# Passive Global Localisation of Mobile Robot through Panoramic 2D LIDAR-based Scan-to-map-scan Matching in Static Environments via 2D Fourier-Mellin Invariant Matching

Alexandros Filotheou, Anastasios Tzitzis, Emmanouil Tsardoulias,
Antonis Dimitriou, Andreas Symeonidis, George Sergiadis, Loukas Petrou
Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece
{alefilot, atzitzis}@auth.gr, etsardou@eng.auth.gr, antodimi@auth.gr, symeonid@eng.auth.gr, sergiadi@auth.gr, loukas@eng.auth.gr

***Abstract*—Passive global localisation is defined as locating a robot on a map, under global pose uncertainty, without prescribing motion controls. The majority of current solutions either assume structured environments or require tuning of parameters relevant to establishing correspondences between sensor measurements and segments of the map. This article advocates for a solution that dispenses with both in order to achieve greater portability and universality across disparate environments. A single 2D panoramic LIDAR sensor is used as the measurement device, this way reducing computational and investment costs. The proposed method disperses pose hypotheses on the map of the robot's environment and then captures virtual scans from each of them. Subsequently, each virtual scan is matched against the one derived from the physical sensor. Angular alignment is performed via 2D Fourier-Mellin Invariant Matching (FMI); positional alignment is performed via feedback of the position estimation error. In order to deduce the robot's pose the method sifts through hypotheses by using measures extracted from FMI. Simulations illustrate the efficacy of the proposed global localisation solution in realistic surroundings and scenarios. In addition, the proposed method is pitted against the most effective ICP variant under the same task, and three conclusions are drawn. The first is that the proposed method is effective in both structured and unstructured environments. The second is that it concludes to fewer false positives. The third is that the two methods are largely equivalent in terms of pose error.**

## I. Introduction

Mobile robot localisation on a plane is a well-studied field in robotics and several diverse approaches to its solution have been proposed in the past. The term "robot localisation" conflates two distinct problems arising from practical needs. The first one, termed *Global Localisation*, is the problem of estimating the pose of a robot within a map of its surroundings under global uncertainty of location and orientation. The second, termed *Pose Tracking*, is the problem of estimating a robot's pose when past poses, controls, and measurements are available. This article focuses on the first aspect of localisation.

When motion commands are dictated from the localisation system itself, global localisation is termed "active" [1]-[9]. When they are dictated by some external system, global localisation is termed "passive" [10]-[35]. However, in the stringent context of autonomy, motion commands may be denied altogether, since generating motion commands without knowledge of the robot's whereabouts may be considered a mission safety risk. This article focuses on this last case of systems.

Global localisation of a robot is conditioned on the knowledge of the map of its environment and measurements of the environment derived from on-board sensor(s). The correct association between the two by a method of global localisation determines the success of localisation. The association depends on the nature of the sensor(s) used to generate the map and the nature of the on-board sensor(s) used in solving the problem of (global) localisation. LIDAR sensors capture distances to objects within their range, producing 2D or 3D point clouds, and therefore they are used in generating 2D or 3D maps [36]-[39]. Such sensors have been widely adopted due to their low measurement error, real-time operability, and almost no need for preprocessing [40]. In the context of robot localisation LIDAR sensors have become the prominent candidate for deriving information from, since they are the same sensors used for mapping an environment. Other sensors, such as RGB(D) cameras and sonar sensors, have also been used to solve the problem of global localisation [3][24][26]. This article focuses on its solution with the use of a single 2D LIDAR sensor due to its superior measurement accuracy compared to them, its significantly lower cost compared to a 3D LIDAR sensor, and the need for fewer computational resources compared to employing sensors producing 3D point clouds.

Approaches to the solution of the problem of global localisation establish associations between measurements from one or more sensors and the map of the robot's surroundings, either in the native measurement domain [18][20][33], or in an extracted abstraction space. In the latter case, features of the measurement vectors and the map are located, extracted (coded), and associated—features such as lines, planes, corners, doors, or others [12][13][16][30]. Knowledge of the

environment as well as of its structure are presupposed, and therefore these approaches cannot be employed in unknown or unstructured environments. On the other hand, in structured environments the sought-for features are not guaranteed to exist in both surroundings of the robot and the map, or they do exist but not in a sufficiently undisturbed state. In any case, current approaches usually require manual tuning of parameters governing their success, regardless of their reliance on features; therefore they are conditioned in an ad hoc manner on circumstance and intuition. Furthermore, different environments exhibit different structures and geometry, and, if robot portability is sought for, their adoption is hindered due to the effort needed to tailor each approach to each environment. Consequently it would be valuable to strive to develop methods that do not depend on the existence of local features or the tuning of internal parameters, and that can operate in a variety of environments, without the need or overhead of adaptation.

The contribution of the proposed method is the solution of the passive global localisation problem in a 2D setting with a panoramic 2D LIDAR sensor (a) without knowledge of the structure of the robot's environment, and (b) without the need of tailoring parameters to the particular characteristics of the LIDAR sensor used to acquire knowledge of the environment. The proposed solution operates directly in measurement space (thus without information loss), does not require the existence of features in either map or measurements, and requires no parameters to be tuned—apart from an optionally set parameter, which may be configured in order to help resolve pose ambiguities in cases of repeating environment structures. The proposed method requires only one information input, whose value is, in principle, proportional to the area of the map of the robot's surroundings. Compared to state-of-the-art methods in scan-matching, the proposed method exhibits fewer false positives, especially in the case of missing range measurements (which are due to sensor fault or shortness of the sensor's maximum range compared to the distances to its surrounding objects).

The proposed method disperses hypotheses over the unoccupied space of the map. It tries to match the input range scan to the range scans captured from the hypotheses' poses. The latter are constructed by raycasting the map from each hypothesis' location and orientation. The rotation and translation correction between the pose of a hypothesis and the unknown pose of the robot is therefore performed via scan-to-map-scan matching, and the distinction between true and false positives is performed via the similarity degree extracted from the orientation correction method. The latter rests on matching the projections of the two scans onto the 2D plane via Fourier-Mellin Invariant Matching. The translation of a pose hypothesis is performed by computing the centroids of the two ray-endpoint-sets and moving the hypothesis's position by their difference.

The rest of this article is structured as follows: section II formulates the problem, provides useful notions and clarifications, necessary definitions, and the systemic rationale of tools used subsequently. Section III provides an overview of solutions to the problem of global localisation heretofore presented in the literature, and how the underlying method of rotation correction used by the proposed method has been previously leveraged to serve robotics-related purposes. Section IV states the objective of this article and illustrates the motivation behind the introduced method. Section V describes its internal structure and processes, and illustrates the latter along with their effects in a model scenario. Section VI evaluates the introduced method in various simulational scenarios and (structured and unstructured) environments; simultaneously it is pitted against the performance of the same method but whose core method is substituted for a state-of-the-art scan-matching algorithm, now repurposed in the scan-to-map-scan context. The latter relies on establishing correspondences and parameter-tuning. Finally, section VII sums up and analyses the results of the previous section, while section VIII offers a recapitulation and concludes this paper.

## II. Preliminaries

### A. Definitions

**Problem *P*.** *Global localisation in mobile robotics.* Global localisation is the problem of recovering a robot's pose within the global frame of reference of a map. The map of the environment that the mobile robot operates in is given. No initial assumption about the robot's pose within the global frame of reference of the map exists. Measurements from at least one sensor source are provided.

**Definition I.** *Passive and Active global localisation.* A way to classify the existing solutions to the global localisation problem is through the robot's own navigation system. In order to maximise the probability of successful localisation, *Active* global localisation approaches generate motion commands that move the robot into the environment in poses which enable the localisation system to acquire additional information. If motion commands—if any—are determined without advice from the global localisation system, global localisation is termed *Passive*.

**Remark I.** *Accuracy of solution.* An accurate solution to the global localisation problem is a necessary condition for the solution of the subsequent pose-tracking problem, i.e. tracking the pose of the robot with respect to the global reference frame as the robot moves within its environment. In recent years probabilistic approaches to the problem of pose tracking [41]-[43] have become the standard. However, thanks to their tolerance for increased pose uncertainty, the requirement of maximum accuracy with regard to global localisation has rather diminished. If probabilistic methods are used in the subsequent task of pose-tracking then a larger set of poses can be admitted as solution to the problem of global localisation. A solution residing in the immediate vicinity of the true pose of a robot, with direct line-of-sight to it, will be referred to as a correct solution.

**Remark II.** *Real-time solution.* The requirement to calculate a solution in real time is indispensable in the context of the

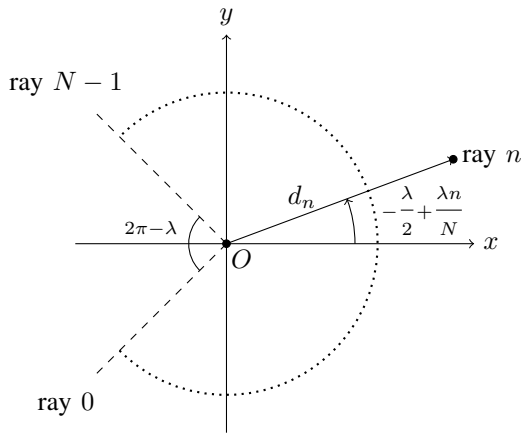



Fig. 1: The (local) frame of reference of a typical conventional range sensor. The sensor is located at $O(0,0)$ and its heading is that of the $x$ axis

pose-tracking problem, as knowledge of the robot's pose is a condition for the successful operation of subsequent tasks, e.g. navigation [44][45][46], or, for instance, localising RFID tags within the the robot's environment [47][48]. In the global localisation problem, on the other hand, there is no strict requirement for real-time solution in the case of passive global localisation if the robot does not move at all.

**Definition II.** *Definition of a range scan captured from a conventional 2D LIDAR sensor.* A conventional 2D LIDAR sensor a finite number of ranges, i.e. distances to objects within its range, on a horizontal cross-section of its environment, at regular angular and temporal intervals, over a defined angular range [40]. We define a range scan $\mathcal{S}$, consisting of $N$ rays over an angular range $\lambda$, to be an ordered sequence of $N$ pairs of (a) one range measurement and (b) one angle: i.e. the ray's angle relative to the sensor's heading, expressed in the sensor's frame of reference, ordered by increasing angle:

$$\mathcal{S} \equiv \{(d_n, -\frac{\lambda}{2} + \frac{\lambda n}{N})\}, n = \{0, 1, \ldots, N-1\} \qquad (1)$$

**Remark III.** The angular range of a conventional LIDAR sensor is symmetrically distributed on either side of its $x$-axis, and each ray is equiangularly spaced from its neighbouring rays (with the exception of the first and last rays if $\lambda < 2\pi$). If $\lambda = 2\pi$ then the sensor is termed *panoramic*.

**Definition III.** *Scan-to-scan matching using a 2D LIDAR sensor* (adapted for use in two dimensions from [49].) Let two range scans as defined by Definition II, $\mathcal{S}_\mathrm{r}$ and $\mathcal{S}_\mathrm{v}$, be captured from a LIDAR sensor operating in the same environment at both capturing times. Let $\boldsymbol{p}_\mathrm{v}(x_\mathrm{v}, y_\mathrm{v}, \theta_\mathrm{v})$ be the pose from which the sensor captured $\mathcal{S}_\mathrm{v}$, expressed in some coordinate system (usually a past pose estimate of the sensor). The objective of scan-to-scan matching in two dimensions is to find the roto-translation $\boldsymbol{q} = (\boldsymbol{t}, \theta)$, $\boldsymbol{t} = (\Delta x, \Delta y)$ that minimises the distance of the endpoints of $\mathcal{S}_\mathrm{v}$ roto-translated by $\boldsymbol{q}$ to their projection on $\mathcal{S}_\mathrm{r}$. Denoting the endpoints of $\mathcal{S}_\mathrm{v}$ by $\{\boldsymbol{p}_\mathrm{v}^i\}$, in formula:

$$\min_{\boldsymbol{q}} \sum_i \left\| \boldsymbol{p}_\mathrm{v}^i \oplus \boldsymbol{q} - \prod\{\mathcal{S}_\mathrm{r}, \boldsymbol{p}_\mathrm{v}^i \oplus \boldsymbol{q}\} \right\|^2 \qquad (2)$$

The symbol "$\oplus$" denotes the roto-translation operator $\boldsymbol{p}_\mathrm{v}^i \oplus (\boldsymbol{t}, \theta) \triangleq \boldsymbol{R}(\theta)\boldsymbol{p}_\mathrm{v}^i + \boldsymbol{t}$, where $\boldsymbol{R}(\theta)$ is the 2D rotation matrix for argument angle $\theta$, and $\prod\{\mathcal{S}_\mathrm{r}, \boldsymbol{p}_\mathrm{v}^i \oplus \boldsymbol{q}\}$ denotes the Euclidean projector on $\mathcal{S}_\mathrm{r}$.

**Remark IV.** The solution to (2) cannot, in principle, be found in closed form due to the arbitrary nature of $\mathcal{S}_\mathrm{r}$ and the non-linearity of the "$\oplus$" operator.

**Remark V.** Scan-to-scan matching is employed in robotics as a means of odometry in non-wheeled robots, where no encoders can be utilised, or as a useful ameliorator of the ever-drifting encoder-ed odometry: scans captured at consecutive time instances, inputted to a scan-matching algorithm, convey an estimate as to the pose of the scan sensor at the second capture time relative to that captured first. Scan-to-scan matching is being successfully employed in the tasks of Simultaneous Localisation and Mapping [50]-[52], local map construction [53]-[55], or in people-tracking systems [56].

**Definition IV.** *Definition of a map-scan.* A map-scan is a virtual scan that encapsulates the same pieces of information that a scan derived from a physical sensor does; only their underlying operating principle is different due to the fact the map-scan refers to distances to obstacles within the map of the robot's environment (hence its virtuality) rather than within the environment itself. A map-scan is captured from a virtual sensor and derived by means of locating intersections of rays emanating from the estimate of the sensor's pose and boundaries demarcating obstacles in the map.

**Definition V.** *Scan–to–map-scan matching in two dimensions.* Scan–to–map-scan matching is defined in the same way as scan-to-scan matching (definition III) but with $\mathcal{S}_\mathrm{v}$ now derived not from the physical environment of the robot but from its map. This subtle difference makes $\boldsymbol{p}_\mathrm{v}$, the pose from which the map-scan $\mathcal{S}_\mathrm{v}$ was captured, continually expressible in the map's frame of reference, and therefore in absolute terms (rather than relative to its previous estimate, which is recursively relative to a convention of the robot's starting pose).

**Remark VI.** The benefit of matching a scan derived from a physical sensor from its actual pose and a map-scan derived from a virtual sensor from its estimated pose comes now into light: Assume that a pose hypothesis exists in the general vicinity of the true pose of a mobile robot equipped with a 2D range-scan sensor; assuming that the range sensor is fixed at the same pose relative to the robot in both real and virtual environments, the roto-translation of the virtual scan's endpoints that minimises their distance to their projection on the physical scan equals the roto-translation that, when applied to the robot's estimated pose, will minimise its distance to its real pose. Therefore extracting the relative roto-translation of

the virtual scan with respect to the real scan can be used as a correction of the estimate of the robot's pose within the map. The significance of this correction lies both in the fact that it may facilitate global localisation and that it may improve a robot's pose estimate during pose-tracking.

### *B. Fourier-Mellin Invariant Matching in 2D*

In this section we recount how the Fourier-Mellin transform can be used to match two 2D grids, hereafter also referred to as images, related to one-another only by affine transformations, namely translation, rotation, and/or scaling [57]-[59].

Consider two fixed-size images, $\boldsymbol{r}(x, y)$, $\boldsymbol{s}(x, y)$, where the latter is a translated, rotated, and scaled copy of the former:

$$\boldsymbol{s}(x, y) = \boldsymbol{r}\big(\sigma(x\cos\xi + y\sin\xi) - x_0, \\ \sigma(-x\sin\xi + y\cos\xi) - y_0\big) \tag{3}$$

Here $x_0, y_0$ are the translational offsets, $\xi$ the rotation angle, and $\sigma$ the scale factor. Let the Fourier transform of a function $h$ be denoted by $\mathcal{H} = \mathcal{F}\{h\}$, the inverse Fourier transform of $\mathcal{H}$ be denoted $h = \mathcal{F}^{-1}\{\mathcal{H}\}$, and $|\mathcal{H}|$ denote the magnitude of $\mathcal{H}$; then the Fourier transforms of $\boldsymbol{s}$, $\boldsymbol{r}$ are related by

$$\mathcal{S}(u, v) = e^{-j\phi_{\boldsymbol{s}}(u,v)}\sigma^{-2}\big|\mathcal{R}\big(\sigma^{-1}(u\cos\xi + v\sin\xi), \\ \sigma^{-1}(-u\sin\xi + v\cos\xi)\big)\big| \tag{4}$$

where $j^2 = -1$, and $\phi_{\boldsymbol{s}}$ is the spectral phase of $\boldsymbol{s}$. $\phi_{\boldsymbol{s}}$ depends on the translation, rotation, and scale of $\boldsymbol{s}$ with regard to $\boldsymbol{r}$, but the spectral magnitude $|\mathcal{S}(u, v)|$

$$\big|\mathcal{S}(u, v)\big| = \big|e^{-j\phi_{\boldsymbol{s}}(u,v)}\sigma^{-2}\mathcal{R}\big(\sigma^{-1}(u\cos\xi + v\sin\xi), \\ \sigma^{-1}(-u\sin\xi + v\cos\xi)\big)\big|$$

$$= \big|\mathcal{R}\big(\sigma^{-1}(u\cos\xi + v\sin\xi), \sigma^{-1}(-u\sin\xi + v\cos\xi)\big)\big| \tag{5}$$

is invariant to translation. Equation 5 shows that a rotation of $\boldsymbol{r}$ rotates the spectral magnitude of $\boldsymbol{s}$ by the same angle, and that a scaling of $\boldsymbol{r}$ scales the spectral magnitude of $\boldsymbol{s}$ by the inverse of the scale factor. Now that translation has been decoupled by the other two linear operations, rotation and scaling can be further decoupled by defining the spectral magnitudes of $\boldsymbol{r}$ and $\boldsymbol{s}$ in polar coordinates:

$$\boldsymbol{r}_p(\theta, \rho) \triangleq |\mathcal{R}(\rho\cos\theta, \rho\sin\theta)|$$
$$\boldsymbol{s}_p(\theta, \rho) \triangleq |\mathcal{S}(\rho\cos\theta, \rho\sin\theta)|$$

Using

$$\sigma^{-1}(u\cos\xi + v\sin\xi) = \frac{\rho}{\sigma}\cos(\theta - \xi)$$
$$\sigma^{-1}(-u\sin\xi + v\cos\xi) = \frac{\rho}{\sigma}\sin(\theta - \xi)$$

one obtains

$$\boldsymbol{s}_p(\theta, \rho) = \sigma^{-2}\boldsymbol{r}_p(\theta - \xi, \rho/\sigma)$$

With the cast to polar coordinates, the rotation of $\boldsymbol{s}$ with respect to $\boldsymbol{r}$ has been transformed into a translation of $\boldsymbol{s}_p$ with respect to $\boldsymbol{r}_p$ along the angular axis. However, scaling in the original domain has not been transformed with this cast. Using a logarithmic scale for the radial axis, scaling can also be reduced to a translation. Let $\lambda = \log\rho$ and $\kappa = \log\sigma$. Then by defining

$$\boldsymbol{r}_{pl}(\theta, \lambda) \triangleq \boldsymbol{r}_p(\theta, \rho) \tag{6}$$
$$\boldsymbol{s}_{pl}(\theta, \lambda) \triangleq \boldsymbol{s}_p(\theta, \rho) = \sigma^{-2}\boldsymbol{r}_{pl}(\theta - \xi, \lambda - \kappa) \tag{7}$$

rotation and scaling are reduced to translations in the angular and radial domain respectively. By Fourier-transforming equations 6 and 7:

$$\mathcal{S}_{pl}(\alpha, \beta) = \sigma^{-2}e^{-2\pi j(\alpha\kappa + \beta\xi)}\mathcal{R}_{pl}(\alpha, \beta)$$

Here the rotation angle $\xi$ and scale factor $\sigma = e^{\kappa}$ appear as phase shifts. The above-described technique decouples translation, rotation, and scaling from one-another and is therefore effective, numerically tractable, and efficient.

