cloud, in which each point stands for a 3D landmark. *We argue that, using a point cloud to represent 3D landmarks is a convenient choice, but not necessarily the most efficient for active localization.* First, to evaluate the localization quality of a single pose, one needs to evaluate the information for all the points (see Section II-B), the complexity of which increases *linearly* with the number of landmarks. Second, this process has to be repeated many times in both sampling-based (i.e., evaluating motion samples) and optimization-based methods (i.e., optimization iterations), which introduces redundant computation, especially when the planning is performed multiple times in the same environment.

Using a volumetric representation which stores the quantities of interest as a function of 3D positions is a potential solution to the aforementioned problems. For example, the distance field summarizes the distance from obstacles and, without extra knowledge, contains sufficient information for collision avoidance (e.g., [2]). However, applying the similar idea to active localization exhibits one major difficulty: the localization quality additionally depends on the sensor orientation due to the fact that the visibility of landmarks can vary drastically with orientations. The naive solution of discretizing the rotation space would lead to the explosion of required storage. In this work, we propose a novel volumetric representation for localization information, namely the *Fisher information field* (or *information field* for short), which overcomes this difficulty. The key idea is that, for each discrete spatial unit (namely a *voxel*) in the field, we summarize a rotation-independent component of the Fisher information, which is a valuable tool to quantify parameter estimation quality (see Section II-A), from all the 3D landmarks and store it in the voxel. We can then recover the information (under some approximation) of an arbitrary 6 degree-of-freedom (DoF) pose by applying a linear transformation to the stored information in the field, which is of *constant* time complexity instead of linear, as illustrated in Fig. 1. Moreover, the information field

The authors are with the Robotics and Perception Group, Dep. of Informatics, University of Zurich , and Dep. of Neuroinformatics, University of Zurich and ETH Zurich, Switzerland— http://rpg.ifi.uzh.ch.
This research was supported by the National Centre of Competence in Research (NCCR) Robotics, through the Swiss National Science Foundation and the SNSF-ERC Starting Grant.

can be reused for multiple planning sessions and easily updated when landmarks are added to or deleted from the environment.

To demonstrate the practical value of our proposal, we show that the information field can be built in a batch processing fashion or incrementally (e.g., from the continuous output of a SLAM system). Furthermore, we show that it can be used to determine the optimal orientation efficiently.

A. *Related Work*

Considering perception performance in planning has been extensively studied in different contexts. Early works include maximizing the Fisher information (or equally minimizing the covariance) about the robot state and the map in navigation tasks [3], [4], minimizing the entropy of the robot state in known environments [5], [6], and actively searching features in SLAM systems [7]. Recently, with the advance of drones, several works have been done to couple perception, planning and control on agile aerial platforms [8]–[14].

Despite the extreme diversity of the research in this topic, related work can be categorized based on the method to generate motion profiles. One paradigm used sampling-based methods, which discretize the space of possible motions and find the optimal one in a discrete set. Roy *et al.* [6] used the Dijkstra's algorithm to find the path on a grid that minimizes a combined cost of collision and localization. Papachristos *et al.* [11] and Costante *et al.* [15] adapted the rapidly-exploring random tree (RRT) algorithms to incorporate the perception cost, and the latter additionally considered the photometric property of the environment. Zhang *et al.* [12] proposed to evaluate motion primitives against multiple costs, including the localization uncertainty, in a receding horizon fashion. Instead of a combined cost, as in most of previous works, Ichter *et al.* [16] used multi-objective search for perception-aware planning.

Alternatively, researchers have explored to plan in the continuous motion space. Indelman *et al.* [17] considered optimizing the motion within a finite horizon to minimize a joint cost including the final pose covariance, which was later extended to visual-inertial sensing and self-calibration in [18]. Watterson *et al.* [13] studied the general problem of trajectory optimization on manifolds and applied their method to planning under the field-of-view (FoV) constraint of the camera. Falanga *et al.* [14] tackled the problem at the controller level by incorporating related costs in model predictive control (MPC).

In the above methods, calculating the perception related cost/metric is a crucial part and often the computational bottleneck (e.g., [4]). Unfortunately, little work has been done in developing dedicated representations for efficient computation. Roy *et al.* [6] pre-computed and stored the information in a 2D grid, but their method was limited to 360° FoV sensors. Specifically, the visual information (e.g., visibility) are invariant regardless of the camera orientation for omnidirectional sensors, and thus their map did not need to consider the impact of orientations, which is not true for cameras with limited FoVs. More recently, Ichter *et al.* [16] trained a neural network to predict the state estimation error and generated a map of perception cost using the network prediction. However, their map only contains the averaged cost of different orientations and, therefore, cannot be used to evaluate the cost of an arbitrary 6 DoF pose. In contrast, our method explicitly models the FoV constraint and can represent the information of 6 DoF poses efficiently.

B. *Contributions and Outline*

The contribution of this work is twofold:

- We propose a novel representation for the information of 6 DoF visual localization, which enables efficient computation of the Fisher information compared with the standard method of using a point cloud. To the best of our knowledge, this is the first dedicated representation for such tasks.
- We make our implementation of the Fisher information field open source to benefit the research community.

