This full text paper was peer-reviewed at the direction of IEEE Instrumentation and Measurement Society prior to the acceptance and publication.

Optimal Landmark Placement for Indoor Positioning using Context Information and Multi-sensor Data

V. Magnago, P. Bevilacqua, L. Palopoli, R. Passerone
Dep. of Information Engineering and Computer Science
University of Trento, Trento, Italy
E–mail: {valerio.magnago,luigi.palopoli,roberto.passerone}@unitn.it

D. Fontanelli, D. Macii
Dep. of Industrial Engineering
University of Trento, Trento, Italy
E–mail: {daniele.fontanelli, david.macii}@unitn.it

***Abstract*—The most effective solutions for indoor positioning of mobile agents typically rely on multi-sensor data fusion. In particular, good trade-offs in terms of accuracy, scalability and availability can be achieved by combining dead reckoning techniques (e.g. based on odometry) and measurements of distance and attitude with respect to suitable landmarks with a known position and/or orientation within a given reference frame. A crucial problem of this kind of techniques is landmark deployment, which should keep into account not only the limited detection range of the adopted sensors and the non-null probability of missing a landmark, even if it actually lies within the sensor detection area (SDA). This paper focuses on minimum landmark placement taking into account possible environment contextual information. This solution relies on a greedy placement algorithm that optimally solves the problem while keeping positioning uncertainty below a given limit. The correctness of the proposed approach is verified through multiple simulations in the context of the EU project ACANTO, which requires to localise one or more smart robotic walkers in large, public and potentially crowded environments such as shopping malls or airports.**

***Keywords*—Indoor localisation, position tracking, landmark placement, optimisation.**

I. Introduction

Indoor positioning is a fundamental enabler for Ambient Assisted Living (AAL) applications, human centred service robotics and customer guidance in public spaces. Indoor localisation and position tracking suffer from some critical problems, which can be shortly summarised below [1]. First, the signals of Global Navigation Satellite Systems (GNSS) are usually too weak to be detected indoors. Second, an accuracy in the order of a few tens of cm or less is required by most applications, e.g., to avoid obstacles detected with limited sensing systems. Third, possible scalability issues arise when a large number of targets (e.g., multiple customers) need to be localised in the same environment at the same time. At the moment, a one-size-fit-all solution for indoor localisation and positioning still does not exist [2]. A good trade-off addressing most of the issues listed above can be achieved through multi-sensor data fusion algorithms [3], [4], [5], particularly those combining ego-motion relative (e.g., dead reckoning) techniques with distance and heading values measured with respect to "anchor nodes", "tags", "markers" or "landmarks" having known coordinates in a given "absolute" reference frame. In this respect, a crucial problem is where to place such landmarks in the environment considered.

A key requirement for indoor localisation is to keep positioning uncertainty below given target boundaries [6]. Of course, the minimum uncertainty is certainly achieved when a landmark is detected at any time [7]. However, since the sensors (e.g. cameras) detecting the landmarks have always a limited detection area, the data about absolute position and orientation are intrinsically intermittent. In such conditions a robot can track its own position just using dead reckoning. Of course, in this case positioning uncertainty grows until a new landmark is detected. Therefore, landmarks are needed to keep uncertainty bounded, but their number should be properly minimised. This is a subclass of the landmark selection problem addressed in robotics using online [8], [9] or offline [10] approaches. As pointed out in [11], the offline approach corresponds to the landmark deployment problem considered in this paper. Possible solutions to this problem rely on heuristic common-sense criteria (e.g. using strips of RFID tags so that at least one of them is within the detection range of the on-board reader [12]) or on cost function to minimise. The solutions that do not require any knowledge of the target trajectories give effective guarantees, but may be excessively conservative in real scenarios [10], [7]. On the other hand, the a-priori knowledge of possible trajectories may be stochastic [13] or deterministic [14], like in the case considered in this paper. Unfortunately, taking into consideration the accuracy and the detection area of the sensors employed, the trajectories of the target to be tracked and the geometry of the environment make the problem NP-complete [11]. Moreover, if the target uncertainty is included in the problem [4], the solution becomes even more challenging.