Symmetric phase-only matched filtering [58] has been found to be most effective and robust in matching two *similar* (not necessarily identical) images, due to its ability to yield high SNRs and sharp correlation peaks. The core process of acquiring the rotation angle and scale of image $\boldsymbol{s}$ with regard to $\boldsymbol{r}$ is summarised in algorithm I.

**Algorithm I** core `FMI-SPOMF`

**Input:** 2D grids / Images $\boldsymbol{r}$, $\boldsymbol{s}$

**Output:** $Q_0(\alpha, \beta)$

1: compute $\mathcal{R}_{pl}(\alpha, \beta) = \mathcal{F}\{\boldsymbol{r}_{pl}(\theta, \lambda)\}$
2: extract the phase $e^{-j\phi_{\boldsymbol{r}}(\alpha,\beta)}$
3: compute $\mathcal{S}_{pl}(\alpha, \beta) = \mathcal{F}\{\boldsymbol{s}_{pl}(\theta, \lambda)\}$
4: extract the phase $e^{-j\phi_{\boldsymbol{s}}(\alpha,\beta)}$
5: determine $Q_0(\alpha, \beta) = e^{-j(\phi_{\boldsymbol{s}}(\alpha,\beta) - \phi_{\boldsymbol{r}}(\alpha,\beta))}$
6: **return** $Q_0(\alpha, \beta)$

After computing $Q_0$, the rotation angle $\xi$ and scaling factor $\sigma$ can be extracted by calculating the inverse Fourier transform of $Q_0$, $q_0 = \mathcal{F}^{-1}(Q_0)$, and detecting the arguments that maximise it. In this sense, $\max q_0$ can be interpreted as a similarity measure of the two input images.

In image registration problems the two input images are assumed to be identical and the aim is to determine all four scalar parameters of the geometric transformation relating one to the other. The process followed to extract them is summarised in algorithm II.

In pattern recognition problems, on the other hand, where the aim is to identify among a set of reference images the one that best matches an observed image, one can potentially assert the existence of this image by executing the core FMI-SPOMF algorithm (algorithm I) for all reference images, identify the maximum of each output $Q_0$, and place a threshold on it; if the maximum exceeds it, it is a match. False positives can be then filtered by executing algorithm II, comparing the observed image with each input reference image after translation, rotation, and scaling by its identified parameters, and applying a threshold to the maximum of $Q_0$.

**Algorithm II** FMI-SPOMF for image registration

**Input:** 2D grids / Images $\boldsymbol{r}$, $\boldsymbol{s}$

**Output:** Translation $(x_0, y_0)$, rotation angle $\xi$, scale $\sigma$, similarity measure $w$

1: $Q_0(\alpha, \beta) \leftarrow$ execute algorithm I for input ($\boldsymbol{r}$, $\boldsymbol{s}$)
2: Compute $q_0(\theta, \lambda) = \mathcal{F}^{-1}\{Q_0(\alpha, \beta)\}$
3: Determine $(\xi, \kappa) \leftarrow \arg\max_{\theta,\lambda} q_0(\theta, \lambda)$
4: $\boldsymbol{s} \leftarrow$ Rescale $\boldsymbol{s}$ by $\sigma^{-1} = e^{-\kappa}$
5: $\boldsymbol{s}' \leftarrow \boldsymbol{s}$
6: $\boldsymbol{s} \leftarrow$ rotate $\boldsymbol{s}$ by $\xi$
7: $\boldsymbol{s}' \leftarrow$ rotate $\boldsymbol{s}'$ by $\xi + \pi$
8: $Q_1(\alpha, \beta) \leftarrow$ execute algorithm I for input ($\boldsymbol{r}$, $\boldsymbol{s}$)
9: $Q_2(\alpha, \beta) \leftarrow$ execute algorithm I for input ($\boldsymbol{r}$, $\boldsymbol{s}'$)
10: $q_1(\theta, \lambda) \leftarrow \mathcal{F}^{-1}\{Q_1(\alpha, \beta)\}$
11: $q_2(\theta, \lambda) \leftarrow \mathcal{F}^{-1}\{Q_2(\alpha, \beta)\}$
12: Determine $q(\theta, \lambda) \leftarrow \arg\max_{q_1,q_2}\{q_1, q_2\}$
13: Determine similarity measure $w = \max q$
14: $(x_0, y_0) \leftarrow \arg\max_{\theta,\lambda} q$
15: **return** $(x_0, y_0), \xi, \sigma, w$

### C. Centroid of a polygon

Green's theorem [60] states that for a piecewise smooth curve $C$ forming the boundary of a region $D$:

$$\oint_C P(x,y)\,dx + Q(x,y)\,dy = \iint_D \frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y}\,dA \tag{8}$$

where $A = \iint_D dA$ is the area of $D$. Clearly the area of $D$ can be calculated by choosing appropriate $P$, $Q$ such that $\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} = 1$. Let $P = 0$ and $Q = x$, then $\oint_C x\,dy = \iint_D dA = A$. Now consider the polygon of figure 2, defined by the ordered sequence of $n$ vertices $\{C_0, C_1, \ldots, C_{n-1}\}$, whose region is demarcated by the border $C = (C_0C_1) \cup (C_1C_2) \cup \cdots \cup (C_{n-1}C_0)$. Because line integrals over piecewise smooth curves are additive in length:

$$A = \oint_C x\,dy = \int_{C_0C_1} x\,dy + \cdots + \int_{C_{n-1}C_0} x\,dy \tag{9}$$

To compute the $k$-th line integral $C_kC_{k+1}$, let the segment from $(x_k, y_k)$ to $(x_{k+1}, y_{k+1})$ be parameterised by

$$C_kC_{k+1} : \vec{r} = \big((x_{k+1} - x_k)t + x_k, (y_{k+1} - y_k)t + y_k\big) \tag{10}$$

where $t \in \mathbb{R} : 0 \le t \le 1$, $k \in \mathbb{Z} : 0 \le k \le n-1$, and $k+1$ is substituted by 0 if $k = n-1$. A single integral in the right-hand side of equation (9) over $C_kC_{k+1}$ is expressed as

$$\int_{C_kC_{k+1}} x\,dy = \int_0^1 \big((x_{k+1} - x_k)t + x_k\big)(y_{k+1} - y_k)\,dt \tag{11}$$

By plugging this parameterisation into equation (9) and summing over all lines one arrives at the following:

$$A = \frac{1}{2}\sum_{k=0}^{n-1}(x_{k+1} + x_k)(y_{k+1} - y_k) \tag{12}$$

where $(x_n, y_n) \equiv (x_0, y_0)$.

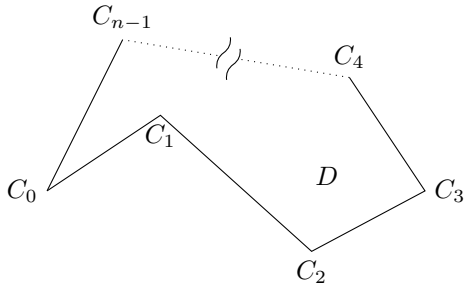



Fig. 2: A polygon of $n$ vertices whose boundary $C = C_0C_1 \cup C_1C_2 \cup \cdots \cup C_{n-1}C_0$ demarcates region $D$

Calculating the polygon's area allows one to obtain its center of mass, i.e. its centroid. The centroid $(x_c, y_c)$ of a polygon is the average position of the area $A$:

$$(x_c, y_c) \equiv \Big(\frac{1}{A}\iint_A x\,dx\,dy, \frac{1}{A}\iint_A y\,dx\,dy\Big) \tag{13}$$

In order to compute $x_c$, let $P = 0$ and $Q = \frac{1}{2}x^2$; then from equation (8):

$$x_c = \frac{1}{A}\iint_A x\,dx\,dy = \frac{1}{A}\oint_C \frac{1}{2}x^2\,dy \tag{14}$$

Using the same parameterisation for each boundary $C_kC_{k+1}$ as in equation (11):

$$\begin{aligned} x_c &= \frac{1}{2A}\int_0^1 \big(x_k^2 + 2x_k(x_{k+1} - x_k)t \\ &\quad + (x_{k+1} - x_k)^2t^2\big)(y_{k+1} - y_k)\,dt \\ &= \frac{1}{6}(x_{k+1}^2 + x_{k+1}x_k + x_k^2)(y_{k+1} - y_k) \end{aligned}$$

Therefore, for a polygon with a finite number of vertices:

$$\begin{aligned} x_c &= \frac{1}{6A}\sum_{k=0}^{n-1}(x_{k+1}^2 + x_{k+1}x_k + x_k^2)(y_{k+1} - y_k) \\ &= \frac{1}{6A}\sum_{k=0}^{n-1}(-y_kx_{k+1}^2 + y_{k+1}x_{k+1}x_k - y_kx_{k+1}x_k + y_{k+1}x_k^2) \\ &= \frac{1}{6A}\sum_{k=0}^{n-1}(x_{k+1} + x_k)(x_ky_{k+1} - x_{k+1}y_k) \end{aligned} \tag{15}$$

In the same fashion:

$$y_c = \frac{1}{6A}\sum_{k=0}^{n-1}(y_{k+1} + y_k)(x_ky_{k+1} - x_{k+1}y_k) \tag{16}$$

where in both equations (15), (16) the coordinate of the $n$-th point are those of the 0-th, and $A$ is given by equation (12).

**Remark VII.** The centroid of a polygon is unique. Its position relative to the points constituting the polygon is independent of the frame of reference in which all points (including the centroid) are expressed.

## III. Literature review

### A. Global Localisation

The problem of global localisation, defined in section II-A, as a distinct component of the problem of pose estimation in mobile robotics, has received extensive attention over the years. The approaches adopted to tackle it, aside from the categorisation into passive and active (ibid), can be also broken down into two more categories: approaches that operate in feature space, and approaches that directly exploit only raw measurements. Furthermore, approaches may be probabilistic filter-based in nature, or deterministic. Some approaches take the minimalistic route, i.e. they rely solely on measurements extracted from one sensor, while others fuse information from various sources to achieve an enhanced result. In this section we give a review of the landscape of solutions to the global localisation problem in mobile robotics.

Active global localisation approaches require the assumption that the robot is free to move in its surroundings even before its pose has been conclusively estimated, thus leaving the robot potentially susceptible to collisions with its environment. The robot's movement in pose space is, in principle, beneficial to the global localisation solution since movement in pose space increases sensor measurements and therefore the probability of sampling diverse and unique environment portions, which subsequently increase the probability of successful pose estimation and resolution of pose ambiguity. In [1] a hybrid approach to global localisation based on Bayesian theory [41] and multiple-hypothesis tracking using Kalman filtering [43] of continuous Gaussian pose hypotheses is presented. The latter and the world model, a topological map of the world, are used to generate movement commands so as to gain more information from the environment and compensate for odometry drift in order to sift through the standing pose hypotheses and resolve ambiguities. Motion commands are generated heuristically by moving on the edges of the topological map (so that the robot is unlikely to collide with obstacles), avoiding to visit the same place twice (since it is unlikely that new information can be gained from the same pose), and selecting to visit the neighbouring node with the maximum number of features in its vicinity. Features in this case are extracted from a range-scan sensor, and comprise door features, line features, and point-pair features, and they are used to either generate new hypotheses or support already existing ones. Each detected feature creates a set of possible poses of the robot, which are treated as measurements of the robot's true pose. The internal operations of each Kalman filter attached to each pose candidate then ensure that the probability of valid hypotheses increases, while incorrect ones perish via additional external thresholding.

Another active global localisation approach [2] commences from an even more minimalist disposition with regard to the number and type of sensors used, by employing only odometry and bumper sensors, and reaches the astonishing (theoretical) conclusion that, given an exact map, global localisation can be achieved through the use of only these two types of sensors, albeit the kinematic model of the robot is constrained to be error-free and that the robot is allowed to collide with its surroundings. The underlying method casts the problem in the robot's information space and solves a discrete-time planning problem, showing that, with relatively common world/map constraints, global localisation is indeed possible, but with a degree of ambiguity proportional to the degree of symmetries present in the environment.

In [3] a particle filter [42] is used as the probabilistic filter of choice due to its inherent ability to represent arbitrary non-unimodal probability distributions. In contrast to [1], a sonar range-finder is employed as the exteroceptive sensor, and no features are detected. At the initialisation phase, particles are uniformly dispersed over the pose space and a weight is calculated for each, according to the inverse least-squared-distance error between the actual sensor measurement and the output of the sensor's observation model for that particular pose hypothesis. The algorithm runs iteratively, without the robot moving, thus realising an optimisation algorithm. In order to avoid particle depletion at this stage, new samples are generated using an evolutionary perspective, where the surviving particles—those without negligible weights—serve as the locus around which new particles are introduced in order to reinforce the presence of particles where probability of finding the robot is highest. In order to reduce the probability of the filter getting trapped into local minima, new particles that guarantee minimal space coverage are introduced randomly in pose space. The algorithm stops when stable solution(s) are identified. These are then used as initial poses from which the closest obstacle to a hypothesis is identified through the sonar sensor, and whose vicinity the robot is instructed to move towards. During navigation each candidate hypothesis is monitored by an Extended Kalman filter and a $\chi$ square test using the Mahalanobis distance is used to validate candidates during data association between observed and theoretically expected measurements.

Research in active global localisation methods seems to have diminished in recent years, with older approaches, including [4]-[9], giving rise to research towards the more demanding problem of passive global localisation, where the robot is denied additional sensor inputs except for those captured from its unknown resting pose.

In [10] a trinocular camera is used to build a 3D map of SIFT landmarks present in the robot's environment, updated over time, while adaptive to dynamic environments, by creating, maintaining, and updating a Kalman filter for each separate landmark detected in the camera's field of view per input frame. This 3D landmark map is built before global

localisation request time, at which, thanks to the specificity of the SIFT features, the robot does not need to move within its environment in order to localise itself. For solving the task of global localisation, SIFT features extracted through the camera at the robot's true pose are matched against those already stored in the 3D map by way of using Hough Transform hashing [11] in order to acquire a rough estimate of the robot's pose (the HT bins with the most votes correspond to pose configurations more likely to achieve a larger number of matches). Then, iterative least-squares minimisation in order to obtain progressively better pose estimates is carried out. The final pose selected is that which has the maximum number of matches and the lowest least-square error.

A method for solving the global localisation problem with regard to robots using laser range finders is demonstrated in [12]. As a preprocessing step, features are extracted from the map of the robot's environment and a database of visible features and attributes from each of the map's grid cells is built. These features are natural landmarks found in both the map and the robot's range scans, and they comprise walls, extracted as straight lines through the use of the Hough transform, concave corners, and discontinuities between consecutive range-scan rays. Landmarks from both sources are then matched together in a two-step process: an initial filter dispenses with the bulk of incorrect pose hypotheses by counting the number of scan-derived landmarks that match the distance, orientation and type of landmark stored in the map's database. Then, a modified discrete relaxation algorithm is employed, using the information of attributes associated with each stored landmark. The grid cell corresponding to the position of the robot is determined among the remaining hypotheses by using a least-square criterion on the distance difference between (a) that cell and each landmark, and (b) the perceived distance to a landmark, derived from the range scan, for all candidate cells. The robot's pose is then calculated as the mean angular deviation between (a) the position of the robot and a perceived landmark and (b) the estimated position of the robot with each landmark visible from that cell.

A general methodology for analysing keypoint design for place recognition is presented in [13], which, in the context of robot navigation and mapping, can be utilised for global localisation. This methodology is particularly informative to the process of tuning the parameters for choice of particular keypoint types, as not all of them are reliable in the presence of noise or occlusions. The authors frame the problem of place recognition as a nearest neighbour search by first selecting a set of keypoints extracted from the field of view of a 2D LIDAR which encode the local region around the robot, and then searching in a database of keypoints previously generated from the map in order to identify places with common features. The map around the robot and the portions of the environment's map identified of having common features are then fed to a pipeline of four verification methods (one of which is the traditional scan-matching ICP method [14]), that sift through false positives, until the best match is found.