The rest of this paper is structured as follows. In Section II, we briefly introduce the Fisher information matrix and its application in active localization, from which we identify the computational bottleneck of using a point cloud. In Section III, we show the separation of the rotation-independent kernels from the approximated Fisher information matrix, which enables a volumetric representation. In Section IV, we describe the method to build and update the proposed information field and introduce an alternative formulation to reduce the memory usage. Finally, we show experimental results from simulated and real-world data in Section V.

II. Fisher Information and Active Localization

A. *Fisher Information Matrix*

For a general parameter estimation problem, the Fisher information matrix (FIM) summarizes the information that the observations carry about the parameters to be estimated. To put it formally, if the measurement process can be described as a conditional probability density function $\mathrm{p}(\mathbf{z}|\mathbf{x})$, where $\mathbf{z}$ is the measurement and $\mathbf{x}$ the parameters, one definition[1] of the Fisher information is

$$\mathtt{I}_{\mathbf{x}}(\mathbf{z}) = (\frac{\partial}{\partial \mathbf{x}} \log \mathrm{p}(\mathbf{z}|\mathbf{x}))^\top (\frac{\partial}{\partial \mathbf{x}} \log \mathrm{p}(\mathbf{z}|\mathbf{x})). \tag{1}$$

With identical and independent zero-mean Gaussian noise $\mathcal{N}(0, \sigma^2)$ on the measurement, (1) can be written as

$$\mathtt{I}_{\mathbf{x}}(\mathbf{z}) = \frac{1}{\sigma^2}(\mathtt{J}_{\mathbf{x}})^\top \mathtt{J}_{\mathbf{x}}, \quad \text{where } \mathtt{J}_{\mathbf{x}} = \frac{\partial \mathbf{z}}{\partial \mathbf{x}}. \tag{2}$$

Note that in practice (1) and (2) are usually evaluated at the estimate $\mathbf{x}^*$ instead of the unknown true value $\mathbf{x}$.

The Fisher information is a pivotal concept in parameter estimation problems. Most notably, the inverse of the FIM defines the Cramér-Rao lower bound, which is the smallest covariance (in terms of Loewner order) that can be achieved by an unbiased estimator [20, App. 3.2] [21, p. 14]. Note that the widely used nonlinear maximum likelihood estimator

[1] The presented definition is the *observed Fisher information*. See [19] for the comparison of different concepts.

(MLE) is in general biased, but the bias also tends to decrease when the Fisher information increases [22]. Due to its rich theoretical implications, the FIM is widely used in different applications, such as optimal design of experiments [23], active SLAM [17] and feature selection [24].

B. Using Fisher Information for Active Localization

Active localization aims to find the motion for a robot in a known environment such that the localization accuracy during the motion can be optimized. Without the loss of generality, we denote the motion as a continuous time function $f(t;\mathbf{m})$, parameterized with $\mathbf{m}$. The output of the function is the 6 DoF pose of the camera at a given time. Then, an active localization algorithm can be formulated to solve

$$\mathbf{m}^* = \arg\min_{\mathbf{m}} \mu_{\mathrm{v}} C_{\mathrm{v}}(f(t;\mathbf{m})) + \mu_{\mathrm{o}} C_{\mathrm{o}}(f(t;\mathbf{m})), \tag{3}$$

where C_{v} is the cost related to visual localization, C_{o} denotes the other cost terms collectively (e.g., collision and execution time) and $\mu_{\mathrm{v}}/\mu_{\mathrm{o}}$ are the corresponding weights. Since localization can be viewed as the estimation of the poses of interest, FIM can be used to quantify the estimation error and, thus, the localization quality. Evaluating the cost using M discrete samples, we have

$$C_{\mathrm{v}} = -s(\begin{bmatrix} \mathtt{I}_{\mathtt{T}_1} & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \mathtt{I}_{\mathtt{T}_M} \end{bmatrix}), \quad \mathtt{I}_{\mathtt{T}_i} = \sum_{k}^{k\in V_i} \mathtt{I}_{\mathtt{T}_i}(\mathbf{u}_{ik}) \tag{4}$$

where $\mathtt{T}_i$ is the ith sample, V_i is the index set of visible landmarks in $\mathtt{T}_i$, and $\mathbf{u}_{ik}$ is the projection of the kth landmark in $\mathtt{T}_i$. $s(\cdot)$ is a metric function that converts the information matrix into a scalar (e.g., determinant).

(3) can be solved using sampling-based methods, such as RRT [11] and motion primitives [12], or optimization-based methods [17]. Either way, the FIMs for individual poses in (4) need to be computed multiple times for different motion samples or the iterations in optimization, which is the computational bottleneck for solving (3). Specifically, the calculation of $\mathtt{I}_{\mathtt{T}_i}$ requires iterating all the landmarks in the environment and evaluating the individual FIM for all the visible ones (the sum in (4)), which scales linearly with the number of landmarks. Moreover, $\mathtt{I}_{\mathtt{T}_i}$ needs to be recomputed from scratch once the pose $\mathtt{T}_i$ changes (both the visibility and the Jacobian in (2) change), even if there can be some invariant components that only need to be computed once. This motivates us to look for an alternative formulation of (4) to mitigate the bottleneck.

It is worth mentioning that, compared with complete probabilistic treatment as in [5], [6], we make the simplification in the problem formulation (3) (4) that the localization process purely depends on the measurements (i.e., no prior from the past). However, this is not a limitation of our work. The computational bottleneck exists as long as the Fisher information is used to characterize the visual estimation process. The essence of this work is a compact representation of the information to allow efficient computation, which is widely applicable.