This paper extends the nearly-optimal greedy landmark placement strategy proposed in [15] (e.g. assuming to deploy Aruco codes on the floor [16]) by considering additional features that can be directly detected, such as the presence of walls, thus further reducing the need for artificial landmarks. Moreover, we explicitly consider a non-negligible probability of missed landmark detection, turning the deterministic target uncertainty boundary in a stochastic limit. To the best of our knowledge, this is the first solution that explicitly considers the possibility to rely on context information in order to deploy landmarks when necessary.

The rest of the paper is structured as follows. Section II provides an overview of system and measurement models and formalises the optimisation problem. Section III is focused on the landmark placement algorithm. Section IV reports some meaningful simulation results. Finally, Section V concludes the paper and outlines future work.

978-1-5386-2222-3/18/$31.00 ©2018 IEEE

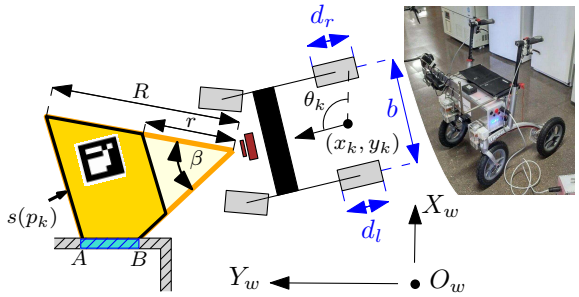

Fig. 1. The *FriWalk* schematic representation and SDA.

II. Models and Problem Formulation

This section presents the reference model adopted for localisation, the available measurements, the metric adopted for the uncertainty and the formalisation of the optimal problem. Even though the presented approach can be applied to a generic platform moving indoors on a horizontal plane using ego-motion data for dead reckoning (e.g. odometers or inertial measurement units), measurements of environmental quantities (e.g. cameras detecting obstacles) as well as landmark detection (e.g. Quick Response (QR) codes, Aruco codes or RFID tags) [4], [16], without loss of generality we will make explicit reference to the *FriWalk* (Fig. 1), a service robot developed in the European project ACANTO [17], which is provided with cognitive [18], [19] and guidance functionalities [20], [21], [22].

A. Platform Model

The *FriWalk* is based on a standard commercial Trionic Walker 12er. The *FriWalk* is modelled as a unicycle-like vehicle [18]. If t_s denotes the sampling period of the onboard sensors, the position of the walker at time kt_s is represented by the coordinates (x_k, y_k) of the mid-point of the rear wheels axle expressed in the fixed right-handed reference frame $\langle W \rangle = \{O_w, X_w, Y_w, Z_w\}$ (Fig. 1). The orientation of the *FriWalk* is instead given by the angle θ_k between the forward vehicle direction and the axis X_w. The robot state can be define as $p = [x, y, \theta, v, \omega]^T$ where $[x, y, \theta]^T$ are the generalised coordinates of the system, v is the forward velocity and ω is the angular speed. The discrete-time kinematic model of the *FriWalk* is then given by:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \\ v_{k+1} \\ \omega_{k+1} \end{bmatrix} = \begin{bmatrix} x_k + v_k \cos(\theta_k) t_s - \frac{v_k \sin(\theta_k) \omega_k t_s^2}{2} \\ y_k + v_k \sin(\theta_k) t_s + \frac{v_k \sin(\theta_k) \omega_k t_s^2}{2} \\ \theta_k + \omega_k t_s \\ v_k \\ \omega_k \end{bmatrix} + g(p_k)\epsilon_k, \tag{1}$$

which is a constant velocity model generating a random walk if the input vector $\epsilon_k = [\epsilon_{v_k}, \epsilon_{\omega_k}]$ modelling the unknown time-varying accelerations is a random noise. In such a case, the noise input matrix $g(p_k)$ is given by:

$$g(p_k) = \begin{bmatrix} \frac{t_s^2}{2}\cos(\theta_k) & \frac{t_s^2}{2}\sin(\theta_k) & 0 & t_s & 0 \\ 0 & 0 & \frac{t_s^2}{2} & 0 & t_s \end{bmatrix}^T$$

The acceleration noises are supposed to be distributed according to a stationary zero-mean white Gaussian process with a 2×2 diagonal covariance matrix E whose diagonal elements are σ_a^2 and σ_α^2.