Although [15] deals with global localisation in outdoor environments, the operating principle of its method is analogous to those used in indoor ones. The author argues that in the full 3D scenario it is inefficient to match the 3D point cloud from the robot's LIDAR sensor to the 3D point cloud map of the robot's environment, and that, therefore, abstractions in the form of landmarks extracted from the former are necessary to be made in this case. The landmarks used in matching are poles, such as traffic signs, traffic lights, and trees. The author's findings indicate that this type of landmark is not reliable neither for global localisation nor pose tracking, since approximately $40\%$ of all received scans do not feature poles in them, either due to actual absence or generation of false positives by the introduced pole detection mechanism.

A method for global localisation in indoor environments is presented in [16]. The authors argue that in this type of environment, the major structures are walls, doors, or cupboards, meaning structures that can be approximated by straight lines in 2D space. Using this assumption, they build a method that first estimates the orientation of the robot in the map's frame of reference by extracting line features from the robot's map and from the robot's onboard 2D LIDAR sensor, and then matching the latter against the former by using Hough Scan Matching [17]. This estimation is conditioned on the absence of symmetries in the environment. Given the robot's heading, its position can be estimated via the beam endpoint model [41], where the likelihood that a cell of the grid map made the scan measurement is computed for all cells in the map; after this exhaustive search, the cell with the highest likelihood is then selected as the position estimate of the robot. Subsequently, a gradient descent algorithm is employed in order to further eliminate the quantisation error induced by the grid representation of the map.

Except for the map, the method introduced in [18] uses pre-stored reference laser scans, extracted through a 2D LIDAR sensor, and the poses from which these were captured prior to global localisation execution. Upon commencement of global localisation, a local grid map based on the first scan measurements from the real sensor is constructed, and then a number of copies are generated from it by rotating it by integer multiples of 4 degrees until a complete circle is formed. Then a local grid map is created from the pre-stored reference scans for each corresponding reference pose, where each is considered as a possible pose candidate. The two sets of local maps are then matched against one another in an optimisation process aiming to maximise the overlap between two submaps. This process starts out in maps of low resolution, and is progressively increased, whereby the output of the previous step is fed as the initial one of the next.

An alternative to pairwise place-comparison of place descriptors is proposed in [19], which reduces the linear query time to sub-linear levels. This work aims at replacing comparisons at the place descriptor level by keypoint descriptors, since the latter reside at a lower level and results in 2D have demonstrated that their use results in high place recognition rates. From a database of pre-stored local descriptors of the map, a constant number of nearest neighbours is made to

vote for each keypoint extracted from 3D LIDAR scans, and their aggregation determines the likely place matches. The authors find that such a scheme results in vote scores whose distribution follows a log-normal distribution, and are thus able to fit a parametric model of hyperparameters in order for a meaningful voting threshold to be established, one that can reliably distinguish between true and false positives, providing an automatic way of tuning critical algorithmic parameters.

The first use of scan-to-map-scan matching (section II-A) with the use of a 2D range-scan sensor in the context of global localisation is witnessed in [20]. The proposed method first generates the generalised Voronoi diagram of the 2D grid map, whereupon its nodes are taken to be initial hypotheses of where the robot is posed, and from there virtual scans over an angular range of $2\pi$ are taken using raycasting on the grid map. Correspondences between each virtual scan and the scan captured from the physical sensor are then established by using a spectral technique [21] that finds pairwise geometric relationships between them. These correspondences are then used to generate 2D geometric histograms that encode a sense of similarity between the true scan and all virtual scans. The nodes from which the latter were captured are then ranked according to this similarity measure and a threshold based on the correlation coefficient of all combinations of scans is used to extract a subset of candidate poses. This process is used to quickly sift between all candidate poses. The final pose is that which achieves the maximum number of correspondence pairs after the same spectral scan matching process used before.

Inspired from the computer vision research, the method introduced in [22] makes use of signatures of landmarks extracted from 2D LIDAR scans, while explicitly taking into account the relative orientation and distance of landmarks in the same scan. Landmarks are points of high curvature, shown to be adequately descriptive in the domain of 2D range data [23]. For each set of landmarks extracted and stored a priori during a SLAM session, the distribution of spatial relations between these landmarks is captured and stored. A 2D histogram is then constructed from a 2D bin grid in which these relations are encoded by centering a 2D normal distribution over each bin. The signature of each landmark is computed as the sum of of all Gaussians over that landmark, and the signature of the range scan is consequently computed as the sum of all signatures of that scan's landmarks. The normal distribution is used so that, in this procedure of voting, uncertainties in relative depth and orientation can be incorporated. Once these signatures are stored in a database, global localisation can be performed on the basis of approximate nearest neighbour search of signatures extracted from the real scan sensor at the time of global localisation in this database. The resulting pose is that whose signature has the least distance to that of the input range scan.

Far from most established techniques in research around global localisation, the method introduced in [24] utilises neural networks as a means of acquiring the resting pose of a robot, without the use of a traditional map. Instead, at the initial step, the robot is made to traverse its environment and an input pair of an image from the robot's front-facing RGB camera and a pose from which it was captured are stored in a database. After collecting all such pairs, a neural network is trained on the captured images so that the system learns to output the unique index of each image. In this process, each neuron samples the input image as a whole, and the gaussian-filtered version of the same image in a feature-based fashion, so that high frequency noise is effectively eliminated while details of each scene are maintained. When global localisation is requested, the system captures one image from the robot's camera, and inputs it to the neural network, where all neurons output their inferred index. Voting for each index decides the robot's final pose estimate.

The authors of [25] argue that in [22], the geometric landmark relations signatures, although robust, are not invariant to rotations. The method in this paper use improved falko keypoints and fast point-to-point association in order to match scans for place recognition. Although this technique is limited to place recognition in the context of loop closure during slam, we find that it could be extended to scan-to-map-scan matching for global localisation. See therein for a survey on keypoint and keypoint descriptors.

The method introduced in [26] and [27] globally localises a robot using a RGBD camera together with a 2D LIDAR by using visual features in assistance to the range scan sensor. To this end, when the map of the environment is being built, concurrently to the LIDAR-derived map, a visual information map is constructed in a preprocessing step, and the poses from where the visual info is captured are also stored along the way. The visual info stored is a series of keyframes and their corresponding GIST [28][29] descriptors. When global localisation is triggered, the gist vector from a RGB image of the true camera sensor is computed, and its Minkowski distance to all stored pre-stored gist vectors is used to rank the $n$-top matches. These are then clustered, and the transformation between the robot's pose and the global frame of reference is inferred by computing the transformation between the closest image to the cluster's centroid and the input image, considering the pose from which the former was captured.

The authors of [30] introduce a novel descriptor for 3D LIDAR scans based not on their range component, but on their intensity component. This descriptor is run for each of the 3D point clouds that were stored at the time of the environment's map generation and also for the 3D point cloud captured at global localisation time. The latter is first divided into bins for which an intensity histogram is computed. These histograms are then combined into an intensity-based global descriptor whose similarity with those extracted from the global 3D point cloud map is compared. Once the most similar cloud is identified, local geometric descriptors are used to find keypoint-to-keypoint correspondences, and these are used to provide the full 6DOF transform between the two input point clouds.

In a step forward from previously mentioned approaches, the method proposed in [31], termed GLFP, does not rely on an a priori built map but on a floor plan of the environment the

robot is tasked to globally localise itself in. In order to overcome the gap between having visited the environment before and the being supplied only with its floor plan, this method identifies features that co-occur both in a low-quality bird's eye map and from the robot's 3D LIDAR sensor: GLFP extracts vertical edges from the input 3D point cloud and corners from the map floor plan. Data association is then performed using max-mixtures [32] and nearest neighbour methods. The robot's pose is then found via a factor graph-based algorithm, an optimisation problem where both landmark positions and the robot's pose are treated as variables.

The method introduced in [33] extends SA-MCL [34] [35] for use in conditions where multiple 2D or 3D LIDAR sensors utilised for localisation are positioned on an AGV in an arbitrary spatial configuration. Both use the underlying MCL [42] mechanism for global localisation, i.e. particles are dispersed over the map and the likelihood field is used as a measurement model in order to locate the particle that explains the measurement from the LIDAR the best among all particles. However, in SA-MCL particles are not dispersed uniformly over the entirety of the map, but only over similar energy regions. In a preprocessing step, each cell of the 2D grid map is associated with an energy value that encodes proximity to obstacles. At global localisation time, the energy of the input measurement is computed and compared against that of all cells in the grid. Those under a certain manually-set threshold are those over which particles will be distributed.

### *B. The Fourier-Mellin Transform in Robotics*

The Fourier-Mellin Transform (FMT) has received limited attention in the context of mobile robotics, both in scope and extent, due mainly to its ability/constraint of being applicable to 2D grids/images. Most of its applications are hence limited to robots using sensors that directly or indirectly produce images, i.e. monocular camera, sonar, or radar sensors. Additionally, it has been used in merging/aligning of digital maps, while most applications exploit it in the task of mapping or odometry estimation.

A method for ego-motion estimation using images from a panoramic field-of-view radar sensor is presented in [61] and [62], based on Frequency Modulated Continuous Wave technology [63], in the context of Simultaneous Localisation and Mapping (SLAM). The authors argue that the sensitivity of 2D range scans sensors to atmospheric conditions in outdoor scenarios has given rise to carrying out SLAM with radars and sonars, and their research focuses on the former due to a radar sensor's large range with low transmission power, and its inherent ability to estimate sharp changes of temporal variables more easily in the frequency domain rather than the temporal one. Ego-motion estimation based on radar images is carried out via the Fourier-Mellin Transform, where the latter is inputted consecutive radar images and outputs their relative translation and rotation parameters, which are exactly those relating the pose of the robot from which the second image was captured with respect to that from which the first one was captured.

In [64], the basic principle of FMT is combined with a Phase-Only Matched Filter (as opposed to a Symmetric Phase-Only Matched Filter —SPOMF, section II-B), due to the lack of need to extract the scaling factor, in order to register sonar range scans in the context of control of autonomous underwater vehicles and mapping in underwater conditions. The authors remark that a basic correspondence-finding scan-matching algorithm, such as ICP [14] or its notably best-performing variant PL-ICP [49], cannot be used with sonar range scans as inputs, as sonar sensors report not only one single range reading, but a multitude of values corresponding to echo magnitudes at different ranges, thus violating the fundamental assumption of range uniqueness that ICP and its variants adhere to. Additionally, sonar sensors exhibit such noise levels which cause ICP-based algorithms to yield sub-optimal results, typically arising when a sonar beam hits a surface at an angle. In [65] this approach is extended in three dimensions, and in [66] it is adapted to the probabilistic domain, where covariance matrices are fitted around all three translation, rotation and scaling registration results, depending on the intensity in each parameter space, and treated as a probability mass function. This process is integrated into a maximum likelihood mapping framework that is used to build maps of underwater structures from overhead monocular image sequences (rather than sonar sensors) via pose-based graph optimisation.

The same principle is applied in [67] but with an improved FMT method in the context of Unmanned Aerial Vehicles (UAVs) for the purpose of capturing top-down view maps from a camera sensor by stitching together matched images, and which, at the same time, can be used as a form of (visual) odometry for the vehicle. The same method is used in [68] to solve the problem of identifying structural errors in occupancy grid maps generated by SLAM algorithms running in Unmanned Ground Vehicles (UGVs), which are those that arise when regions of the overall map are locally consistent with a ground truth map, but inconsistent with respect to each other, by introducing the concept of a map's brokenness. The solution to this problem is of particular interest in the task of multi-robot mapping, or during SLAM sessions where symmetries in the environment result in erroneous loop closures. Partitions of alignment between a reference (ground truth) map and a potentially misaligned map are detected using a similarity measure derived from this improved version.

In [69] FMT is used to provide the translation and rotation of a UGV relative to a past pose, i.e. odometric information. These transformation parameters are extracted by feeding two images captured at consecutive time steps from a RGB camera mounted at the underside of a vehicle and facing the ground to a Phase-Only Matched Filter FMT algorithm, from which the translation and rotation of the most recent image with respect to the one preceding it are calculated. The pixel-wise parameters are then transformed into robot motions through the intrinsic calibration parameters of the camera sensor, and from there the pose from which the second image was captured is expressed with respect to that from which the first image

was captured.

Similarly to [64], FMT is employed in underwater mapping conditions in [70]. The authors remark that, in underwater situations, the use of optical cameras is prohibitive due to their limited visibility range, but forward-looking sonar sensors are not affected in this respect. However, the hostility of the environment, coupled with the acoustic nature of sonar sensors' operating principle poses severe challenges, as sonar images have low resolution, low signal-to-noise ratio, while they are highly susceptible to inhomogeneous insonification and intensity alterations due to changes in the sensor's viewpoint. These difficulties hinder successful operation of feature-based methods [71][72], especially when accurate loop-closures need to be established. Posing the mapping problem with a sonar sensor as a pose-based graph optimisation problem, the authors show that FMT-based registration of sonar images is robust in the face of the aforementioned hindrances and lack of features, with their method producing globally consistent results.

FMT is utilised in [73] in the context of SLAM-generated occupancy grid maps through the use of standard 2D range scan sensors. FMT is used to match submaps together: during mapping, whenever a fixed number of new scans have been processed, a local submap is generated and stored in a database, along with its pose relative to the frame of reference of the overall map and the pose of the robot relative to the submap; when a loop closure needs to be detected, or when two maps from two different mapping sessions of the same environment need to be merged, FMT is used to find the relative transformation parameters between either the latest constructed submap and a submap generated from the previous time the robot visited the same place, stored in memory, or between the submaps generated and stored during the two distinct sessions, thus augmenting the operational functionality, reliability, and operation time of a standard SLAM algorithm.

In [74] FMT is used in the context of outdoor localisation with the use of a 3D lidar, wheel odometry, and a commercial GPS sensor, as the means of calculating an additional odometry source, and of pose tracking. With regard to the former, successive 3D range scans are projected in the $x - y$ plane, converted to grids, and matched using FMT-SPOMF, the extracted transformation parameters from which provide the translation and rotation of the second scan with respect to the former, and therefore those between the two successive poses from which they were captured. With regard to the latter, a 3D range scan is converted to a grid image, with its center being the pose measured by the GPS sensor, and then matched against a map whose dimensions depend on the pose's uncertainty. A coarse knowledge of the lidar sensor's pose makes it possible to extract a correction to the GPS sensor's pose estimate.

## IV. Motivation & Objective

The majority of global localisation approaches identify features in the map of the environment in which a robot is placed, and in the measurement vector(s) derived from the physical sensor(s) mounted on the robot before establishing correspondences between them. In principle, unstructured environments cannot be relied upon for the existence of features, due to their complete absense, or their sparse and fortuitous distribution. Structured environments on the other hand, and hence their maps, manifest different features depending on the particularities of the environment. In any case, features may be present but not in a sufficiently undisturbed state due to sensor noise or map-to-environment mismatch. Therefore one would not be unreasonable to investigate methods that dispense with features altogether, so as to seek to achieve a greater degree of universality and reliability in multiple and disparate environments.

Another concern that needs to be given attention to is the practice of establishing correspondences. This affects all feature-finding approaches as well as approaches resting on scan-matching and scan–to–map-scan matching, such as ICP methods and its galore of variants. The problem here is that, given the sensor noise and the map-to-environment discrepancies, establishing correspondences may lead to inaccurate results, or altogether to false ones. In practice, with regard to methods that operate directly in measurement space, filtering of outliers is performed through processes that depend on the fulfillment of assumptions and the accurate setting of externally-supplied parameters. The latter include for instance an estimate of the standard deviation of normally distributed, zero-mean noise acting on the measurements of a range scan sensor—when the sensor's measurements may actually be biased, or escape the Gaussian assumption altogether [40]—, or an estimate of the percentage of outliers within their inputs.

This leads us to another crucial point, namely the issue of parameterisation. The performance of the majority of scan-(to–map-scan-)matching methods—not only those based on ICP—rests upon the accurate tuning of parameters that govern their internal processes. In general it is rightly assumed that these parameters need to be tuned for a specific environment and for specific noise levels but in reality, in the absence of online automatic parameter tuning, different parameterisations can lead to volatile or unintuitive results; and this result may be exhibited even for the same pose in the same environment. Let us clarify this quality with a simple yet characteristic example.

Suppose a fairly common scenario: global localisation is to be carried out in an environment by a robot equipped only with a 2D LIDAR sensor. Suppose also that for generality purposes, i.e. for use in unknown, different, and disparate environments, the underlying method of localisation is required to be robust and thus avoid using features. One way to solve this problem is to disperse a finite number of pose hypotheses into the unoccupied space confined within the borders of the map and perform a scan–to–map-scan matching between the range scan received from the physical LIDAR sensor and the map-scan derived from each hypothesis. A natural choice would be to employ ICP or one of its variants, as they have demonstrated their efficacy and robustness in various contexts [46][49][14]; all deal in correspondences and require parameters to be tuned, those relating to operations including, but not limited to, establishing correspondences by filtering outliers. The result

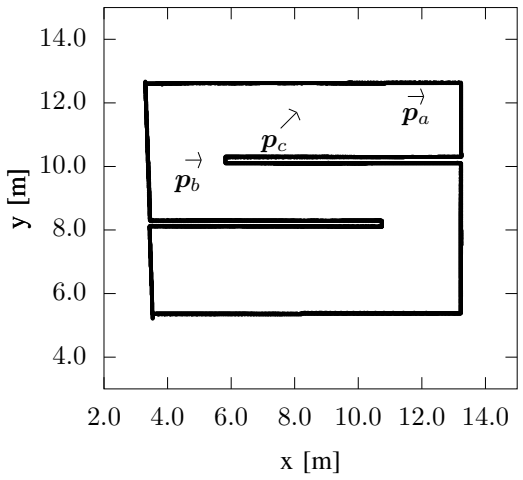



Fig. 3: The map of environment CORRIDOR, $\boldsymbol{M}_C$, and two poses within it, $\boldsymbol{p}_a(11.56, 12.2, 0.0)$, and $\boldsymbol{p}_b(4.56, 10.2, 0.0)$

of this method of global localisation would then be the pose for which ICP reports the lowest matching error (eq. 1).