III. Rotation Separation and Invariant Kernels

In this section, we focus on the formulation of the Fisher information for a single pose, since the FIMs of different poses are calculated independently in the same way. Let $\mathtt{T}_{\mathtt{WC}} = \{\mathtt{R}_{\mathtt{WC}}, \mathbf{t}_{\mathtt{WC}}\}$ stands for the pose of the camera in the world frame, $\{\mathbf{p}_k^{\mathtt{W}}\}_{k=1}^N$ the 3D landmarks in the world frame and $\mathtt{I}_k$ the information matrix corresponding to the observation of the kth landmark. The FIM for the pose can be written as

$$\mathtt{I}_{\mathtt{T}_{\mathtt{WC}}} = \sum_{k=1}^{N} v(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}) \mathtt{I}_i, \tag{5}$$

where $v(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}})$ is a binary valued function indicating the visibility of the ith landmark. Conceptually, our goal is to find an approximation $\mathtt{S}(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}) \approx v(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}})\mathtt{I}_i$ that can be written as $\mathtt{S}(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}) = \mathtt{S}(\mathtt{H}(\mathbf{t}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}), \mathtt{R}_{\mathtt{WC}})$ and satisfies

$$\mathtt{I}_{\mathtt{T}_{\mathtt{WC}}} \approx \sum_{i=1}^{N} \mathtt{S}(\mathtt{T}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}) = \mathtt{S}(\sum_{i=1}^{N} \mathtt{H}(\mathbf{t}_{\mathtt{WC}}, \mathbf{p}_i^{\mathtt{W}}), \mathtt{R}_{\mathtt{WC}}). \tag{6}$$

In words, we would like to find an approximation that can be factored into two components, one of which does not depend on rotation (i.e., $\mathtt{H}(\cdot)$ in (6)), and the approximation is linear in terms of the rotation-independent part. The linear form lead to two favorable properties. First, for one position $\mathbf{t}_{\mathtt{WC}}$, the sum of the rotation-independent $\mathtt{H}(\cdot)$ of all the landmarks need to be computed *only once*, and the sum can be used to calculate the approximated information at this position for arbitrary rotations; second, we can easily update the sum when new landmarks are added or old ones deleted. This form naturally leads to a volumetric representation that allows online update, as described in Section IV.

The approximation (6) is achieved by first carefully parameterizing the information matrix $\mathtt{I}_i$ to be rotation-invariant (Section III-A) and replacing the binary valued function $v(\cdot)$ with a smooth alternative (Section III-B).

A. Rotation Invariant FIM

The observation of a 3D landmark can be represented in different forms, such as (normalized) pixel coordinates and bearing vectors. In this work, we choose to use the bearing vector $\mathbf{f}$ because of its ability to model arbitrary FoVs. Then the noise-free measurement model of a landmark $\mathbf{p}_i^{\mathtt{W}}$ is

$$\mathbf{f}_i = \frac{\mathbf{p}_i^{\mathtt{C}}}{\|\mathbf{p}_i^{\mathtt{C}}\|_2} = \frac{1}{n_i}\mathbf{p}_i^{\mathtt{C}}, \quad \mathbf{p}_i^{\mathtt{C}} = \mathtt{T}_{\mathtt{CW}}\mathbf{p}_i^{\mathtt{W}}, \tag{7}$$

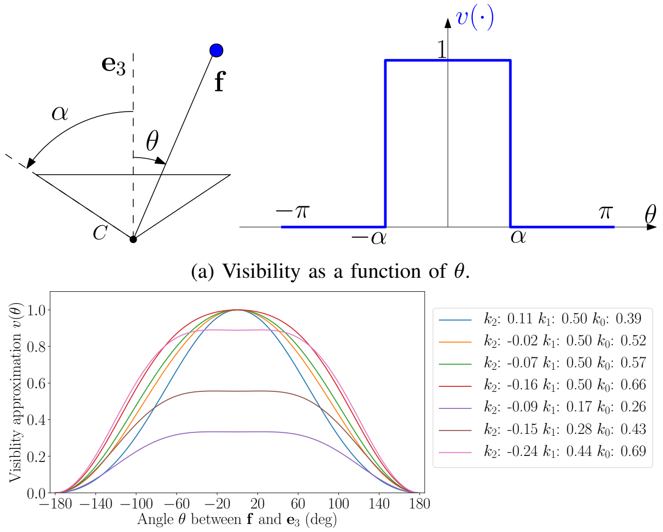
and the Jacobian of interest is

$$\mathtt{J}_i = \frac{\partial \mathbf{f}_i}{\partial \mathbf{p}_i^{\mathtt{C}}} \frac{\partial \mathbf{p}_i^{\mathtt{C}}}{\partial \mathtt{T}_{\mathtt{WC}}}. \tag{8}$$

While the first part in (8) is trivially

$$\frac{\partial \mathbf{f}_i}{\partial \mathbf{p}_i^{\mathtt{C}}} = \frac{1}{n_i}\mathcal{I}_3 - \frac{1}{n_i^3}\mathbf{p}_i^{\mathtt{C}}(\mathbf{p}_i^{\mathtt{C}})^\top, \tag{9}$$

the derivative $\frac{\partial \mathbf{p}_i^{\mathtt{C}}}{\partial \mathtt{T}_{\mathtt{WC}}}$ is more involved. To handle the derivatives related to 6 DoF poses without overparametrization, the



(a) Visibility as a function of θ.