From (1), the overall system model can be more compactly rewritten as

$$\begin{cases} p_{k+1} = f(p_k) + g(p_k)\epsilon_k \\ z_k = h(p_k) + \eta_k \end{cases} \tag{2}$$

where $h(p_k)$ is the nonlinear output function representing the available sensors and η_k is the vector of the uncertainties affecting the sensor measurement data available at time kt_s.

B. Measurements

The *FriWalk* is equipped with heterogeneous sensors that are: a) a pair of encoders installed on the rear wheels for dead reckoning; b) an embedded vision system able to detect Aruco codes placed on the floor and associated with known coordinates in $\langle W \rangle$ and to measure the relative pose between the walker and the detected Aruco code; c) a frontal RGB-D camera (namely a Astra Orbbec[1]) to detect environmental features, such as walls.

Encoders: The encoders measure the angular displacements $z_k^e = [\delta_r^k, \delta_l^k]^T$ of the right and left wheel, respectively, between the $(k-1)t_s$ and kt_s time instants. As a consequence, the right (or left) wheel linear displacements in one sampling period are given by $\frac{d_R}{2}\delta_k^r$ (or $\frac{d_l}{2}\delta_k^l$), where d_r and d_l are the wheel diameters. Due to the linear relationship between the vehicle velocities v_k and ω_k and the wheels angular velocities, we have

$$z_k^e = \begin{bmatrix} 0 & 0 & 0 & \frac{2t_s}{d_r} & \frac{bt_s}{d_r} \\ 0 & 0 & 0 & \frac{2t_s}{d_l} & -\frac{bt_s}{d_l} \end{bmatrix} p_k + \eta_k^e,$$

where b is the vehicle interaxle length (see Figure 1). The encoder measurements are affected by uncertainty η_k^e, which is a bivariate zero-mean normally distributed random vector with covariance matrix $N^e = \mathrm{diag}(\sigma_r^2, \sigma_l^2)$.

Embedded vision system: The sensor detection area (SDA) of the vision system (shortly denoted as $s(p_k)$ for a given position p_k) is assumed to be a trapezoid, as depicted in Fig. 1. The measurement data vector

$$z_k^c = \begin{bmatrix} \mathbf{R}(\theta_c)\left(\mathbf{R}(\theta_k)\begin{bmatrix} x_{q_i} - x_k \\ y_{q_i} - y_k \end{bmatrix} - \begin{bmatrix} x_c \\ y_c \end{bmatrix}\right) \\ \theta_k + \theta_c - \theta_{q_i} \end{bmatrix} + \eta_k^c, \tag{3}$$

where (x_{q_i}, y_{q_i}) and θ_{q_i} are the known position and orientation of the i-th Aruco code in $\langle W \rangle$, respectively, $\mathbf{R}(\theta)$ is the rotation on the plane of angle θ, i.e.

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix},$$

and (x_c, y_c) and θ_c represent the position and orientation of the camera with respect to the reference point of coordinates (x_k, y_k). The measurement data are supposed to be affected by a zero-mean and normally distributed uncertainty vector η_k^c with covariance matrix $N^c = \mathrm{diag}(\sigma_{x_c}^2, \sigma_{y_c}^2, \sigma_{\theta_c}^2)$.

RGB-D camera: The front Astra Orbbec detects and measures the distance from possible fixed obstacles, e.g. walls, when they are within its own SDA, which can be also modelled

[1] https://orbbec3d.com/product-astra/

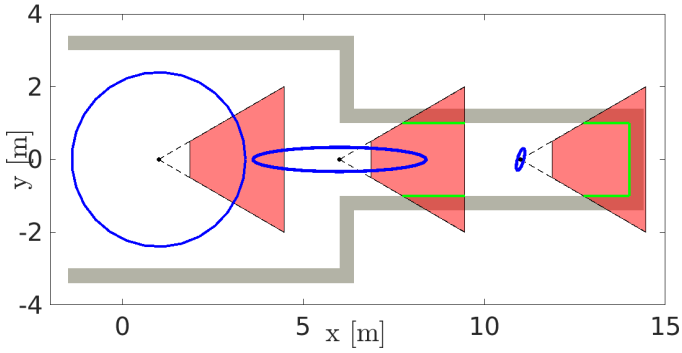

Fig. 2. Example of uncertainty reduction due to the fusion of encoder and RGB-D measurement data. As the amount of information increases, the positioning uncertainty tends to decrease.