Figure 3 shows the map of a simple, structured, non-complex environment termed CORRIDOR, $\boldsymbol{M}_C$, in which we performed a number of simulations with a robot equipped with a 2D panoramic LIDAR sensor. In order to examine how the output of an ICP method is influenced by the modification of its parameters, we choose PLICP [49]—the most robust and best-performing variant of ICP-based scan-matching methods in robotics—to carry out the localisation of the robot. A default parameter configuration serves as the basis from which 8 key parameters are modified once and then set to their default value. In order to survey its solution landscape, we ran global localisation under the above-mentioned scan–to–map-scan matching regime, for robot poses $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$, and two different levels of noise, with a fixed number of fixed pose hypotheses, for 10 times, making $N_S = 2 \times 2 \times 14 \times 10 = 560$ simulations. The placement of poses was kept fixed so as to be able to perform direct comparisons; care was taken so that the entirety of the unoccupied space was filled with hypotheses, and hence PLICP was not starved for matches. Table I illustrates the parameters under modification and the total pose error for each solution found. The pose error concerning true pose $\boldsymbol{p}(x, y, \theta)$ and the solution $\hat{\boldsymbol{p}}(\hat{x}, \hat{y}, \hat{\theta})$ to the global localisation problem for that resting pose is denoted by $e(\boldsymbol{p}) = \left((x-\hat{x})^2 + (y-\hat{y})^2 + (\theta-\hat{\theta})^2\right)^{1/2}$. The exceptional behaviour of PLICP—beyond the literature—and the well-behavedness of the default parameter-set have been determined as such after extensive tests during research over tandem combinations of particle filters with scan–to–map-scan matching [75]. Details of the meaning and use of each stated and modified parameter can be found in [76] and [77].

Let us begin our analysis of the volatile behaviour of PLICP by focusing on the results when sensor noise is absent. For these two particular poses, in this particular environment, modifying parameters relating to the use of enhanced methods of finding correspondences (`use_corr_tricks`), restarting when a solution exceeds a threshold (`restart`), clustering of points in the plane (`clustering_threshold`), testing a solution while considering the orientation of the normal of the surface of scans (`do_alpha_test`), and the number of neighbouring rays used to estimate the orientation (`orientation_neighbourhood` $= 2 - 20$)—modifying these seems to have no effect on the solution for each tested robot pose. If we examine the pose error with regard to these parameters when sensor noise is present, we observe that modification of a parameter may have positive impact on the solution for one pose but negative for another (e.g. `use_corr_tricks`, `clustering_threshold`). In addition, functionality whose purpose is to enhance the method's performance does not always lead to the desired result (e.g. `use_corr_tricks`, `restart`). The positive modification of other parameters (e.g. `outliers_remove_doubles`) produces consistent results across sensor noise levels for one pose ($\boldsymbol{p}_b$), inconsistent for others ($\boldsymbol{p}_a$), or altogether catastrophically false (`orientation_neighbourhood` $=$ 200).

The highest sensitivity of PLICP, however, is exhibited with regard to parameters relating to filtering of outlier correspondences, denoted by the prefix `outliers_`. The value of $1.0-$`outliers_maxPerc` determines the percentage of correspondences with the most error to be discarded, while the value of `outliers_adaptive_order` determines the lower-most percentage of correspondences (according to their error) on which an adaptive algorithm for discarding correspondences is run. With regard to the former what we observe for both robot poses is that discarding the top $30\%$ of most-erroneous correspondences results in catastrophic failure in the absence of sensor noise, but accurate behaviour in the face of disturbances. As for other values, no consistent behaviour is observed, although all result in correct convergence. With regard to the latter, the inconsistency between results emerges at the level of different poses; setting this parameter to $70\%$ exhibits increased accuracy for $\boldsymbol{p}_a$, but catastrophic failure to converge for $\boldsymbol{p}_b$.

The above-carried analysis was performed in order to illustrate the perplexities into which a sound method can find itself when relying on tuning of delicate intrinsic parameters. Even if all tested values resulted in correct convergence, the issue of (in)consistency across different sensor noise levels and different poses within the same map would still remain, along with that of inconsistency of intuition about their effect. Wherefore we conclude that the advantage of tailoring parameters to specific circumstances is not without its merits but also its side-effects. Subsequently, and in recapitulation, it would be meritorious for research to focus on pose estimation methods that dispense with establishing correspondences and tuning of parameters for different poses and environments. This leads to the formulation of the objective of this article:

**Objective *O*.** The aim of this article is to formulate a

| Solution error $e$ regarding robot pose $\boldsymbol{p}_*$ | | | $e(\boldsymbol{p}_a)$ | | $e(\boldsymbol{p}_b)$ | |
|---|---|---|---|---|---|---|
| Sensor noise $\mathcal{N}(0,\sigma)$, $\sigma$: | | | 0.0 | 0.01 | 0.0 | 0.01 |
| Default parameter set | | | 0.006579 | 0.005601 | 0.036817 | 0.037745 |
| Parameter modified | value | default | | | | |
| `use_corr_tricks` | true | false | 0.006579 | 0.006401 | 0.036817 | 0.037744 |
| `restart` | true | false | 0.006579 | 0.007433 | 0.036818 | 0.037786 |
| `clustering_threshold` | 1.025 | 0.025 | 0.006579 | 0.006051 | 0.036818 | 0.038122 |
| `do_alpha_test` | false | true | 0.006579 | 0.006972 | 0.036817 | 0.038493 |
| `orientation_neighbourhood` | 2 | 20 | 0.006579 | 0.006628 | 0.036818 | 0.037225 |
| | 200 | | 15.425727 | 15.425727 | 9.915300 | 9.915300 |
| `outliers_maxPerc` | 0.9 | 1.0 | 0.004319 | 0.006053 | 0.035770 | 0.035864 |
| | 0.8 | | 0.004486 | 0.005135 | 0.035900 | 0.038418 |
| | 0.7 | | 4.709158 | 0.004731 | 10.298579 | 0.037788 |
| `outliers_adaptive_order` | 0.9 | 1.0 | 0.004472 | 0.004722 | 0.035586 | 0.036388 |
| | 0.8 | | 0.004359 | 0.005790 | 0.036678 | 0.036897 |
| | 0.7 | | 0.004272 | 0.004062 | 2.922564 | 4.498574 |
| `outliers_remove_doubles` | true | false | 0.006227 | 0.006404 | 0.036268 | 0.036732 |

TABLE I: The pose error $e(\boldsymbol{p})$ for the best match found by PLICP in the environment CORRIDOR (fig. 3) over $N_S$ simulations for a default parameter set and for varying values of core parameters, and two levels of sensor noise, which is assumed to be normally distributed with standard deviation $\sigma$ [m]. The unit of measurement of the pose error is $(\mathrm{m}^2 + \mathrm{rad}^2)^{1/2}$

method that solves the passive variant of the problem of global localisation (problem *P*, definition I), considering the afforded laxness of remarks I and II, for a robot equipped with a 2D LIDAR sensor, that does not require associating correspondences, or the ad-hoc tuning of parameters that govern its response.

In section V it is demonstrated how scan–to–map-scan matching (definition V), used in conjunction with Fourier-Mellin Invariant matching (section II-B) and the calculation of the centroid of the polygon whose vertices consist of the projection of a 2D range scan's endpoints in the $x-y$ plane (section II-C) can be combined in such a way as to achieve objective *O*.

## V. Methodology

### A. Overview

The structure of the overall passive global localisation system proposed in this article, denoted by the acronym PGL-FMIC (Passive Global Localisation–Fourier-Mellin Invariant matching with Centroids for translation), is illustrated in figure 4. The system requires as inputs the scan captured from the robot's 2D LIDAR sensor $\mathcal{S}_r$, the map of the robot's environment $\boldsymbol{M}$, and the number of pose hypotheses to generate over the unoccupied space demarcated by the borders of the map, $|\mathcal{H}|$. Once the set of hypotheses $\mathcal{H} = \{\boldsymbol{h}_i\}$, $i = \{0, 1, \ldots, |\mathcal{H}| - 1\}$ is generated, its contents are placed in a queue $\boldsymbol{q}$ and serially dequeued and inputted to the core method, denoted by the acronym FMIC (detailed in sections V-B and V-C). Its output reports a global pose candidate $\boldsymbol{h}_i''$, a scale factor $\sigma_i$, and a similarity measure $w_i$; the external meaning and utility of the latter two will be detailed in section V-D. All three outputs per hypothesis are stored and, when the queue empties, the fittest global pose candidate hypothesis is reported as the system's pose estimate via a sifting and ranking process. The latter discards candidates based on their reported scale factor $\sigma_i$ before selecting that whose similarity measure $w_i$ is the highest among all candidate hypotheses (subsection V-D).

Figure 5 depicts the internal structure of the core method proposed in this article, FMIC. Once a hypothesis $\boldsymbol{h}_i$ is inputted, the method tries to estimate the orientation of the robot first, and then its position within the map $\boldsymbol{M}$. The internals of these two operations are detailed in sections V-B and V-C respectively.

### B. Extraction of Rotation and Scale

Figure 6 depicts the rotation and scale estimation process.

Given a pose hypothesis $\boldsymbol{h}_i$, the scan captured from the physical sensor $\mathcal{S}_r$, and a map $\boldsymbol{M}$, the rotation and scale estimation subsystem tries to estimate (a) the relative orientation of $\mathcal{S}_r$ with regard to the virtual scan taken from $\boldsymbol{h}_i$, (b) the scale factor between the two scans $\sigma_i$ and (c) a measure of their similarity $w_i$.

The orientation estimation module first computes the virtual scan from $\boldsymbol{h}_i$ (definition IV), i.e. a range scan captured from pose $\boldsymbol{h}_i$ within $\boldsymbol{M}$, producing a virtual scan $\mathcal{S}_{\mathrm{v}}^i$. At this point two range scans are available: one from the physical range scan sensor of the robot (definition III), captured from its true pose, and one from the virtual range scan sensor, captured from random pose $\boldsymbol{h}_i$. The endpoints of the two scans' rays are then projected to the $x-y$ plane, according to formulae

$$x_r = x_0 + d_r \cos(\frac{2\pi r}{N} - \pi + \theta_0) \tag{17}$$

$$y_r = y_0 + d_r \sin(\frac{2\pi r}{N} - \pi + \theta_0) \tag{18}$$

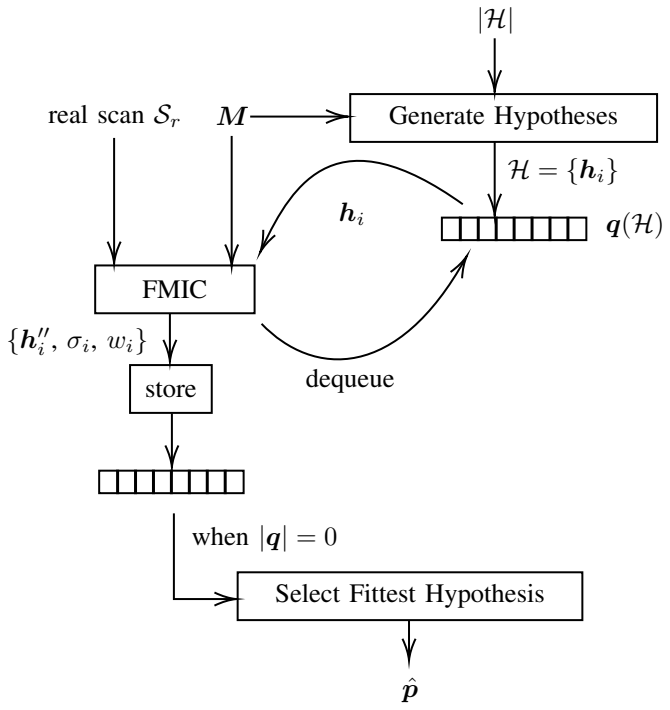



Fig. 4: The structure of the overall passive global localisation system PGL-FMIC. Once the set of pose hypotheses $\mathcal{H}$ is generated and ordered in queue $\boldsymbol{q}$, its contents are fed one-by-one to a system estimating global pose candidates $\boldsymbol{h}_i''$ and measures of the value of their estimate $\sigma_i$ and $w_i$. Once all hypotheses have been processed, the two measures are used to determine the overall system's global pose estimate output $\hat{\boldsymbol{p}}$

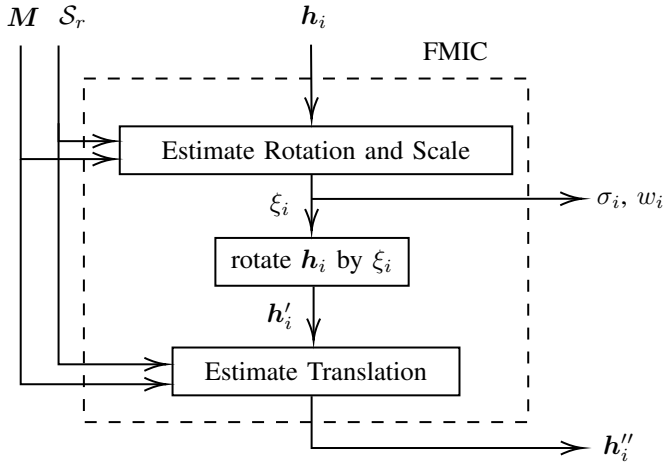



Fig. 5: The structure of the core algorithm of the system, FMIC. Given the map of the robot's environment $\boldsymbol{M}$, the scan captured from its true pose $\mathcal{S}_r$, and a pose hypothesis $\boldsymbol{h}_i$, an estimate of the robot's orientation relative to $\boldsymbol{h}_i$ is extracted first, $\xi_i$, along with measures $\sigma_i$, $w_i$ that determine the output pose's $\boldsymbol{h}''$ value as a global pose candidate. Subsequently, the position of the global pose candidate is determined

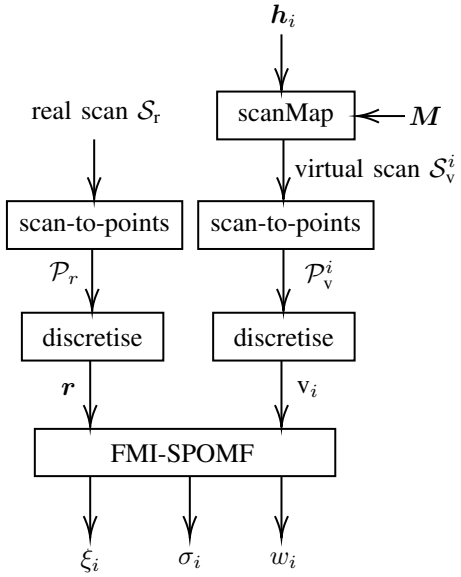



Fig. 6: The internal structure of the rotation and scale estimation subsystem (fig. 5)

where $(x_r, y_r)$ are the coordinates of the endpoint of ray $r$, $r = \{0, 1, \ldots, N-1\}$, $d_r$ the range reported for $r$, and $N$ the number of rays emitted by the physical range scan sensor. The triplet $(x_0, y_0, \theta_0) \triangleq (0, 0, 0)$, that is, the endpoints of both scans are projected in the local coordinate system of each range scan sensor. Point-sets $\mathcal{P}_r$ and $\mathcal{P}_\mathrm{v}^i$ are then subject to discretisation over a fixed grid size $N_G \times N_G$ (which is advised to be square with $N_G = 2^c$ where $c \in \mathbb{Z}^+$ due to the efficiency of FFT when dealing with grids whose size is a power-of-two [64]), which produces 2D grids $\boldsymbol{r}$ and $\mathrm{v}_i$ respectively. The two are then inputted to the FMI-SPOMF process (algorithm II), which yields a rotation angle $\xi_i$, scale factor $\sigma_i$, and similarity degree between the two images $\boldsymbol{r}$ and $\mathrm{v}_i$, denoted by $w_i$.

The applicability of FMI-SPOMF with respect to the discretised versions of point-sets $\mathcal{P}_r$ and $\mathcal{P}_\mathrm{v}^i$ is feasible and valid in both situations where a pose hypothesis falls in the vicinity of the true pose and in that where it does not:

- In the former case the two point-sets consist of a majority of points representing portions of the environment/map visible from both poses, and a minority of points visible exclusively from one but not the other. This effect is attributed solely to the displacement of the pose hypothesis with regard to the robot's true position. Points in the first category are related to one another via translation and rotation due to world-to-map consistency (with scale playing a role whose significance is inversely proportional to the map's resolution). Points in the second category are viewed as noise or non-linear distortions, thereby having a decreasing effect on the similarity measure between the images corresponding to the two point-sets. Notwithstanding this type of discrepancy, the similarity degree between the two images and the quality of the estimate of translation, rotation, and scale are proportional to

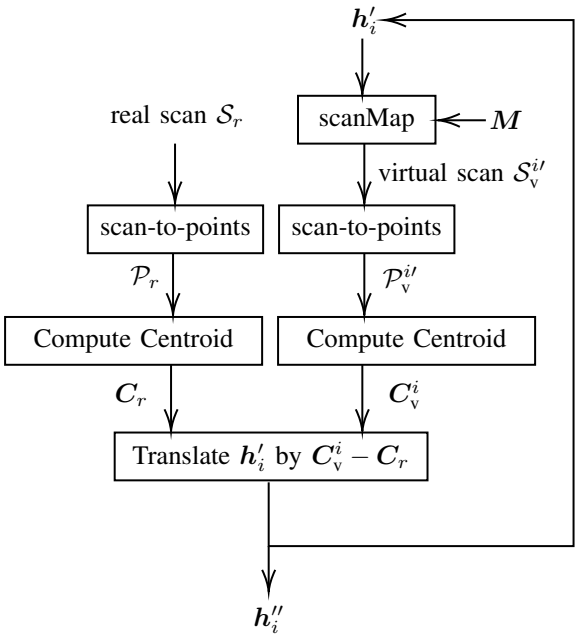



Fig. 7: The internal structure of the translation estimation subsystem (fig. 5)

the percentage of the points in the first category over that of the second—an effect largely attributed to the documented robustness of FMI-SPOMF [58].