(b) Polynomial model (13).

Fig. 2: The approximation of visibility function $v(\cdot)$. (a) α is half of the FoV, $\mathbf{f}$ is the bearing vector observation, $\mathbf{e}_3$ is the optical axis of the camera, and C is the projection center. (b) Examples of the polynomial model (13).

element in $\mathfrak{se}(3)$ (denoted as $\boldsymbol{\xi}$) is often used. In our case, $\frac{\partial \mathbf{p}_i^{\mathtt{c}}}{\partial \mathtt{T}_{\mathtt{wc}}}$ is replaced by

$$\frac{\partial \mathbf{p}_i^{\mathtt{c}}}{\partial \mathtt{T}_{\mathtt{wc}}} \rightarrow \frac{\partial(\exp(\boldsymbol{\xi}^\wedge)\mathtt{T}_{\mathtt{wc}})^{-1}\mathbf{p}_i^{\mathtt{w}}}{\partial \boldsymbol{\xi}} \text{ or } \frac{\partial(\mathtt{T}_{\mathtt{wc}}\exp(\boldsymbol{\xi}^\wedge))^{-1}\mathbf{p}_i^{\mathtt{w}}}{\partial \boldsymbol{\xi}}, \quad (10)$$

where $\exp(\cdot)$ is the exponential map of the Special Euclidean group SE(3). The two forms corresponds to expressing the perturbation $\delta\boldsymbol{\xi}$ globally in the world frame or locally in the camera frame respectively. Using the first form, we have the Jacobian in (8) as

$$\mathtt{J}_i = \frac{\partial \mathbf{f}_i}{\partial \mathbf{p}_i^{\mathtt{c}}} \mathtt{R}_{\mathtt{cw}}[-\mathcal{I}_3, \quad [\mathbf{p}_i^{\mathtt{w}}]_\times]. \quad (11)$$

With the global perturbation formulation, for two poses that differ by a relative rotation $\mathtt{T}_{\mathtt{wc}}$ and $\mathtt{T}_{\mathtt{wc'}} = \{\mathtt{R}_{\mathtt{wc}}\mathtt{R}_{\mathtt{cc'}}, \mathbf{t}_{\mathtt{wc}}\}$, their Jacobians (11) have a simple relation $\mathtt{J}'_i = \mathtt{R}_{\mathtt{c'c}}\mathtt{J}_i$, from which the corresponding FIMs turn out to be the same

$$\mathtt{I}'_i \overset{(2)}{=} \frac{1}{\sigma^2}\mathtt{J}_i^\top \mathtt{R}_{\mathtt{cc'}}\mathtt{R}_{\mathtt{c'c}}\mathtt{J}_i = \mathtt{I}_i. \quad (12)$$

The rotation-invariance is not surprising. Intuitively, since we are considering only part of (5) (without visibility constraint) and modeling the camera as a general bearing sensor, the camera should receive the same information regardless of its rotation. Moreover, the choice of global frame expresses the constant information in a fixed frame, resulting in the invariance (12). If the local perturbation in (10) is chosen, such invariance is not possible, and the information matrix will be related by an adjoint map of SE(3) [25, Ch. 2].

To summarize, by choosing the bearing vector as the observation and parameterizing the pose perturbation in the global frame, the information matrix, without the visibility constratint, is rotation-invariant. Next, we will see how to handle the visibility function $v(\cdot)$ in (5).

B. Visibility Modeling

The exact visibility $v(\cdot)$ is a non-trivial function (e.g., horizontal/vertical/diagonal FoVs are not the same) of the bearing vector $\mathbf{f}$. To arrive at the desired form (6), we first simplify $v(\cdot)$ as a function of the angle θ between the bearing vector $\mathbf{f}$ and the optical axis $\mathbf{e}_3 = [0, 0, 1]^\top$, as illustrated in Fig. 2a. Then, we further replace the simplified (still binary valued) function with a second order polynomial of $\cos\theta$, and for the ith landmark we have:

$$v(\mathtt{T}_{\mathtt{wc}}, \mathbf{p}_i^{\mathtt{w}}) \rightarrow v(\theta_i) = a_2\cos^2\theta_i + a_1\cos\theta_i + a_0, \quad (13)$$

where $\{a_0, a_1, a_2\}$ are design parameters that can be used to model: 1) the decay of the visibility when $\mathbf{f}$ moves away from the center of the FoV; 2) the effect of different FoVs. Several examples are shown in Fig. 2b.

The visibility model (13) is motivated by two reasons. First, the binary valued visibility function is not differentiable, which can bring problems for optimization-based motion planning methods that relies on the gradient of the cost function; second, the cosine function can be conveniently written as $\cos\theta_i = \frac{(\mathbf{p}_i^{\mathtt{c}})^\top\mathbf{e}_3}{n_i} = \frac{(\mathbf{e}_3)^\top\mathbf{p}_i^{\mathtt{c}}}{n_i}$, which allows us to write the overall information (5) in the form of (6).