as a trapezoid. The map of the indoor environment, including the walls, is assumed to be given. When the Orbbec detects a wall, a segment parallel to the plane of motion and delimited by points A and B can be measured, as shown in Figure 1. If (x_a, y_a) and (x_b, y_b) are the coordinates of A and B in the reference frame of the Orbbec, the angle and the distance between the Orbbec and point A are given by $\theta_a = \arctan(y_a/x_a)$, and $\rho_a = \sqrt{x_a^2 + y_a^2}$, respectively. As a consequence, the function modelling the measurements performed by the Orbbec is $z_k^r = [\rho_k, \theta_{ab_k}] + \eta_k^r$, where

$$\begin{aligned} \rho_k =& \sqrt{[x_p - x_k - x_r\cos(\theta_k) - y_r\sin(\theta_k)]^2 + \\ & [y_p - y_k - x_r\sin(\theta_k) - y_r\cos(\theta_k)]^2}, \\ \theta_{ab_k} =& \theta_k + \theta_r, \end{aligned} \tag{4}$$

(x_r, y_r) is the position of the Orbbec camera with respect to (x_k, y_k), θ_r is orientation of the Orbbec, (x_p, y_p) is one generic point belonging to the detected wall and available in the map $\mathcal{M}$ of the environment, and η_k^r denotes the measurement uncertainty vector. Again, this is a bivariate and normally-distributed random vector, with zero mean and covariance matrix $N^r = \text{diag}(\sigma_{\rho_r}^2, \sigma_{\theta_r}^2)$.

To clarify the role of the Orbbec camera, consider the qualitative straight path shown in Figure 2. Let us assume that the initial standard uncertainty of the *FriWalk* along axes X_w and Y_w is $\sigma_x = \sigma_y = 0.8$ m. If the position of the *FriWalk* were estimated only through dead-reckoning (i.e. using just encoder data), the initial positioning uncertainty would grow unboundedly. If instead the RGB-D camera detects the walls of the corridor, the distance and orientation measurements given by (4) can be used to reduce σ_y, e.g. by using an Extended Kalman Filter (EKF) based on model (2) [4], [16]. Moreover, as the *FriWalk* keeps on moving along the corridor, the localisation uncertainty can be further reduced along both X_w and Y_w, due to the larger amount of measurement data used to update the estimated state of (2).

Of course, the general approach above can be extended by including the measurements given by (3) when an Aruco code is detected. In this case, with reference to (2), a full vector of measurement data $z_k = [z_k^e, z_k^c, z_k^r]^T$ is collected at time kt_s and the corresponding overall measurement uncertainty vector $\eta_k = [\eta_k^e, \eta_k^c, \eta_k^r]^T$ has covariance matrix $N = \text{diag}(N^e, N^c, N^r)$, since the measurements performed by different sensors can be reasonably assumed to be uncorrelated. Of course, the presence of z_k^c and z_k^r in z_k as well as of η_k^c and η_k^r in η_k is inherently intermittent, as it depends on whether an Aruco code and/or a wall is detected at time kt_s.

C. Uncertainty Evaluation

One crucial issue for the formalisation of the placement optimisation problem is the definition of a scalar function able to provide a trustworthy expression of positioning uncertainty. In this paper, such a function is

$$f(P_k) = \max\sqrt{\text{Eig}\,(P_k^{x,y})}, \tag{5}$$

where operator $\text{Eig}(M)$ returns the eigenvalues of matrix M and $P_k^{x,y} \in \mathbb{R}^{2\times 2}$ is the upper diagonal block of the covariance matrix $P_k \in \mathbb{R}^{5\times 5}$ associated with state vector p_k. The rationale underlying the choice of (5) is twofold. First, the elements of the main diagonal of $P_k^{x,y}$ are the squared standard uncertainties associated with the *FriWalk* planar coordinates in $\langle W \rangle$. Second, by selecting the larger eigenvalue of $P_k^{x,y}$, the ellipse representing geometrically the positioning uncertainty around point (x_k, y_k) is approximated by the circumscribing circle. This is a conservative, but reasonable choice for the problem at hand [14], since the uncertainty has to be kept below a given limit. In particular, the condition