- In the latter case the output of algorithms I and II and, most crucially, the similarity measure $w$ are all arbitrary; yet, if a second pose hypothesis happens to fall in the vicinity of the true robot's pose, then, in addition, they are also lower in magnitude than those of the second hypothesis, making distinction between the two feasible.

Although FMI-SPOMF is able to extract the translation offset between the two images, the fact that (a) these are captured from each sensor's local frame of reference, and (b) the robot's true pose is unknown, renders the translation vector without physical meaning. However, if the pose hypothesis $\boldsymbol{h}_i$ resides in the vicinity of the robot's true pose, once the former is rotated by $\xi_i$, an estimate of the translation of the former with respect to the latter can be extracted. The positional displacement correction process is detailed in section V-C.

### C. Extraction of Translation

Once the pose hypothesis $\boldsymbol{h}_i$ is rotated by the angle found by FMI-SPOMF, $\xi_i$, the orientation of the robot has been estimated. The resulting pose hypothesis, denoted by $\boldsymbol{h}'_i$, is then inputted to the position estimation module. Its block diagram is depicted in figure 7.

At this point a new map scan $\mathcal{S}_{\mathrm{v}}^{i\prime}$ is computed from the angularly-corrected pose hypothesis $\boldsymbol{h}'_i$. Its end-points along with those of the range scan derived from the physical sensor are again projected into the $x-y$ plane with the use of equations 17 and 18. In these equations, having the orientation been estimated, the value of $\theta_0$ is substituted by the robot's orientation estimate $\theta_{\boldsymbol{h}_i} + \xi_i$. The two resulting point-sets $\mathcal{P}_r$ and $\mathcal{P}_{\mathrm{v}}^{i\prime}$ are now aligned with respect to orientation and centered around the origin $O(0,0)$. Subsequently the centroid of each polygon with vertices $\mathcal{P}_r$ and $\mathcal{P}_{\mathrm{v}}^{i\prime}$ is calculated through equations 15 and 16. If a pose hypothesis is in the vicinity of the robot's true pose, then the centroid of $\mathcal{P}_{\mathrm{v}}^{i\prime}$ will be translated with respect to that of $\mathcal{P}_r$ by the offset between the pose hypothesis $\boldsymbol{h}_i$ and the robot's true pose—due to remarks VI and VII—plus some additional offset caused by points which are visible from the one but not the other. This is again due to the fact that different parts of the environment/map become (in-)visible from different poses. The position of pose hypothesis $\boldsymbol{h}'_i$ is then corrected by the difference in position between the two centroids $\boldsymbol{C}_{\mathrm{v}}^{i}$ and $\boldsymbol{C}_r$, and this process is iterated until convergence has been achieved, or a maximum number of iterations has been met. The output of this process is the final pose estimate $\boldsymbol{h}''_i$.

### D. Fittest Hypothesis Selection

After all $|\mathcal{H}|$ hypotheses have been processed, a collection of equal in number triplets $\{\boldsymbol{h}''_i, \sigma_i, w_i\}$ is due for sifting in order to determine the final pose estimate of the robot. In theory, for the closest pose hypothesis $\boldsymbol{h}_c$ to the robot, FMI-SPOMF should report a scaling factor $\sigma_c = 1.0$ and the highest similarity degree among all pose hypotheses, $w_c = \max\{w_0, w_1, \ldots, w_{|\mathcal{H}|-1}\}$. However, in practice, violations of these conjectures may manifest in this article's context due, for instance, to ambiguities arising from similarities between discrete portions of the map[1] or due to the fact that the map may be an Occupancy Grid Map (as is typical in indoor robotic applications) of finite resolution: obviously the lower the resolution the more the absolute value of the scaling factor deviates from its theoretical limit.

Although the relevant literature dispenses with the scaling factor (since the environment, range scans derived in it, and its map are all of the same scale), we have found that due to discretisation errors,[2] sensor noise, map discrepancies with respect to the modeled environment, and the existence of map resolution, the scale factor varies around $1.0$ even for hypotheses lying in the vicinity of the robot's true pose. For pose hypotheses which do, the value of their corresponding scaling factors may be closer to one than that for poses which do not; therefore an initial sift of candidate poses consists in discarding hypotheses $\boldsymbol{h}_i$ for which $\underline{\sigma} < \sigma_i < \overline{\sigma}$ applies, where $\underline{\sigma} \leq 1.0$ and $\overline{\sigma} \geq 1.0$. As for the remaining hypotheses the value of the similarity degree $w_i$ between the true sensor's range image $\boldsymbol{r}$ and a virtual sensor's range image $\mathrm{v}_i$ is, in

[1] Consider for instance a map of two empty rooms, identical in proportions but not in length and width; the similarity degree reported for two hypotheses centered in each room may be equal, but the scale factor between them will vary, and the pose hypothesis which should be reported as the robot's pose estimate should be the one for which the scaling factor is closest to $1.0$

[2] Discretisation errors arise due to (a) intersections of rays with the map during the computation of a virtual scan when the map is represented by an occupancy grid, which is of finite resolution, and (b) the loss of resolution during the discretisation procedure itself when converting a range scan to a point-set and then to a 2D grid

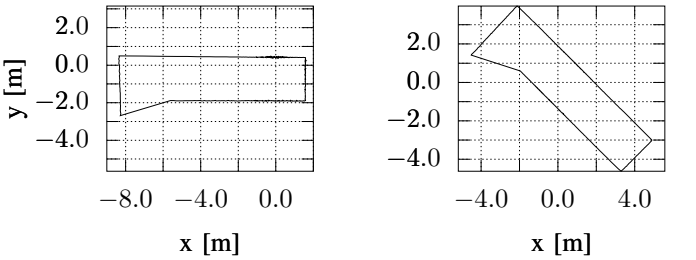
theory, proportional to the similarity between the two images. Therefore the same applies between the two range scans $\mathcal{S}_r$ and $\mathcal{S}_{\text{v}}^i$, and to the proximity between the true robot's pose and hypothesis $\boldsymbol{h}_i$. It is consequently reasonable that the hypothesis with the highest similarity degree, after appropriate rotation and translation, be the system's reported final pose estimate $\hat{\boldsymbol{p}}$.

### *E. Illustration*

Let us illustrate the methodology introduced with an example: consider again the map depicted in figure 3, and let $\boldsymbol{p}_a(11.56, 12.20, 0.0)$ [m, m, rad] be the robot's true pose and a pose hypothesis $\boldsymbol{p}_c(7.56, 11.20, \pi/4)$ [m, m, rad] be disposed by $(-4.0, -1.0, \pi/4)$. At the rotation estimation stage the range scan captured from the robot's true pose, $\mathcal{S}_r^a$, and the virtual range scan captured from the hypothesis, $\mathcal{S}_{\text{v}}^c$, are projected to the $x-y$ plane as if each was captured from $(0, 0, 0)$. Figure 8a shows the projected range scan points from $\boldsymbol{p}_a$, $\mathcal{P}_r^a$, while figure 8b shows those from $\boldsymbol{p}_c$, $\mathcal{P}_{\text{v}}^c$. Notice that these connected point-sets consist of the surroundings of the real and virtual range scan sensors from their local reference frame perspective. These point-sets are then converted into 2D grids via discretisation, inputted to FMI-SPOMF, and the rotation angle between them is used to align the orientation of $\boldsymbol{p}_c$ with respect to that of $\boldsymbol{p}_a$.

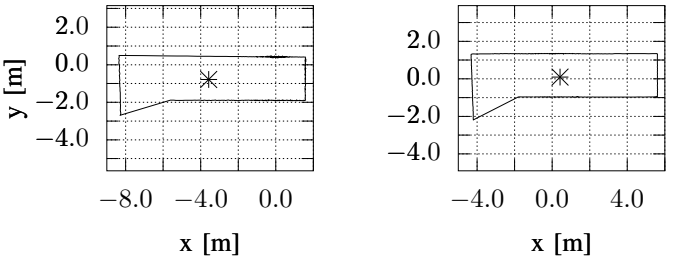
Once the orientation of the hypothesis is corrected, a new map scan is captured from the renewed hypothesis $\boldsymbol{p}'_c$. Then the centroids of $\mathcal{P}_r^a$ and the point set of the newly projected map scan $\mathcal{P}_{\text{v}}^{c\prime}$ are computed. Figure 8c depicts $\mathcal{P}_r^a$ and its centroid $\boldsymbol{C}_a(-3.57, -0.78)$ [m, m], while figure 8d depicts $\mathcal{P}_{\text{v}}^{c\prime}$ and its corresponding centroid $\boldsymbol{C}_c(0.42, 0.09)$ [m, m]. Notice how the two shapes are almost identical, but differ in terms of their position in the $x-y$ plane. Notice also the discrepancy between these two points sets at the left-hand side: due to the offset between the positions of $\boldsymbol{p}_a$ and $\boldsymbol{p}'_c$, a larger proportion of the map is visible from the latter, and therefore the difference between their corresponding points-sets' centroids $\boldsymbol{C}_a - \boldsymbol{C}_c = [3.99, 0.87]^\top$ does not correspond exactly with the difference in position between the two poses, which is $[4.0, 1.0]^\top$. Adding, however, $\boldsymbol{C}_a - \boldsymbol{C}_c$ to $\boldsymbol{p}'_c$ and repeating the same translation estimation process makes $\mathcal{P}_{\text{v}}^{c\prime}$ converge to $\mathcal{P}_r^a$, and therefore $\boldsymbol{p}'_c$ to $\boldsymbol{p}_a$. Figure 8e shows the final point-set $\mathcal{P}_{\text{v}}^{c\prime}$, which is overlaid in figure 8f (coloured with red) on top of $\mathcal{P}_r^a$ (black).

Let us now examine the possible outcomes of the same process for a false candidate pose, for instance $\boldsymbol{p}_b$. In theory $\boldsymbol{p}_b$ would be either discarded at the end of the angular estimation process due to the extraction of an (viewed externally) arbitrary scaling factor $\sigma \in (-\infty, \underline{\sigma}] \cup [\overline{\sigma}, +\infty)$, or accepted for position estimation, whereupon the position of the hypothesis would in all probability be moved divergently from the robot's true pose. If $\boldsymbol{p}_c \in \mathcal{H}$ then FMT-SPOMF would report a higher similarity degree $w_c > w_b$, and $\boldsymbol{p}_b$ would be filtered out as a true negative. Consequently, if no pose hypothesis resided in the vicinity of $\boldsymbol{p}_a$, the projected range scan images captured from every hypothesis would not be able to be angularly



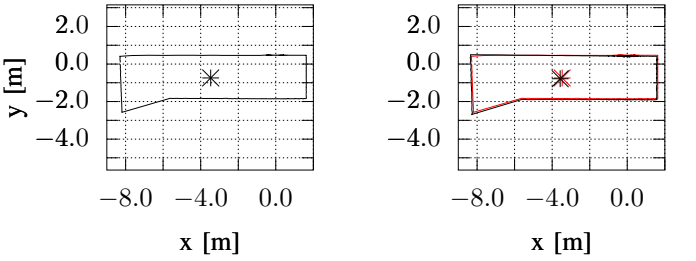

(a) The connected endpoints $\mathcal{P}_r^a$ of the range scan captured from the robot's true pose $\boldsymbol{p}_a$, as seen from the local reference frame of the range scan sensor (effectively $(0, 0)$)

(b) The connected endpoints $\mathcal{P}_{\text{v}}^c$ of the range scan captured from the robot's hypothesised pose $\boldsymbol{p}_c$, as seen from the local reference frame of the virtual range scan sensor $(0, 0)$




(c) The connected endpoints $\mathcal{P}_r^a$ of the range scan captured from the robot's true pose $\boldsymbol{p}_a$, as seen from the local reference frame of the range scan sensor, and the centroid of the polygon formed by them

(d) The connected endpoints $\mathcal{P}_{\text{v}}^{c\prime}$ of the range scan captured from the robot's angularly-aligned hypothesised pose $\boldsymbol{p}'_c$, as seen from the local reference frame of the virtual range scan sensor, and the centroid of the polygon formed by them. Notice that the pose has been aligned angularly, but not positionally




(e) The connected endpoints $\mathcal{P}_{\text{v}}^{c\prime}$ of the range scan captured from the robot's positionally- and angularly-aligned hypothesised pose $\boldsymbol{p}'_c$, as seen from the local reference frame of the virtual range scan sensor, and the centroid of the polygon formed by them

(f) The connected endpoints $\mathcal{P}_r^a$ (black) and $\mathcal{P}_{\text{v}}^{c\prime}$ (red), and their corresponding centroids, all as seen from the local reference frames of the range scan sensors that captured them

Fig. 8: Illustration of orientational and then positional alignment of candidate pose $\boldsymbol{p}_c$ with respect to true pose $\boldsymbol{p}_a$ in environment CORRIDOR (figure 3)

aligned by FMI-SPOMF, and a false hypothesis would be erroneously reported as the system's pose estimate. This leads to the formulation of the following observation:

**Remark VIII.** A pose hypothesis $\boldsymbol{h} \in \mathcal{H}$ that resides in the vicinity of the robot's true pose is a necessary condition for a correct solution to problem $P$ (in the sense of remark I) when approached by a scan–to–map-scan method. In consequence: the supplied number of pose hypotheses $|\mathcal{H}|$ should be proportional to the area of $\boldsymbol{M}$.

In more complex environments, where the environment and its map feature repetitive structures, it may be the case that ambiguity cannot be resolved at all regardless of the number of pose hypotheses. In others, wide open spaces may result to missing information due to sensor maximum range limits. The effects of these conditions may be so pronounced that higher similarity is established between an incorrect pose and the robot's true pose than between the robot's true pose and a pose residing near it. The first issue plagues all global localisation methods as, even for a human, its solution is undecidable, and maximum sensitivity is of paramount importance in such conditions. The second issue is also uncontrollable, as it manifests itself as a limitation imposed by the combination of environment and robot equipment limits.

## VI. Simulations & Results

In order to test the efficacy of the proposed method of correspondenceless passive global localisation, numerous simulations are performed in disparate—structured and unstructured—environments, map configurations and resolutions. In order to test the proposed method's robustness in the face of disturbances, the noise acting on the measurements of the physical range sensor is varied across levels taken from the specifications of commercial sensors. The robot used in all simulations was a Turtlebot v.2, equipped with a single panoramic rangefinder of maximum range $10.0$ m and $N = 720$ rays emitted. The maximum range was set to this value in order to be able to limit the volume of information available to the sensor, and hence to the methods under scrutiny, so as to test their robustness in the face of missing information, in addition to that of uncertain information. In parallel, the proposed method PGL-FMIC (Passive Global Localisation–Fourier-Mellin Invariant matching with Centroids for translation) is pitted against the state of the art ICP algorithm PL-ICP (Point to Line Iterative Closest Point) in an attempt to capture the comparative advantages and disadvantages of the two methods. The fashion in which the latter can be used to tackle the problem of global localisation in a scan–to–map-scan fashion was described in section IV.

Both algorithms were tested in five environments termed CORRIDOR, HOME, WAREHOUSE, WILLOWGARAGE, and LANDFILL, over $38$ robot poses in total, and over four levels of sensor noise. The noise was set to be normally distributed with standard deviation $d \in \mathcal{D} = \{0.0, 0.01, 0.02, 0.05\}$ m, so that both methods are tested against the accuracy of real commercial rangefinders. The ranges of virtual scans are disturbed in proportion to a map's misrepresentation of the environment it corresponds to. The maps of environments CORRIDOR ($\boldsymbol{M}_C$), HOME ($\boldsymbol{M}_H$), WAREHOUSE ($\boldsymbol{M}_W$), and LANDFILL ($\boldsymbol{M}_L$) were constructed with the same robot setup for sensor noise with standard deviation $d = 0.01$ m. The environment from which $\boldsymbol{M}_L$ was generated was constructed from models of the `3DGEMS` dataset [78], while map $\boldsymbol{M}_G$ was taken from [79].

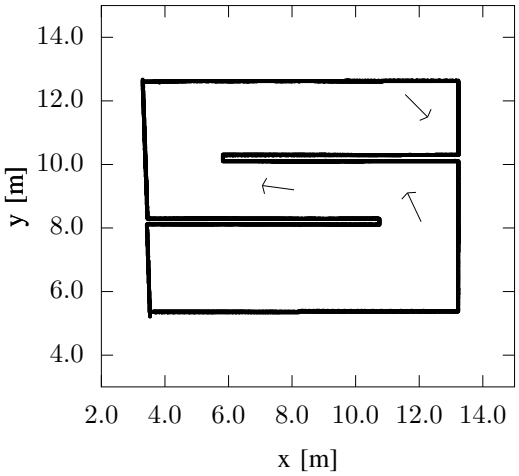



Fig. 9: The map of environment CORRIDOR $\boldsymbol{M}_C$ and three tested true robot poses

A simulation over the determination of one robot pose was conducted for both algorithms $N_s = 100$ times for purposes of reliability of conclusion. The total number of simulations conducted was therefore $38 \times 100 \times 4 \times 2 \sim \mathcal{O}(4)$.

The five simulated environments are depicted in figures 9-13; figure 9 depicts a simple and almost symmetric environment that was used for preliminary and distinction-of-place assessment purposes. Figure 10 depicts a typical domestic or commercial space cluttered with chairs, tables, columns, and box-like furniture. Figure 11 depicts a typical warehouse setting, with wide open spaces in which the methods' ability to cope with missing information is tested. Figure 12 depicts a large office-like complex, where the methods are assessed more closely as to their ability to resolve ambiguities. Figure 13 depicts a non-structured environment similar to a landfill that was used to ascertain the validity of the claim that PGL-FMIC's performance does not discriminate between non-structured and structured environments. The maps of the latter two environments (WILLOWGARAGE and LANDFILL) are of resolution $0.05 \times 0.05$ m$^2$, while that of the other three is $0.01 \times 0.01$ m$^2$.