C. Information Kernel

With the rotation-invariant Fisher information (12) and the visibility approximation (13), the complete FIM for one pose can be written as

$$\begin{aligned}\mathtt{I}_{\mathtt{T}_{\mathtt{wc}}} &\approx \sum_{i=1}^{N} a_2\cos\theta_i^2\mathtt{I}_i + a_1\cos\theta_i\mathtt{I}_i + a_0\mathtt{I}_i \\ &\approx \sum_{i=1}^{N} \frac{a_2}{n_i^2}\operatorname{diag}_6(\mathbf{e}_3^\top\mathbf{p}_i^{\mathtt{c}})\mathtt{I}_i\operatorname{diag}_6((\mathbf{p}_i^{\mathtt{c}})^\top\mathbf{e}_3) \\ &\qquad + \frac{a_1}{n_i}\operatorname{diag}_6(\mathbf{e}_3^\top\mathbf{p}_i^{\mathtt{c}})\mathtt{I}_i + k_0\mathtt{I}_i.\end{aligned} \quad (14)$$

We use $\operatorname{diag}_n(A)$ to denote a (block) diagonal matrix by repeating A by n times on the diagonal and $\operatorname{diag}([A_1, \ldots, A_n])$ for the diagonal blocks of $A_1, A_2, \ldots, A_n$. Observing $\mathbf{p}_i^{\mathtt{c}} = \mathtt{R}_{\mathtt{cw}}(\mathbf{p}_i^{\mathtt{w}} - \mathbf{t}_{\mathtt{wc}})$, $\mathbf{p}_i^0 \triangleq \mathbf{p}_i^{\mathtt{w}} - \mathbf{t}_{\mathtt{wc}}$ and collecting the rotation-independent parts into one matrix, we have

$$\mathtt{I}_{\mathtt{T}_{\mathtt{wc}}} \approx \sum_{i=1}^{N}\mathtt{A}(\mathtt{R}_{\mathtt{wc}})\mathtt{H}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}})\mathtt{B}(\mathtt{R}_{\mathtt{wc}}), \quad (15)$$

where

$$\begin{aligned}\mathtt{A}(\mathtt{R}_{\mathtt{wc}}) &= [a_2\operatorname{diag}_6(\mathbf{e}_3^\top\mathtt{R}_{\mathtt{wc}}^\top),\ a_1\operatorname{diag}_6(\mathbf{e}_3^\top\mathtt{R}_{\mathtt{wc}}^\top),\ a_0\mathcal{I}_6] \\ \mathtt{B}(\mathtt{R}_{\mathtt{wc}}) &= [\operatorname{diag}_6(\mathbf{e}_3^\top\mathtt{R}_{\mathtt{wc}}^\top),\ \mathcal{I}_6,\ \mathcal{I}_6]^\top,\end{aligned} \quad (16)$$

and the rotation independent matrix is $\mathtt{H}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}}) = \operatorname{diag}([\mathtt{H}_2, \mathtt{H}_1, \mathtt{H}_0])$ with

$$\begin{aligned}\mathtt{H}_2 &= \frac{1}{\|\mathbf{p}_i^0\|_2^2}\operatorname{diag}_6(\mathbf{p}_i^0)\mathtt{I}_i\operatorname{diag}_6^\top(\mathbf{p}_i^0), \\ \mathtt{H}_1 &= \frac{1}{\|\mathbf{p}_i^0\|_2}\operatorname{diag}_6(\mathbf{p}_i^0)\mathtt{I}_i, \quad \mathtt{H}_0 = \mathtt{I}_i,\end{aligned} \quad (17)$$

where $\mathtt{H}_2$, $\mathtt{H}_1$ and $\mathtt{H}_0$ are of the dimension 18×18, 18×6 and 6×6 respectively. Finally, it can be seen that (15) satisfies the desired property of (6):

$$\mathtt{I}_{\mathtt{T}_{\mathtt{wc}}} \approx \mathtt{A}(\mathtt{R}_{\mathtt{wc}})(\sum_{i=1}^{N}\mathtt{H}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}}))\mathtt{B}(\mathtt{R}_{\mathtt{wc}}). \quad (18)$$

We call the sum of $\mathtt{H}(\cdot)$ the *information kernel* at $\mathbf{t}_{\mathtt{wc}}$:

$$\mathtt{K}_{\mathtt{I}}(\mathbf{t}_{\mathtt{wc}}) = (\sum_{i=1}^{N} \mathtt{H}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}})), \tag{19}$$

which we will use in the proposed Fisher information field.

IV. Information Field for Active Localization

A. Representation, Query and Update

Using the kernel (19), we propose a volumetric representation, namely the *Fisher information field*, for active localization. In particular, after discretizing the space of interest into voxels, we compute the kernels for the voxels (from all the 3D landmark) and store each kernel in the corresponding voxel. Then, when the information of a certain pose is queried, the related kernels (by nearest neighbor or interpolation) are retrieved, and (18) is used to recover the information in constant time. The method is illustrated in Fig. 1. Once the field is built, the query of the information for an arbitrary pose only requires a linear transformation of the related kernels instead of checking all the points in the point cloud, which is the key advantage of the proposed method.