$$g(N^c) \leq f(P_k) \leq \xi(p_k), \tag{6}$$

must hold. The lower uncertainty bound $g(N^c)$ is achieved as soon as an Aruco code is detected and depends on the metrological characteristics of the embedded vision system. On the contrary, the upper uncertainty bound $\xi(p_k)$ depends on the specifications of the overall localisation system. Quite intuitively, if $\xi(p_k)$ decreases, then the number of landmarks to be deployed grows.

D. Problem Formulation

Let $\mathcal{Q} \subseteq \mathbb{R}^2 \times [0, 2\pi)$ be the space reachable by the *FriWalk* inside the environment (i.e. $p_k \in \mathcal{Q}\ \forall k$) and

$$\mathcal{D} = \left\{(x, y) \in \mathbb{R}^2 \mid \exists p_k \in \mathcal{Q}, (x, y) \in s(p_k)\right\},$$

be the points lying in the SDA of the vision system from at least one position p_k. If $\mathcal{L}_p \subseteq \mathcal{D}$ denotes the area in which it is possible to place landmarks, then

$$\mathcal{L}_f \cap s(p_k) \neq \emptyset, \forall p_k \in \mathcal{Q}, \tag{7}$$

represents a subset of points of $\mathcal{L}_p$ with a minimal cardinality, where the Aruco codes can be initially placed so that at least one of them is always in the SDA of the vision system [7].

If we assume to know the possible paths of the *FriWalk*, then at least one landmark should be placed along each path to avoid unbounded uncertainty growth due to dead reckoning. This may require a preliminary observation of the target's trajectories in the indoor environment considered. In the case at hand, the set of available trajectories, dubbed $\mathcal{T}$, can be conveniently represented using the set $\mathcal{L}_f$. Indeed, due to (7), $\mathcal{S}_{i,k} \triangleq s(p_k) \cap \mathcal{L}_f \neq \emptyset$ can be regarded as the set of possible locations where the Aruco codes could be detected at time kt_s along the i–th trajectory $T_i \in \mathcal{T}$. Notice that $\mathcal{S}_{i,k}$ may contain more than one element of $\mathcal{L}_f$. This way, the i-th path, even if

continuous, can be approximately represented as a *quantised and finite* sequence of points $\mathcal{S}_{i,k}$. Ultimately, the optimisation problem can be formulated as follows, i.e.

Problem 1: Given $\mathcal{L}_f$, the sets of $\mathcal{S}_i$, the map $\mathcal{M}$ and $\xi(p_k) \geq g(N^c)$, $\forall p_k \in \mathcal{T}$, find:

$$\mathcal{L} = \arg\min_{\mathcal{L}_x} |\mathcal{L}_x| \text{ s.t.}$$
$$\mathcal{L}_x \subseteq \cup_i \mathcal{S}_i \text{ and } f(P_k) \leq \xi(p_k), p_k \in \mathcal{T}, \forall k.$$

Observe that, in the worst-case, $\mathcal{L} = \mathcal{L}_f$ can be a solution to the problem considered.

III. Optimal Landmark Placement

This section first shows how to cast Problem 1 into a binary programming problem [15], assuming to include the measurements based on the RGB-D camera as well. Then, Section III-B describes the greedy placement algorithm. Section III-C considers the probability of missing a visual landmark.