The positions at which the robot was placed in each environment were determined by the target assessment purpose, i.e. in environment CORRIDOR the robot was placed close to one end of it so as to evaluate the methods' response to environmental symmetry, close to the middle so as to evaluate their response with respect to ambiguity of orientation, and

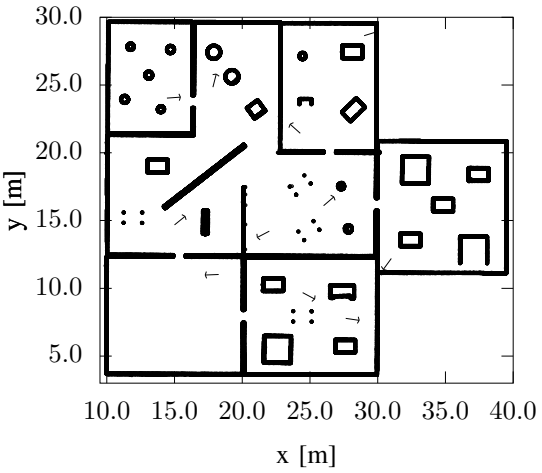



Fig. 10: The map of environment HOME $\boldsymbol{M}_H$ and eleven tested true robot poses

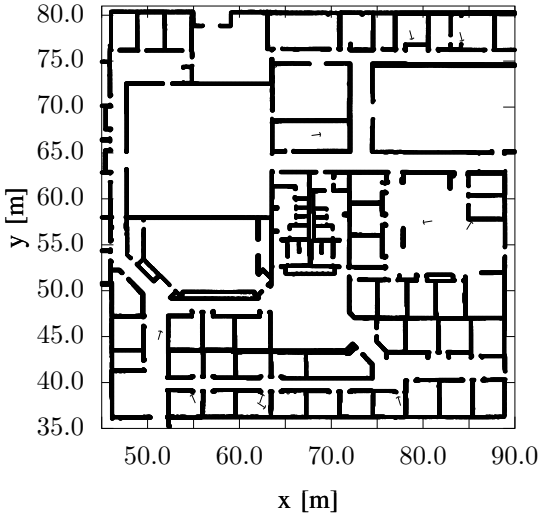



Fig. 12: The map of environment WILLOWGARAGE $\boldsymbol{M}_G$ and ten tested true robot poses

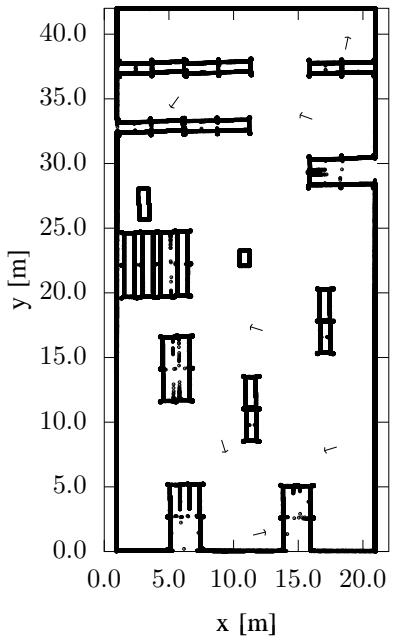



Fig. 11: The map of environment WAREHOUSE $\boldsymbol{M}_W$ and seven tested true robot poses

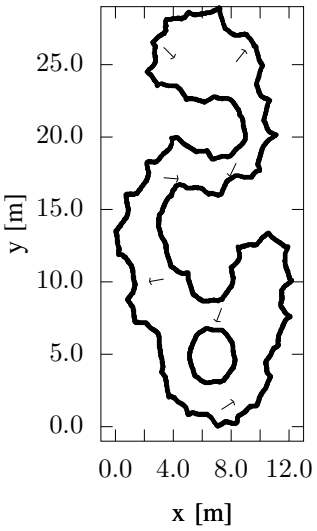



Fig. 13: The map of environment LANDFILL $\boldsymbol{M}_L$ and seven tested true robot poses

near a turn so as to evaluate the effect of non-uniformity of the environment. In environment HOME the robot was positioned at random and at challenging positions, close to or away from objects, and at locations whose surroundings were near-similar to other parts of the environment. In environment WAREHOUSE it was positioned in ways that one would expect that the robot's operation would first be initiated, and in positions such that would challenge the methods' response to missing range of rays (a typical LIDAR sensor reports a reading of maximum range when that ray does not encounter objects in its path). In environment WILLOWGARAGE the robot was positioned in rooms that where either identical or near-identical to others and at random. The orientation of the robot's true pose is of no significance since the range finder sensor is panoramic, and was determined at random by sampling from a uniform distribution $U(-\pi, \pi)$. Table II shows the placement of poses in each environment/map along with their signifiers in subsequent figures.

Given remark I, a correct global pose solution was regarded as such when the deviation of its location from the true pose's

| pose id | $x$ [m] | $y$ [m] | $\theta$ [rad] | sign |
|---|---|---|---|---|
| CORRIDOR | | | | |
| $\boldsymbol{p}_a^C$ | 11.56 | 12.2 | $-0.79$ | $+$ |
| $\boldsymbol{p}_b^C$ | 12.06 | 8.2 | 2.01 | $\circ$ |
| $\boldsymbol{p}_c^C$ | 8.06 | 9.20 | $-3.28$ | $*$ |
| HOME | | | | |
| $\boldsymbol{p}_a^H$ | 14.44 | 24.04 | 0.065 | $+$ |
| $\boldsymbol{p}_b^H$ | 17.84 | 24.84 | 1.33 | $\circ$ |
| $\boldsymbol{p}_c^H$ | 15.0 | 14.68 | 0.68 | $*$ |
| $\boldsymbol{p}_d^H$ | 22.0 | 14.22 | $-2.66$ | $\cdot$ |
| $\boldsymbol{p}_e^H$ | 26.0 | 16.10 | 0.69 | $\times$ |
| $\boldsymbol{p}_f^H$ | 24.46 | 9.68 | $-0.49$ | $\square$ |
| $\boldsymbol{p}_g^H$ | 18.26 | 11.02 | $-3.10$ | $\diamond$ |
| $\boldsymbol{p}_h^H$ | 27.64 | 7.78 | $-0.14$ | $\wedge$ |
| $\boldsymbol{p}_i^H$ | 24.28 | 21.44 | 2.41 | $\vee$ |
| $\boldsymbol{p}_j^H$ | 29.0 | 28.64 | 0.31 | $>$ |
| $\boldsymbol{p}_k^H$ | 31.0 | 12.2 | $-2.19$ | $<$ |
| WAREHOUSE | | | | |
| $\boldsymbol{p}_a^W$ | 8.08 | 3.02 | $-2.85$ | $+$ |
| $\boldsymbol{p}_b^W$ | 35.16 | 15.25 | $-2.20$ | $\circ$ |
| $\boldsymbol{p}_c^W$ | 38.81 | 2.35 | 1.36 | $*$ |
| $\boldsymbol{p}_d^W$ | 33.42 | 4.92 | 2.78 | $\cdot$ |
| $\boldsymbol{p}_e^W$ | 17.08 | 8.75 | 2.83 | $\times$ |
| $\boldsymbol{p}_f^W$ | 8.63 | 11.93 | $-1.26$ | $\square$ |
| $\boldsymbol{p}_g^W$ | 1.27 | 9.5 | 0.239 | $\diamond$ |
| WILLOWGARAGE | | | | |
| $\boldsymbol{p}_a^G$ | 77.56 | 37.48 | $-1.27$ | $+$ |
| $\boldsymbol{p}_b^G$ | 67.85 | 66.90 | 0.13 | $\circ$ |
| $\boldsymbol{p}_c^G$ | 81.0 | 57.60 | $-2.97$ | $*$ |
| $\boldsymbol{p}_d^G$ | 78.55 | 78.35 | $-1.31$ | $\cdot$ |
| $\boldsymbol{p}_e^G$ | 84.0 | 78.15 | 0.94 | $\times$ |
| $\boldsymbol{p}_f^G$ | 84.75 | 56.65 | 1.02 | $\square$ |
| $\boldsymbol{p}_g^G$ | 51.15 | 44.65 | 1.29 | $\diamond$ |
| $\boldsymbol{p}_h^G$ | 61.95 | 37.80 | $-0.58$ | $\wedge$ |
| $\boldsymbol{p}_i^G$ | 62.20 | 37.76 | $-1.22$ | $\vee$ |
| $\boldsymbol{p}_j^G$ | 55.20 | 37.76 | 0.91 | $>$ |
| LANDFILL | | | | |
| $\boldsymbol{p}_a^L$ | 3.34 | 17.19 | $-0.082$ | $+$ |
| $\boldsymbol{p}_b^L$ | 3.34 | 26.19 | $-0.78$ | $\circ$ |
| $\boldsymbol{p}_c^L$ | 8.34 | 25.19 | 0.90 | $*$ |
| $\boldsymbol{p}_d^L$ | 3.34 | 10.19 | $-2.97$ | $\cdot$ |
| $\boldsymbol{p}_e^L$ | 7.34 | 8.19 | $-1.93$ | $\times$ |
| $\boldsymbol{p}_f^L$ | 7.34 | 1.19 | 0.58 | $\square$ |
| $\boldsymbol{p}_g^L$ | 8.34 | 18.19 | $-2.02$ | $\diamond$ |

TABLE II: The robot's true poses tested per simulated environment and the sign by which they are referred to in figures

location was less than 1.0 m.[3]

Those residing outside this circle are hereafter termed outliers. No error threshold was placed on orientation, as it is not certain that a probabilistic pose tracking method, used in subsequence to global localisation, would fail to further localise the robot and track its pose. In terms of the magnitude of the hypothesis set $\mathcal{H}$ of the hypotheses dispersed in each environment, contingent on remark VIII, no method was starved for matches in either environment: for environment CORRIDOR $|\mathcal{H}_C| = 100$; for environments HOME and WAREHOUSE $|\mathcal{H}_H| = |\mathcal{H}_W| = 200$; for environment WILLOWGARAGE $|\mathcal{H}_G| = 500$; and for environment LANDFILL $|\mathcal{H}_L| = 100$. The scale thresholds were set to $(\underline{\sigma}, \overline{\sigma}) = (0.9, 1.2)$. The width and height of images inputted to FMI-SPOMF was set to $N_G = 2^8$; experimental tests showed that such setting resulted in both low execution times and accurate distinction between true and false positive solutions. Higher values of $N_G$ would increase the method's sensitivity with regard to distinguishing between poses set at similar surroundings, but at a cost of increased execution time.

All simulations were conducted in the Gazebo environment[4] through ROS[5], in C++ and Ubuntu 16.04, with a processor of 12 threads, running at 4.00 GHz, using up to 32Gb of RAM. Maps of environments were constructed using ROS's `slam_karto` package[6]. As an implementation note with regard to maps and mapping, we stress caution in that the result of the mapping algorithm may be imperfect in the sense that free space may be introduced in-between the continuity of even a single obstacle.[7] This type of structural error may corrupt the result of scan–to–map-scan matching, and therefore caution must be taken so that either maps are correct by default, or erroneously-generated free space be corrected manually.[8] For the implementation of the 2D Fourier-Mellin transform the `imgreg_dft` python modules were used.[9] With regard to its inputs, these were images produced via `GNU Octave` v.4.0.0:[10] real and virtual scans were plotted at standard pixel

[3]The value of the arbitrary outlier/inlier discrimination threshold was set as such after experiments in real conditions with the same robot when Monte Carlo Localisation (MCL) was utilised for pose tracking. The methodology was the following: first the provided initial estimate of the robot's position was set to a given offset from its true position. Then the robot was tasked with navigating autonomously to a set position on the map using MCL's pose estimate as the robot's pose. If the robot did not manage to arrive at the set position, the offset was lowered until it repeatedly did

[4]https://www.gazebosim.org/

[5]https://www.ros.org/

[6]http://wiki.ros.org/slam_karto

[7]Imperfections of this type are due to the discrete nature of the range sensor with regard to angular sampling, the maps' requested resolution, or the algorithm's execution time compared to the volume of input information

[8]Consider for instance the case where a wall separates one room from another. In the case of an occupancy grid map the introduction of free space as a "crack" in the wall would result in a state where, if a virtual scan is taken at a distance to the wall that is within the range sensor's maximum range, at least one ray may seep through that crack and report an erroneous range measurement. This behaviour may be exacerbated if that wall separates the mapped region with the unmapped region and the intersection-finding routine is not designed to handle unmarked space

[9]https://pypi.org/project/imreg_dft/

[10]https://www.gnu.org/software/octave/

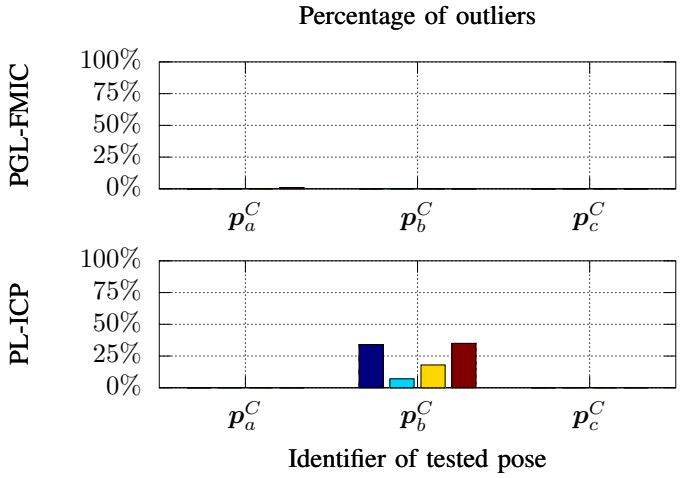



Fig. 16: The distribution of outliers with regard to environment CORRIDOR per sensor noise level tested

size $N_G \times N_G$ and then written to disc as `.png` images. Images of formats `.gif`, `.jpg`, and `.tiff` did not produce adequate results. A delay between producing an image and processing a pose hypothesis was needed in order for images to be stored to disc; execution times of PGL-FMIC in subsection VI-C reflect the algorithm's net execution times. The proposed method's computation cost was 483MB (including `Octave`) in memory and approximately $25\%$ usage of one CPU core.

The following sections present the results of simulations conducted with the proposed method (PGL-FMIC) and a state-of-the-art algorithm (PL-ICP) for solving the same problem of global localisation under motionlessness in the above five environments over the conditions described.

### A. Results with regard to location and orientation error

Figure 14 depicts the mean solution location errors of PGL-FMIC and PL-ICP for solutions whose location error did not exceed the outlier classification error. Figure 15 depicts the corresponding mean solution orientation errors. The results in each environment were recorded over $N_s = 100$ simulations per pose (table II) and per sensor noise level tested.

### B. Results with regard to outlier solutions

Figures 16-20 depict the corresponding percentage of solutions over $N_s$ simulations outputted by PGL-FMIC and PL-ICP that are considered to be outliers. Dark blue denotes the percentage of outliers when $d = 0.0$ m, light blue when $d = 0.01$ m, yellow when $d = 0.02$ m, and brown when $d = 0.05$ m.

### C. Results with regard to execution time

Figure 21 depicts the corresponding mean execution times of PGL-FMIC and PL-ICP.

## VII. Discussion

Examining figures 14-21 overall, what is discerned at first glance is that the location and orientation errors of PGL-FMIC were uncorrelated to the amount of noise disturbing the ranges of the physical LIDAR sensor. Those of PL-ICP were proportional or invariant to it, owing to its exceptional performance in finding correspondences between rays whenever possible. As for their magnitudes, there is no clear pattern of dominance of one algorithm over the other. For PGL-FMIC the maximum inlier location error was less than $0.30$ m over all tested configurations with regard to structured environments, and its execution time was roughly $50\%$ more than that of PL-ICP.

Examining the results more closely reveals that PGL-FMIC's responses included a lower number of solution outliers than those of PL-ICP. In environment CORRIDOR the former produced one outlier over $3 \times 4 \times 100 = 1200$ simulations. PL-ICP failed approximately once every four attempts to solve the problem when the robot was posed at $\boldsymbol{p}_b^C$, being unable to differentiate between the two near-identical bifurcations in the environment.