Field Update: In practice, especially during the exploration of an unknown environment, new landmarks may be added and existing ones deleted over time, and our representation needs to adapt to such changes. Fortunately, since the kernel (19) is in the form of the summation of components calculated from each landmark independently, adding/deleting the contribution of a landmark can be done trivially by adding/subtracting the corresponding components from existing kernels.

B. Memory Usage and Trace Kernel

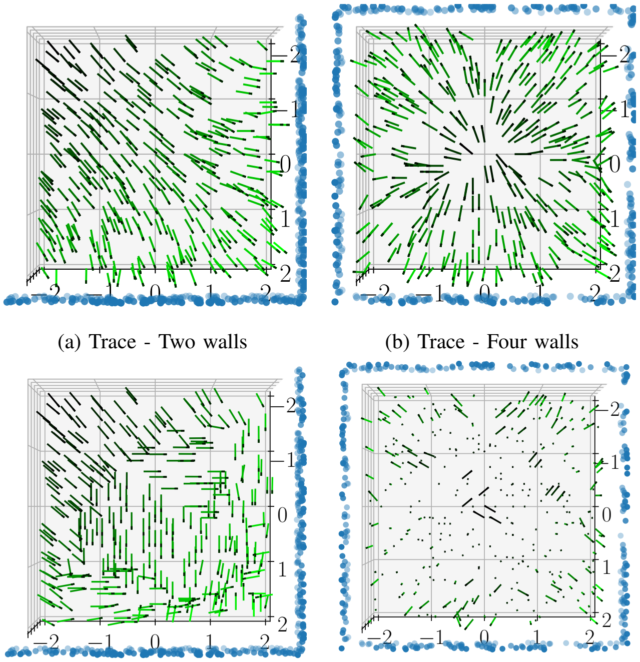
The constant query time comes at the cost of extra memory. The information kernel at each location consists of three parts (17), which in total require 468 float numbers. Admittedly, the size of storage needed is non-negligible, and it increases linearly with the number of voxels in the field. But the memory footprint is still acceptable in practice (and much less than discretizing and sampling the rotation space exhaustively), as we will show in Section V-A.

Note that the aforementioned information representation can be used to recover the full approximated information matrices (6×6). However, in the cost (4), only one scalar metric $s(\cdot)$ is needed in the overall cost for active localization. This brings the possibility of reducing the memory usage by directly expressing one specific metric instead of the full information matrix. Out of different metrics often used with the Fisher information [23, Ch. 6&9], the T-optimality criterion, which is the matrix trace, is especially suitable (i.e., a linear function) for this purpose. In particular, taking the trace of the approximated information (14), we can arrive at the similar form as (15)

$$\mathrm{Tr}(\mathtt{I}_{\mathtt{T}_{\mathtt{wc}}}) \approx \sum_{i=1}^{N} \mathtt{A}_{\mathrm{Tr}}(\mathtt{R}_{\mathtt{wc}})\mathtt{G}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}})\mathtt{B}_{\mathrm{Tr}}(\mathtt{R}_{\mathtt{wc}}), \tag{20}$$

TABLE I: Time and memory cost in a realistic room environment.

Method	$t_{\mathtt{build}}$ (s)	$t_{\mathtt{query}}$ (s)	Memory (MB)
Information Kernels	~ 1.4	~ 0.4	~ 270
Trace Kernels	~ 0.3	~ 0.05	~ 8
Point Cloud	0	~ 2	~ 0.11



(a) Trace - Two walls

(b) Trace - Four walls

(c) Determinant - Two walls

(d) Determinant - Four walls

Fig. 3: Visualization of the information field in simulated scenes for the trace and determinant metric. Blue circles are 3D landmarks. Each line segment stands for one optimal view direction. Brighter color means better localization quality. Fig. 3d is one failure case due to the visibility model.

where $\mathtt{G}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}}) = \mathrm{diag}([\frac{\mathbf{p}_i^{0}\,\mathrm{Tr}(\mathtt{I}_i)(\mathbf{p}_i^{0})^{\top}}{\|\mathbf{p}_i^{0}\|_2^2}, \frac{\mathbf{p}_i^{0}\,\mathrm{Tr}(\mathtt{I}_i)}{\|\mathbf{p}_i^{0}\|_2}, \mathrm{Tr}(\mathtt{I}_i)])$ The derivation of $\mathtt{A}_{\mathrm{Tr}}(\mathtt{R}_{\mathtt{wc}})$ and $\mathtt{B}_{\mathrm{Tr}}(\mathtt{R}_{\mathtt{wc}})$ is trivial and omitted here. The corresponding kernel for the trace is then

$$\mathtt{K}_{\mathrm{Tr}}(\mathbf{t}_{\mathtt{wc}}) = \sum_{i=1}^{N} \mathtt{G}(\mathbf{p}_i^{\mathtt{w}}, \mathbf{t}_{\mathtt{wc}}). \tag{21}$$

(21) can be used in the same way as (19), but only requires 13 float numbers for one voxel (at the cost of losing certain information contained in the full FIMs).

V. Experiments

To implement the proposed method, we used the voxel hashing method [26] with the implementation from [2]. We took 3D landmarks as input and built the proposed representation incrementally or in a batch. When the information field was built incrementally, new voxels were added to the area around the camera as the camera moves in order to simulate navigation scenarios of mobile robots. Unless specified otherwise, we used the trace kernel formulation.