A. Boolean Representation

Each possible Aruco code location $l_i \in \mathcal{L}_f$ can be associated with a Boolean variable a_i, which is set to 1 if the landmark is placed in l_i or 0 otherwise. It follows that the minimum number of landmarks is deployed if the number of boolean variables set to 1 is minimised as well. To determine where Aruco codes must be placed to meet the wanted uncertainty limit $\xi(p_k)$, we start from a generic position $p_q \in T_i$ at time qt_s, assuming that $f(P_q) = g(N^c)$. The motion of the vehicle along each path can be simulated to compute the evolution of the covariance matrix $P_{q+1}, P_{q+2}, \ldots$ along T_i fusing the measurement data from both encoders and vision systems. While the robot motion is simulated along T_i, we keep track of the landmark positions $\mathcal{S}_{i,j}$ in view. If at time $k+1 > q$, $f(P_{k+1}) > \xi(p_{k+1})$, then we have a violation. To avoid it, at least one landmark must be present in one of the positions $\mathcal{S}_{i,j}$ in view. To increase the robustness of the system at the expense of the number of landmarks to be deployed, we can solve this problem adding the constraint that each clause is covered by at least n landmarks. When n is equal to 1 we have the minimum number of landmarks in the environment, but robustness to missing data is undermined. This condition can be expressed as follows:

$$\lambda_{i,q} = a_{i_q} \vee a_{i_{q+1}} \cdots \vee a_{i_k}.$$

This analysis can be repeated for all starting positions and for all trajectories. The corresponding clauses can be collected in a set Λ. For the problem to be solved, it is necessary and sufficient that all the generated clauses evaluate to true. A compact representation of Λ is given by a coverage matrix C whose columns are the possible landmarks locations $l_i \in \mathcal{L}_f$ while the rows are the clauses $\lambda_{i,q}$. A generic matrix entry indexed by (r, c) is set to 1 if the r-th clause is fulfilled by the c-th landmark, or 0 otherwise. An example is shown in Table I.

B. Greedy Placement

As shown in [15], the greedy algorithm for landmark placement leads to a good approximation of the optimal solution within a reasonable computation time. We start with the coverage matrix C, computed as described previously, where the columns are sorted with a decreasing number of elements equal to 1. With reference to Table I, the first column will be l_2, then l_3 and so on. Initially, no landmarks are placed. Since the clauses are not covered by any landmark, the cover counter is set to 0. Afterwards, a landmark is placed in the position corresponding to the first column, i.e., the one satisfying the greatest number of clauses. As a result, the first column is removed. Then, the cover counter of the satisfied clauses (the matrix rows) is incremented by one (e.g. the counters of $\lambda_{2,3}$, $\lambda_{4,1}$ and $\lambda_{3,2}$ of Table I). If the cover counter of the row is equal to the desired cover factor of clause (n), the row can be removed and the matrix can be reordered.

TABLE I. Coverage matrix expressing the clause as disjunction of boolean variables: $\lambda_{2,3} = a_1 \vee a_2 \vee a_8 \vee a_9$; $\lambda_{4,1} = a_2 \vee a_3 \vee a_6$; $\lambda_{3,2} = a_2 \vee a_4$; $\lambda_{3,4} = a_1 \vee a_3 \vee a_5 \vee a_7 \vee a_{10}$; $\lambda_{3,5} = a_3 \vee a_5 \vee a_7 \vee a_{10}$.

	1	2	3	4	5	6	7	8	9	10
$\lambda_{2,3}$	1	1	0	0	0	0	0	1	1	0
$\lambda_{4,1}$	0	1	1	0	0	1	0	0	0	0
$\lambda_{3,2}$	0	1	0	1	0	0	0	0	0	0
$\lambda_{3,4}$	1	0	1	0	1	0	1	0	0	1
$\lambda_{3,5}$	0	0	1	0	1	0	1	0	0	1

C. Probability of Missed Aruco Code Detection

Let γ_f be the non-zero probability of not detecting an Aruco code. In such conditions, constraint (6) can then be guaranteed only in a stochastic sense, that is with a probability greater than ν_d. Being n the number of landmarks for each clause, the constraint (6) is then violated if the sensor misses a number of visual landmarks greater than or equal to n. Let m_i be the number of landmarks along the i–th trajectory T_i, the probability of missing n consecutive landmarks is given by the following recursive formula:

$$\zeta(m_i, n) = \begin{cases} 0 & \text{if } n > m_i \\ \gamma_f^n + \sum_{j=1}^{n_i} (1-\gamma_f)\gamma_f^{j-1}\, \zeta(m_i - j, n) & \text{otherwise.} \end{cases}$$

Of course, such a probability increases with m_i and decreases if n grows. Since, the uncertainty associated with T_i depends on the possibility to measure contextual information along a given path (hence increasing m_i), $\zeta(m_i, n)$ is computed contextually with the greedy solution, while checking if the probability constraint ν_d is met. In other words, the greedy algorithm searches for the smallest n such that

$$1 - \zeta(m_i, n) > \nu_d \tag{8}$$

is satisfied for all the trajectories T_i.