In environment HOME both algorithms failed to localise the robot when its true pose was $\boldsymbol{p}_j^H$ due to the repetition of its surroundings in the environment and the map, and therefore to the ambiguity of the robot's surroundings. PL-ICP failed to localise the robot for an additional two poses ($\boldsymbol{p}_h^H$ and $\boldsymbol{p}_i^H$), and overall exhibited more outliers than PGL-FMIC with regard to all eleven tested poses in this environment. The above lead to the expected conclusion that, whenever and if at all possible, it is better to place the robot at locations whose surroundings are non-repeating in the entirety of the map.[11]

In environment WAREHOUSE the inadequacy of maximum range by the robot's LIDAR sensor with respect to the distances between objects and their sparsity resulted in PL-ICP being unable to localise the robot over all but one tested poses. In comparison PGL-FMIC was less affected by the effect of missing information, being unable to localise the robot when its pose was $\boldsymbol{p}_e^W$. This leads to the conclusion that, at least as

[11]Placing a disambiguating object in the environment before constructing its map would at least increase the probability of successful resolution of pose ambiguity in environments exhibiting repetitive structures

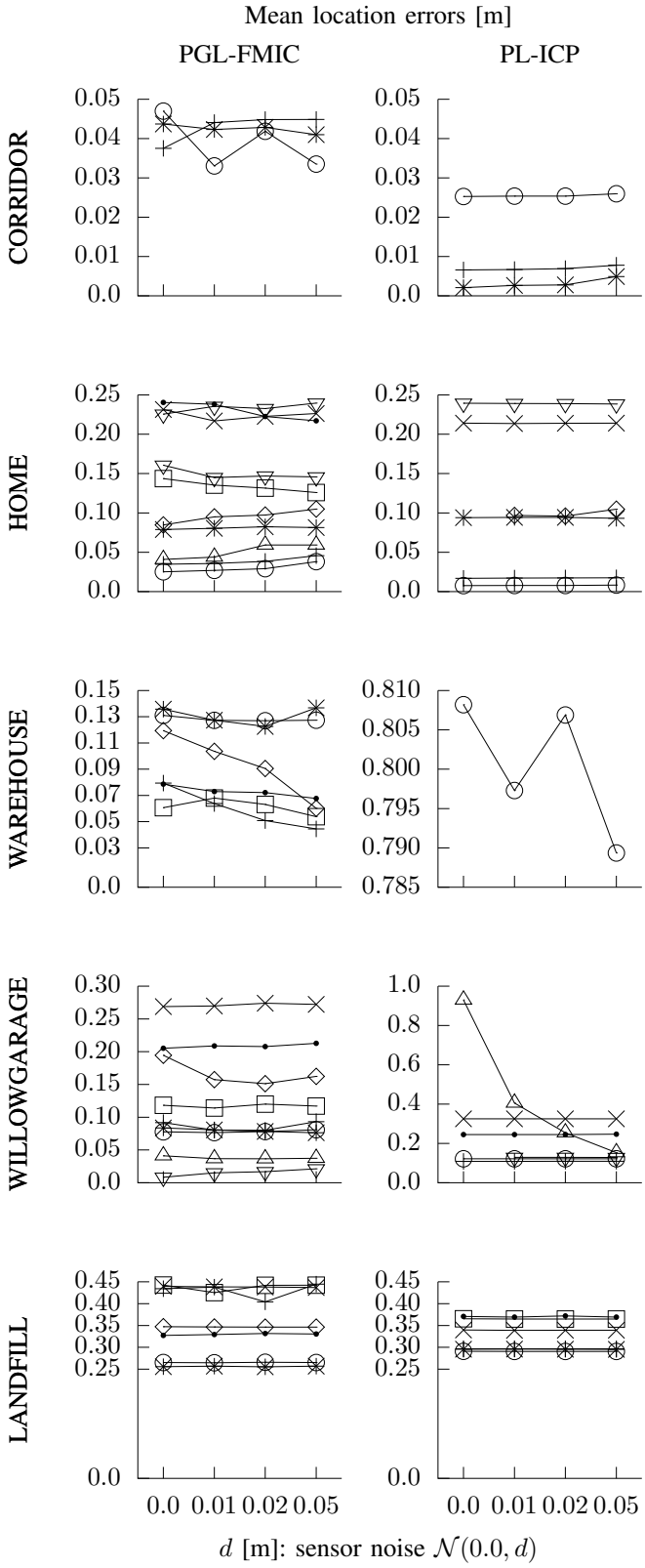



Fig. 14: Mean location error per tested method and sensor noise level with regard to each environment

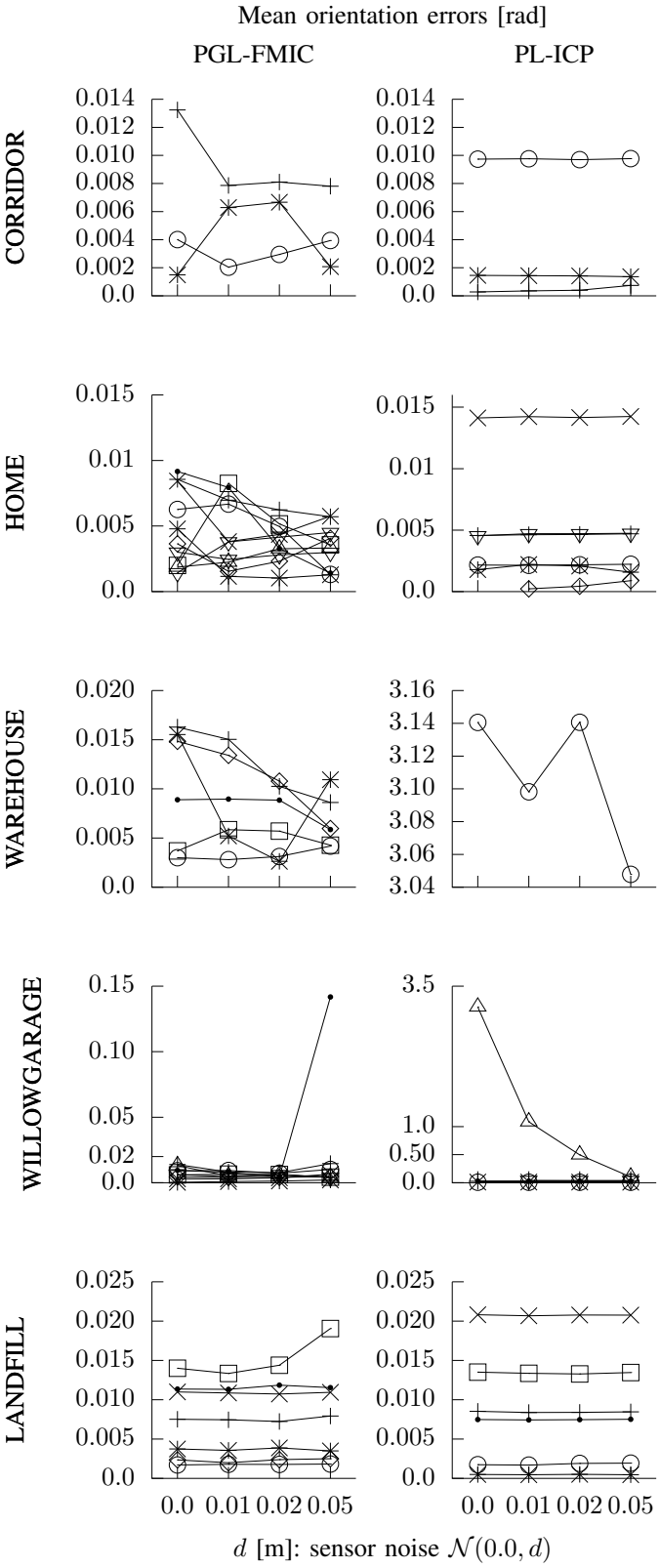



Fig. 15: Mean orientation error per tested method and sensor noise level with regard to each environment

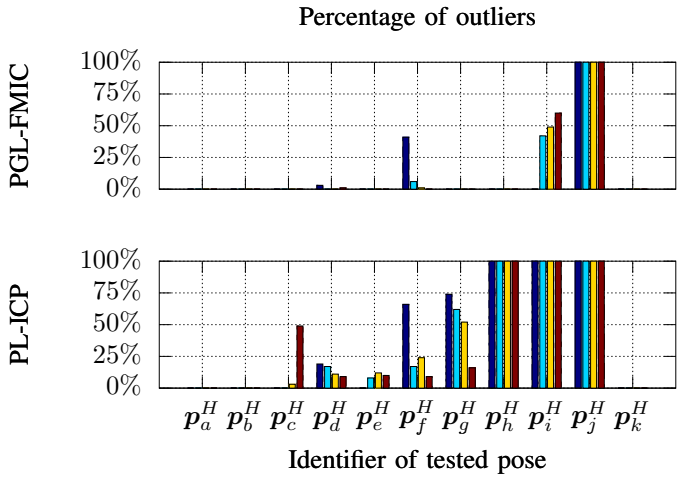



Fig. 17: The distribution of outliers with regard to environment HOME per sensor noise level tested

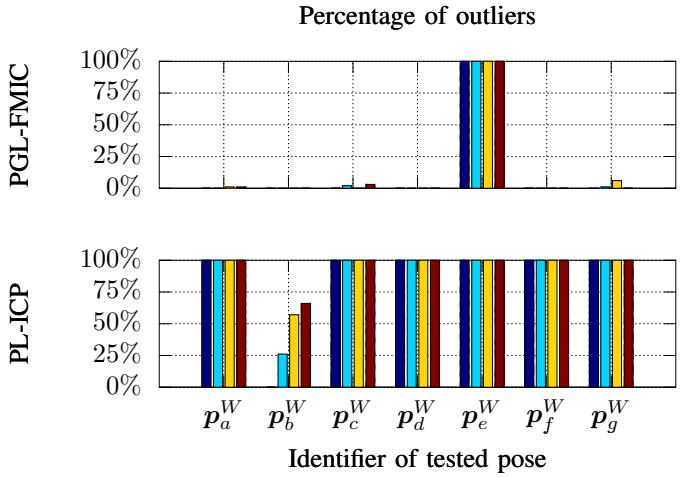



Fig. 18: The distribution of outliers with regard to environment WAREHOUSE per sensor noise level tested

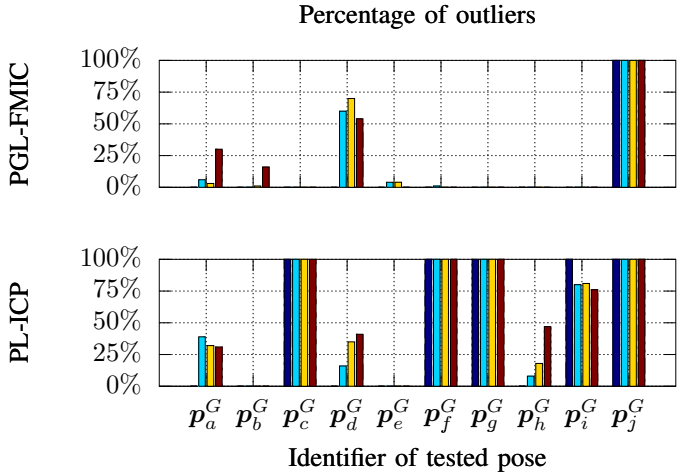



Fig. 19: The distribution of outliers with regard to environment WILLOWGARAGE per sensor noise level tested

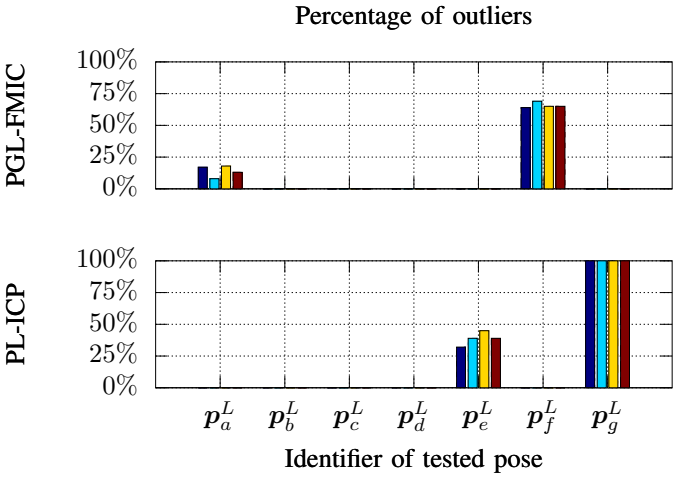



Fig. 20: The distribution of outliers with regard to environment LANDFILL per sensor noise level tested

regards PGL-FMIC, the robot should be placed at a location that maximises the number of range measurements that report values other than that of the LIDAR's maximum range.

In environment WILLOWGARAGE PL-ICP was unable to localise the robot for four out of ten poses tested. For those four poses only a limited number of rays carried missing information. Perhaps this effect, coupled with the conjecture that these must be regarded as outliers by PL-ICP—and therefore that parameters relating to rejection of outliers should be tuned accordingly—led to its inadequacy in this setting. Accordingly, if the same rationale is applied and followed, the same could be said for simulations in environment WAREHOUSE; however our goal here is to avoid tuning parameters in an ad hoc fashion or in a per location or environment manner altogether, as explained in section IV. On the contrary PGL-FMIC was unable to localise the robot only in one instance ($\boldsymbol{p}_j^G$), for which both algorithms could not resolve the eight-way ambiguity of the true pose's surroundings. Curiously though, PGL-FMIC managed to resolve the same ambiguity for poses $\boldsymbol{p}_h^G$ and $\boldsymbol{p}_i^G$, while PL-ICP surpassed itself significantly for the former but marginally for the latter. Overall PGL-FMIC managed to produce correct solutions for all cases and resolve all pose ambiguities arising from repetition of surroundings, except for the cases of $\boldsymbol{p}_j^G$ and $\boldsymbol{p}_d^G$. For the latter almost $50\%$ of all solutions were false and PGL-FMIC exhibited more outliers than PL-ICP. Additional ambiguity resolution was facilitated by decreasing the upper scale threshold $\overline{\sigma}$ from $1.2$ to $1.1$ in the cases where the robot was placed in ambiguous surroundings. PGL-FMIC produced a limited number of outlier solutions when the standard distribution of the noise acting on the real scan's rays was set to its highest value. Given the results for this environment, where the resolution of its map is cruder than in the three cases that precede it, there is no evidence to support that varying resolution deteriorates (or enhances) the outcome of either method tested.

In environment LANDFILL PGL-FMIC exhibited fewer outliers than PL-ICP, with different poses challenging each

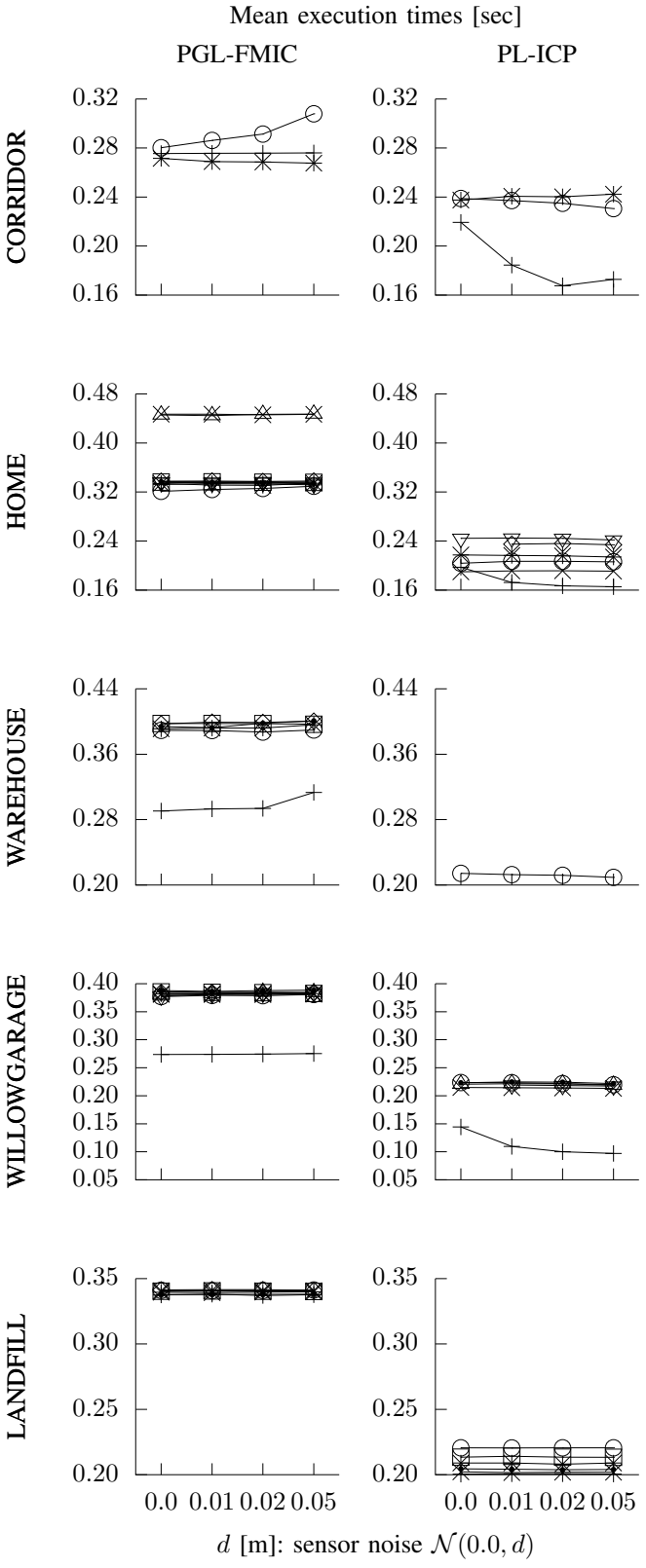



Fig. 21: Mean execution times per tested method and sensor noise level with regard to each environment

algorithm: PL-ICP was unable to globally localise one robot pose ($\boldsymbol{p}_g^L$), while PGL-FMIC successfully localised the robot approximately 3 out of 10 times when it was placed at $\boldsymbol{p}_f^L$. The unstructured nature of environment LANDFILL affected PGL-FMIC's performance more than the other (structured) environments, and most in terms of its positional response rather than its orientation errors.

Overall, with respect to criteria on pose solutions, PGL-FMIC exhibited significant difference in the number of pose solutions that would be admissible in the subsequent task of pose tracking compared to PL-ICP. PGL-FMIC's failure with respect to admissible solutions was due to (a) failure of resolution of identical or similar surroundings, (b) missing range information to an excessive degree, and (c) excessive range noise. In terms of the accuracy of admissible pose solutions, no significant difference in pose accuracy is observed compared to PL-ICP. The final criterion of global pose determination based on the similarity degree outputted by FMI-SPOMF and the scale thresholds used (subsection V-D) were, within reason, well-behaved universal distinctors between hypotheses.

## VIII. Conclusions & future steps

This article introduced a method for solving the problem of motionless passive global localisation in the context of robot localisation on a plane using solely a 2D LIDAR sensor. The proposed method's contribution to the state of the art is that it approaches the solution to the problem without using features or other ad-hoc mediators for estabilishing correspondences between measurements from the environment and its map. Thereby it is suitable for application in structured as well as unstructured environments. By the same virtue it does not require tuning of internal parameters, which bodes well for portability across different and disparate environments, and different poses in the same environment. This exceptional ability is facilitated by the employment of the closed-form Fourier-Mellin transform. Simulations showed that, compared to a non-feature-finding state of the art method, the proposed method achieves a greater degree of global pose discoverability, and that it can accommodate LIDAR sensors of short maximum range compared to the area of a robot's unoccupied surroundings. However, its main limitation is in terms of the physical sensor's field of view, where that must be $360^\circ$.