We performed experiments on both simulated and real-world data. In simulation, we simulated an ideal pinhole camera and used groundtruth depth for 3D landmarks. For real-world data, we ran a stereo visual-inertial odometry pipeline that consists of an efficient frontend [27] and an

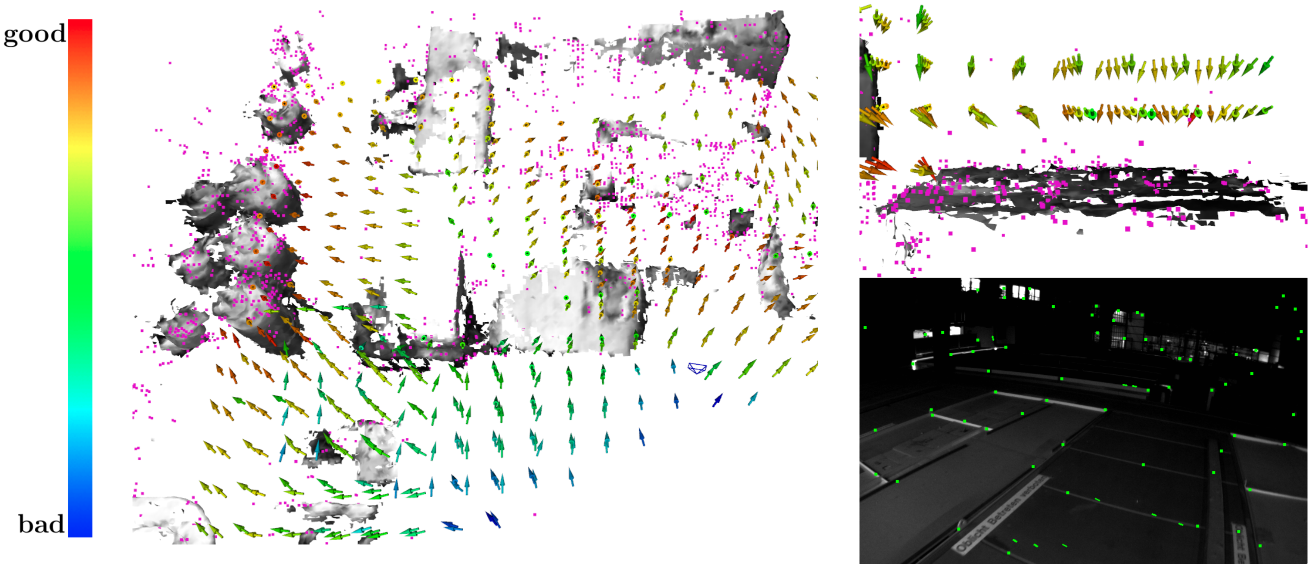


Fig. 4: Visualization of the information layer built from the EuRoC *Machine Hall 05* sequence. **Right**: the overall information field. The magenta points are the landmarks from a visual-inertial odometry. The arrows stand for the optimal orientations for visual localization at the corresponding positions. The color coding indicating the localization quality. **Left**: the zoom-in view on the one part of the scene and the corresponding camera image, where most landmarks are located nearby below the MAV. The optimal orientations calculated by the information field correctly point to these nearby landmarks.

optimization backend [28], and used the estimated landmarks and camera poses to build the information field.

A. Time and Space Complexity

We first compared the time and memory complexity of the proposed method with that using a point cloud. The simulated test scenario was a $5\mathrm{m}\times5\mathrm{m}\times3\mathrm{m}$ room, with $5{,}000$ landmarks. We used a voxel size of $10\mathrm{cm}$ and computed the two types of kernels (19) and (21) for all the voxels in the room, which corresponds to $75{,}000$ voxels. After building the information field, we queried the information or the trace at random poses inside the room for $500{,}000$ times, for each we assumed that 0.5% of the landmarks (namely 25) were visible. Both the time (building and query) and memory usage of our method and that using a point cloud directly are summarized in Table I.

It can be seen that the query time of the information field is much lower, for both kernel types, than the point cloud method. The memory and building time are higher for our methods, but still within reasonable ranges. Note that we computed the kernels for the voxels densely within the test scenario in a batch. In many applications, the field will be expanded as the robot moves, and is not likely to fill the space entirely (e.g., Fig. 4), and the building time will be amortized as well. Also the assumption of 25 visible landmarks is rather conservative, and a higher number of visible landmarks will have a negative impact on the query time of the point cloud method but not ours.

B. Qualitative Evaluation

To visualize the kernel in each, we calculated the optimal orientation by exhaustively searching possible rotations to find the one that maximizes a certain metric. This is also one practical application. Thanks for the efficient query enabled by the proposed representation, we were able to compute the optimal orientation online. The results from simulation and real-world data are shown in Fig. 3 and Fig. 4 respectively. We can observe in both case: 1) the optimal views determined by our representation in general point to the area with many landmarks, 2) the color coding indicates that our method correctly determined that the locations closer to landmarks can be better localized. However, we indeed see some failure cases (e.g., Fig. 3d), and the reason is discussed below.

C. Discussion

Visibility Modeling: (13) allows us to have the desired form of the Fisher information. Unfortunately, the heavy tail for large θ (see Fig. 2b) can be problematic: it can wrongly take into consideration the information of the landmarks that are actually not visible, resulting in undesired behaviors, as in Fig. 3d. Designing and quantitatively comparing different approximations (e.g., sigmoid) that are compatible with (6) is one important direction for future work.

Occlusion: In our implementation, occlusion is not considered. This did not cause severe problems for sparse environments in the experiments, such as Fig. 4. To handle occlusion properly, visibility check using a dense environment model is required, which falls out of the scope of this work.

VI. Conclusion

In this work, we proposed a novel representation, namely the Fisher information field, to efficiently calculate the Fisher information of 6 DoF visual localization process. The proposed method is based on the approximation of the exact Fisher information and can achieve constant computation time, regardless of the number of landmarks in the environment. The advantage and usefulness of our method were shown on both simulated and real-world data. For future work, we plan to integrate the map representation with closed-loop systems to demonstrate the strength of our method in active settings. Quantitative studies on different visibility approximations and FoVs are also of interest.

References

[1] R. Bajcsy, “Active perception,” *Proc. IEEE*, vol. 76, no. 8, pp. 966–1005, 1988. 1

[2] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, “Voxblox: Incremental 3D euclidean signed distance fields for on-board MAV planning,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2017. 1, 5

[3] H. S. S. Feder, J. J. Leonard, and C. M. Smith, “Adaptive Mobile Robot Navigation and Mapping,” *Int. J. Robot. Research*, vol. 18, no. 7, pp. 650–558, 1999. 2

[4] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, “An experiment in integrated exploration,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Sept. 2002. 2

[5] W. Burgard, D. Fox, and S. Thrun, “Active mobile robot localization,” in *Int. Joint Conf. Artificial Intell. (IJCAI)*, Aug. 1997. 2, 3

[6] N. Roy, W. Burgard, D. Fox, and S. Thrun, “Coastal navigation-mobile robot navigation with uncertainty in dynamic environments,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 1999. 2, 3

[7] A. J. Davison and R. M. Murray, “Simultaneous localization and map-building using active vision,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, 2002. 2

[8] M. Achtelik, S. Weiss, M. Chli, and R. Siegwart, “Path planning for motion dependent state estimation on micro aerial vehicles,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2013. 2

[9] S. A. Sadat, K. Chutskoff, D. Jungic, J. Wawerla, and R. Vaughan, “Feature-rich path planning for robust navigation of mavs with mono-SLAM,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014. 2

[10] C. Mostegel, A. Wendel, and H. Bischof, “Active monocular localization: Towards autonomous monocular exploration for multirotor mavs,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014. 2

[11] C. Papachristos, S. Khattak, and K. Alexis, “Uncertainty-aware receding horizon exploration and mapping using aerial robots,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017, pp. 4568–4575. 2, 3

[12] Z. Zhang and D. Scaramuzza, “Perception-aware receding horizon navigation for MAVs,” *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018. 2, 3

[13] M. Watterson, S. Liu, K. Sun, T. Smith, and V. Kumar, “Trajectory optimization on manifolds with applications to SO(3) and R3XS2,” in *Robotics: Science and Systems (RSS)*, June 2018. 2

[14] D. Falanga, P. Foehn, P. Lu, and D. Scaramuzza, “PAMPC: Perception-aware model predictive control for quadrotors,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Sept. 2018. 2

[15] G. Costante, C. Forster, J. A. Delmerico, P. Valigi, and D. Scaramuzza, “Perception-aware path planning,” *arXiv e-prints*, 2016. 2

[16] B. Ichter, B. Landry, E. Schmerling, and M. Pavone, “Robust motion planning via perception-aware multiobjective search on GPUs,” in *Proc. Int. Symp. Robot. Research (ISRR)*, Dec. 2017. 2

[17] V. Indelman, L. Carlone, and F. Dellaert, “Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments,” *Int. J. Robot. Research*, vol. 34, no. 7, pp. 849–882, 2015. 2, 3

[18] Y. B. Elisha and V. Indelman, “Active online visual-inertial navigation and sensor calibration via belief space planning and factor graph based incremental smoothing,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Sept. 2017. 2

[19] B. Efron and D. V. Hinkley, “Assessing the accuracy of the maximum likelihood estimator: Observed versus expected fisher information,” *Biometrika*, vol. 65, no. 3, pp. 457–483, 1978. 2

[20] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003, 2nd Edition. 2

[21] T. D. Barfoot, *State Estimation for Robotics*, 1st ed. New York, NY, USA: Cambridge University Press, 2017. 2

[22] M. J. Box, “Bias in nonlinear estimation,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 33, no. 2, pp. 171–201, 1971. 3

[23] F. Pukelsheim, *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, 2006. 3, 5

[24] L. Carlone and S. Karaman, “Attention and anticipation in fast visual-inertial navigation,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017. 3

[25] H. Strasdat, “Local accuracy and global consistency for efficient SLAM,” Ph.D. dissertation, Imperial College London, UK, 2012. 4

[26] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, “Real-time 3D reconstruction at scale using voxel hashing,” *ACM Trans. Graph.*, vol. 32, no. 6, p. 169, Nov. 2013. 5

[27] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, “SVO: Semidirect visual odometry for monocular and multicamera systems,” *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, 2017. 5

[28] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, “Keyframe-based visual-inertial SLAM using nonlinear optimization,” *Int. J. Robot. Research*, 2015. 6