Future work will focus on the reduction of the execution time, investigating more sophisticated and accurate methods for estimating the position of the global pose, and research on whether more distinctive measures can be extracted from FMI, so that pose ambiguities can be further reduced.

## IX. Acknowledgements

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH CREATE INNOVATE (project code: T2EDK-02000).

## REFERENCES


[1] P. Jensfelt and S. Kristensen, "Active global localization for a mobile robot using multiple hypothesis tracking", in IEEE Transactions on Robotics and Automation, vol. 17, no. 5, pp. 748-760, Oct. 2001. doi: 10.1109/70.964673

[2] O'Kane, J.M., "Global localization using odometry. Proceedings - IEEE International Conference on Robotics and Automation". 2006. 37 - 42. 10.1109/ROBOT.2006.1641158.

[3] A. Gasparri, S. Panzieri, F. Pascucci and G. Ulivi, "A Hybrid Active Global Localisation Algorithm for Mobile Robots", Proceedings 2007 IEEE International Conference on Robotics and Automation, Roma, 2007, pp. 3148-3153. doi: 10.1109/ROBOT.2007.363958

[4] Mark Manasse, Lyle McGeoch, and Daniel Sleator. 1988. "Competitive algorithms for on-line problems". In Proceedings of the twentieth annual ACM symposium on Theory of computing (STOC 88). Association for Computing Machinery, New York, NY, USA, 322333. DOI:https://doi.org/10.1145/62212.62243

[5] J. M. Kleinberg, "The localization problem for mobile robots," Proceedings 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 1994, pp. 521-531. doi: 10.1109/SFCS.1994.365739

[6] Romanik, K., & Schuierer, S. (1996). "Optimal robot localization in trees". SCG '96.

[7] Gregory Dudek, Kathleen Romanik, and Sue Whitesides. "Localizing a Robot with Minimum Travel". SIAM J. Comput. 27, 2 (April 1998), 583604. DOI:https://doi.org/10.1137/S0097539794279201

[8] J. M. O'Kane and S. M. LaValle, "Almost-Sensorless Localization", Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005, pp. 3764-3769. doi: 10.1109/ROBOT.2005.1570694

[9] Rao, M., Dudek, G., & Whitesides, S. (2007). Randomized Algorithms for Minimum Distance Localization. The International Journal of Robotics Research, 26(9), 917933. https://doi.org/10.1177/0278364907081234

[10] S. Se, D. Lowe and J. Little, "Local and global localization for mobile robots using visual landmarks", Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180), Maui, HI, USA, 2001, pp. 414-420 vol.1. doi: 10.1109/IROS.2001.973392

[11] P.V.C Hough, "Method and means for recognizing complex patterns", 1962, United States Patent 3069654

[12] S. F. Hernandez-Alamilla and E. F. Morales, "Global Localization of Mobile Robots for Indoor Environments Using Natural Landmarks", 2006 IEEE Conference on Robotics, Automation and Mechatronics, Bangkok, 2006, pp. 1-6. doi: 10.1109/RAMECH.2006.252692

[13] Michael Bosse, Robert Zlot, "Keypoint design and evaluation for place recognition in 2D lidar maps", Robotics and Autonomous Systems, Volume 57, Issue 12, 2009, pp. 1211-1224, ISSN 0921-8890, https://doi.org/10.1016/j.robot.2009.07.009.

[14] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, volume 14, number 2, pp. 239-256, doi 10.1109/34.121791, ISSN 0162-8828

[15] Brenner, Claus, "Vehicle localization using landmarks obtained by a lidar mobile mapping system", International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences: [PCV 2010 - Photogrammetric Computer Vision And Image Analysis, Pt I] 38 (2010), Nr. Part 3A, S. 139-144

[16] Zhu, J., Zheng, N., & Yuan, Z. (2011). "An Improved Technique for Robot Global Localization in Indoor Environments. International Journal of Advanced Robotic Systems". https://doi.org/10.5772/10525

[17] Censi, A., Iocchi, L., and Grisetti, G., "Scan Matching in the Hough Domain". Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005, pp. 2739-2744.

[18] J. Xie, F. Nashashibi, M. Parent and O. G. Favrot, "A real-time robust global localization for autonomous mobile robots in large environments", 2010 11th International Conference on Control Automation Robotics & Vision, Singapore, 2010, pp. 1397-1402. doi: 10.1109/ICARCV.2010.5707329

[19] M. Bosse and R. Zlot, "Place recognition using keypoint voting in large 3D lidar datasets", 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, 2013, pp. 2677-2684. doi: 10.1109/ICRA.2013.6630945

[20] Park, S., Park, S. "Global localization for mobile robots using reference scan matching". Int. J. Control Autom. Syst. 12, 156168 (2014). https://doi.org/10.1007/s12555-012-9223-0

[21] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints", Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, 2005, pp. 1482-1489 Vol. 2. doi: 10.1109/ICCV.2005.20

[22] M. Himstedt, J. Frost, S. Hellbach, H. Bhme and E. Maehle, "Large scale place recognition in 2D LIDAR scans using Geometrical Landmark Relations", 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, 2014, pp. 5030-5035. doi: 10.1109/IROS.2014.6943277

[23] Gian Diego Tipaldi and K. O. Arras, "FLIRT - Interest regions for 2D range data", 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, 2010, pp. 3616-3622. doi: 10.1109/ROBOT.2010.5509864

[24] L. J. Lyrio, T. Oliveira-Santos, A. Forechi, L. Veronese, C. Badue and A. F. De Souza, "Image-based global localization using VG-RAM Weightless Neural Networks", 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, 2014, pp. 3363-3370. doi: 10.1109/IJCNN.2014.6889888

[25] F. Kallasi and D. L. Rizzini, "Efficient loop closure based on FALKO lidar features for online robot localization and mapping", 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, 2016, pp. 1206-1213. doi: 10.1109/IROS.2016.7759202

[26] Z. Su, X. Zhou, T. Cheng, H. Zhang, B. Xu and W. Chen, "Global localization of a mobile robot using lidar and visual features", 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau, 2017, pp. 2377-2383.

[27] Chen, Y., Chen, W., Zhu, L., Su, Z., Zhou, X., Guan, Y., & Liu, G. (2019). "A Study of Sensor-Fusion Mechanism for Mobile Robot Global Localization". Robotica, 37(11), 1835-1849. doi:10.1017/S0263574719000298 doi: 10.1109/ROBIO.2017.8324775

[28] Azzi, C. (2015). Efficient Image-Based Localization Using Context. Master's thesis, University of Waterloo

[29] Singh, G. and Kosecka, J. (2010). Visual Loop Closing using Gist Descriptors in Manhattan World, in ICRA Omnidirectional Vision Workshop

[30] K. P. Cop, P. V. K. Borges and R. Dub, "Delight: An Efficient Descriptor for Global Localisation Using LiDAR Intensities", 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, 2018, pp. 3653-3660. doi: 10.1109/ICRA.2018.8460940

[31] Wang, X., Marcotte, R.J., & Olson, E. "GLFP: Global Localization from a Floor Plan". 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1627-1632.

[32] Olson, E., & Agarwal, P. (2013). "Inference on networks of mixtures for robust robot mapping", The International Journal of Robotics Research, 32(7), 826840. https://doi.org/10.1177/0278364913479413

[33] Abdurrahman Yilmaz, Hakan Temeltas, "Self-adaptive Monte Carlo method for indoor localization of smart AGVs using LIDAR data", Robotics and Autonomous Systems, Volume 122, 2019, 103285, ISSN 0921-8890, https://doi.org/10.1016/j.robot.2019.103285.

[34] Zhang, Lei. (2010). "Self-adaptive Markov Localization for Single-Robot and Multi-Robot Systems", Ph.D thesis, Universite Montpellier II-Sciences et Techniques du Languedoc

[35] L. Zhang, R. Zapata and P. Lpinay, "Self-adaptive Monte Carlo localization for mobile robots using range sensors", 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, 2009, pp. 1541-1546. doi: 10.1109/IROS.2009.5354298

[36] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. "Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters", IEEE Transactions on Robotics, Volume 23, pages 34-46, 2007

[37] W. Hess, D. Kohler, H. Rapp and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, 2016, pp. 1271-1278, doi: 10.1109/ICRA.2016.7487258

[38] G. Bresson, Z. Alsayed, L. Yu and S. Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," in IEEE Transactions on Intelligent Vehicles, vol. 2, no. 3, pp. 194-220, Sept. 2017, doi: 10.1109/TIV.2017.2749181

[39] Labb, M, Michaud, F. "RTABMap as an opensource lidar and visual simultaneous localization and mapping library for largescale and longterm online operation". J Field Robotics. 2019; 35: 416 446. https://doi.org/10.1002/rob.21831

[40] Cooper, M.A.; Raquet, J.F.; Patton, R. “Range Information Characterization of the Hokuyo UST-20LX LIDAR Sensor”. Photonics 2018, 5, 12.

[41] Sebastian Thrun, Wolfram Burgard, and Dieter Fox, “Probabilistic Robotics” (Intelligent Robotics and Autonomous Agents), The MIT Press, 2005

[42] F. Dellaert, D. Fox, W. Burgard and S. Thrun, “Monte Carlo localization for mobile robots,” Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C), Detroit, MI, USA, 1999, pp. 1322-1328 Volume 2, doi: 10.1109/ROBOT.1999.772544

[43] P. Maybeck, “Stochastic Models, Estimation and Control”, Volume 1, Academic Press, New York, 1979

[44] Rsmann, Christoph, Wendelin Feiten, Thomas Wsch, Frank Hoffmann, and Torsten Bertram. ”Trajectory modification considering dynamic constraints of autonomous robots.” In Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on, pp. 1-6. VDE, 2012.

[45] C. Rsmann, F. Hoffmann and T. Bertram, ”Kinodynamic trajectory optimization and control for car-like robots,” 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, 2017, pp. 5681-5686, doi: 10.1109/IROS.2017.8206458

[46] Vasiljevic, G., Miklic, D., Draganjac, I., Kovacic, Z., and Lista, P. “High-accuracy vehicle localization for autonomous warehousing”, 2016

[47] Anastasios Tzitzis, Spyros Megalou, Stavroula Siachalou, Traianos Yioultsis, Athanasios Kehagias, Emmanouil Tsardoulias, Alexandros Filotheou, Andreas Symeonidis, Loukas Petrou, Antonis G. Dimitriou, ”Phase ReLock - Localization of RFID Tags by a Moving Robot,” European Conference on Antennas and Propagation, Krakow, Poland, April 2019.

[48] Spyros Megalou, Anastasios Tzitzis, Stavroula Siachalou, Traianos Yioultsis, John Sahalos, Emmanouil Tsardoulias, Alexandros Filotheou, Andreas Symeonidis, Loukas Petrou, Aggelos Bletsas, Antonis G. Dimitriou, ”Fingerprinting Localization of RFID tags with Real-Time Performance-Assessment, using a Moving Robot,” European Conference on Antennas and Propagation, Krakow, Poland, April 2019.

[49] A. Censi, “An ICP variant using a point-to-line metric,” 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, 2008, pp. 19-25, doi: 10.1109/ROBOT.2008.4543181

[50] J. -. Gutmann and K. Konolige, “Incremental mapping of large cyclic environments” Proceedings 1999 IEEE International Symposium on Computational Intelligence in Robotics and Automation. CIRA’99 (Cat. No.99EX375), Monterey, CA, USA, 1999, pp. 318-325. doi: 10.1109/CIRA.1999.810068

[51] D. Hahnel, W. Burgard, D. Fox and S. Thrun, “An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements,” Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453), Las Vegas, NV, USA, 2003, pp. 206-211 vol.1. doi: 10.1109/IROS.2003.1250629

[52] Chieh-Chih Wang, C. Thorpe and S. Thrun, “Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas,” 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422), Taipei, Taiwan, 2003, pp. 842-849 vol.1. doi: 10.1109/ROBOT.2003.1241698

[53] Lacroix, S., Mallet, A., Bonnafous, D., Bauzil, G., Fleury, S., Herrb, M., and Chatila, R. (2002). “Autonomous Rover Navigation on Unknown Terrains: Functions and Integration”. The International Journal of Robotics Research, 21(1011), 917942. https://doi.org/10.1177/027836490202101084l

[54] Minguez, J., Montesano, L., anc Montano, L. (2004). “An architecture for sensor-based navigation in realistic dynamic and troublesome scenarios”. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), 3, 2750-2756 vol.3.

[55] Montesano, Luis, Minguez, Javier and Montano, Luis, “Modeling dynamic scenarios for local sensor-based motion planning”, Autonomous Robots, 2008, Volume 25, pp 231-251.

[56] D. Schulz, W. Burgard, D. Fox and A. B. Cremers, “Tracking multiple moving targets with a mobile robot using particle filters and statistical data association”, Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164), Seoul, South Korea, 2001, pp. 1665-1670 vol.2. doi: 10.1109/ROBOT.2001.932850

[57] Casasent, David and Psaltis, Demetri, “Position, rotation, and scale invariant optical correlation”, Applied Optics, pp. 1795-9, 1976

[58] Qin-Sheng Chen, M. Defrise and F. Deconinck, “Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition”, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 12, pp. 1156-1168, Dec. 1994. doi: 10.1109/34.387491

[59] B. S. Reddy and B. N. Chatterji, “An FFT-based technique for translation, rotation, and scale-invariant image registration”, in IEEE Transactions on Image Processing, vol. 5, no. 8, pp. 1266-1271, Aug. 1996. doi: 10.1109/83.506761

[60] Bernhard Riemann, “Grundlagen fr eine allgemeine Theorie der Functionen einer vernderlichen complexen Grsse”, Doctoral Thesis, 1851

[61] Checchin, Paul, Gérossier, Franck, Blanc, Christophe, Chapuis, Roland, Trassoudaine, Laurent, “Radar Scan Matching SLAM Using the Fourier-Mellin Transform”, “Field and Service Robotics”, 2010, pp. 151–161, 978-3-642-13408-1

[62] Vivet, D., Grossier, F., Checchin, P., Trassoudaine, L., & Chapuis, R. (2013). “Mobile Ground-Based Radar Sensor for Localization and Mapping: An Evaluation of two Approaches”. International Journal of Advanced Robotic Systems. https://doi.org/10.5772/56636

[63] Monod, M.O., “Frequency modulated radar: a new sensor for natural environment and mbile robotics”, Ph.D. Thesis, Paris IV University, France, 1995

[64] Heiko Blow, Max Pfingsthorn, Andreas Birk, “Using Robust Spectral Registration for Scan Matching of Sonar Range Data”, IFAC Proceedings Volumes, Volume 43, Issue 16, 2010, pp. 611-616, ISSN 1474-6670, ISBN 9783902661876, https://doi.org/10.3182/20100906-3-IT-2019.00105.

[65] blow, h., birk, a. “Spectral registration of noisy sonar data for underwater 3D mapping”. Autonomous Robots 30, 307331 (2011). https://doi.org/10.1007/s10514-011-9221-8

[66] M. Pfingsthorn, A. Birk, S. Schwertfeger, H. Blow and K. Pathak, “Maximum likelihood mapping with spectral image registration”, 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, 2010, pp. 4282-4287. doi: 10.1109/ROBOT.2010.5509366

[67] H. Blow and A. Birk, “Fast and robust photomapping with an Unmanned Aerial Vehicle (UAV)”, 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, 2009, pp. 3368-3373. doi: 10.1109/IROS.2009.5354505

[68] A. Birk, “Using recursive spectral registrations to determine brokenness as measure of structural map errors”, 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, 2010, pp. 3472-3477. doi: 10.1109/ROBOT.2010.5509322

[69] T. Kazik and A. H. Gktoan, “Visual odometry based on the Fourier-Mellin transform for a rover using a monocular ground-facing camera”, 2011 IEEE International Conference on Mechatronics, Istanbul, 2011, pp. 469-474. doi: 10.1109/ICMECH.2011.5971331

[70] N. Hurts, X. Cuf’, Y. Petillot and J. Salvi, “Fourier-based registrations for two-dimensional forward-looking sonar image mosaicing”, 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, 2012, pp. 5298-5305. doi: 10.1109/IROS.2012.6385813

[71] K. Kim, N. Neretti and N. Intrator, “Mosaicing of acoustic camera images”, in IEE Proceedings - Radar, Sonar and Navigation, vol. 152, no. 4, pp. 263-270, 5 Aug. 2005. doi: 10.1049/ip-rsn:20045015

[72] Lowe, D.G. “Distinctive Image Features from Scale-Invariant Keypoints:. International Journal of Computer Vision 60, 91110 (2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94

[73] J. Oberlander, A. Roennau and R. Dillmann, “Hierarchical SLAM using spectral submap matching with opportunities for long-term operation”, 2013 16th International Conference on Advanced Robotics (ICAR), Montevideo, 2013, pp. 1-7. doi: 10.1109/ICAR.2013.6766479

[74] J. Rohde, B. Vlz, H. Mielenz and J. M. Zllner, “Precise vehicle localization in dense urban environments”, 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, 2016, pp. 853-858. doi: 10.1109/ITSC.2016.7795655

[75] Filotheou, A., Tsardoulias, E., Dimitriou, A. et al. “Pose Selection and Feedback Methods in Tandem Combinations of Particle Filters with Scan-Matching for 2D Mobile Robot Localisation”. J Intell Robot Syst (2020). https://doi.org/10.1007/s10846-020-01253-6

[76] https://github.com/AndreaCensi/csm/blob/master/csm_manual.pdf, last accessed 12/03/2020

[77] https://github.com/AndreaCensi/csm/blob/master/sm/csm/algos.h, last accessed 12/03/2020

[78] http://data.nvision2.eecs.yorku.ca/3DGEMS/, last accessed 11/11/2020, based on the preprint A.Rasouli, J.K. Tsotsos. ”The Effect of Color Space Selection on Detectability and Discriminability of Colored Objects.” arXiv preprint arXiv:1702.05421 (2017).

[79] https://github.com/CognitiveRobotics/jarves/tree/master/jarves_gazebo/ worlds/maps, last accessed 30/06/2020