IV. Simulation Results

This section presents the simulation results of two different scenarios based on model (2) with the parameters reported in Table II. Some of these values derive from a preliminary characterisation of the *FriWalk* sensors, e.g. front camera and encoders. For the sake of brevity, only the results with a constant target uncertainty are reported, i.e., $\xi(p_k)$ is the same $\forall p_k$.

TABLE II. PARAMETER VALUES BASED USED FOR SIMULATIONS.

d_r	0.30 m	d_l	0.30 m	σ_r	0.1 mrad	σ_l	0.1 mrad
b	0.8 m	σ_{x_c}	2 cm	σ_{y_c}	6 cm	σ_{θ_c}	0.03 rad
σ_{ρ_r}	50 mm	σ_{θ_r}	0.17 rad	R	2.5 m	r	0.8 m
β	$\pi/3$ rad	γ_f	0.4	ν_d	0.95	$\xi(p_k)$	0.8 m

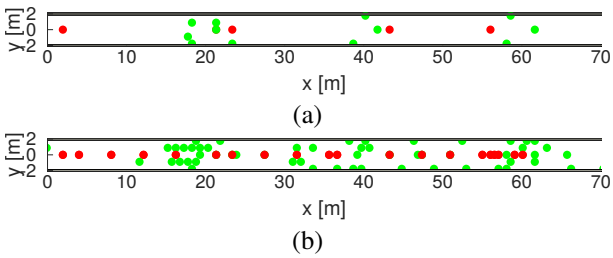

Fig. 3. Landmark placement in a corridor with straight paths and with (red dots) or without (green dots) the RGB-D camera for (a) $n = 1$ and (b) $n = 5$.

A. Demo scenarios

The first scenario is conceived to highlight the features of the proposed greedy placement and consists of a 4×70 m corridor. Globally, 100 straight paths in both directions and uniformly spaced have been generated in the corridor. The possible landmark locations are 430 and result from the solution of [7] using the values of R, r and β reported in Table II. Figure 3 shows the results of landmark placement with (red dots) and without (green dots) using the front RGB-D camera for $n = 1$ (a) and 5 (b). Notice how the use of contextual information (namely the detection of walls) drives the algorithm to place the landmarks in the centre of the corridor, where the RGB-D camera cannot collect data. Another interesting result is shown in Figure 4, which represents the box-and-whiskers diagrams, computed over all the generated paths, of the probabilities $\zeta(m_i, n)$ of not detecting an Aruco code for different values of n with (a) and without (b) using the front RGB-D camera. The larger variability when the RGB-D camera is used (see Figure 4-a) (especially if $n = 1$) depends on the particular path of the agent. When the agent moves in the middle of the corridor the walls are not detected. Hence the uncertainty constraint $\xi(p_k)$ is more likely to be violated. This is not the case for the paths detecting the walls. Of course, this phenomenon does not occur if the RGB-D camera is not used, as shown in Figure 4-b. Notice that, in both cases, to meet the desired stochastic threshold of $\nu_d = 0.95$, than at least $n = 5$ landmarks are needed. However, from Table III it is evident that the number of landmarks to be placed is almost three times larger.

B. Realistic Environment

The realistic environment chosen for simulation purposes is the Department of Information Engineering and Computer Science (DISI) of the University of Trento. In this case, 1000 random trajectories have been generated starting and ending in different rooms. The paths are quite regular, as can be expected in a highly constrained indoor environment. The dots in Figure 5(a) represent the 7363 possible landmark positions obtained using the geometric approach described in [7]. Assuming that the initial positions are known with minimal uncertainty and that the Aruco codes are always detected (i.e. $\gamma_f = 0$), the result of the placement based on the greedy algorithm is shown in Figure 5(b) for $n = 1$.

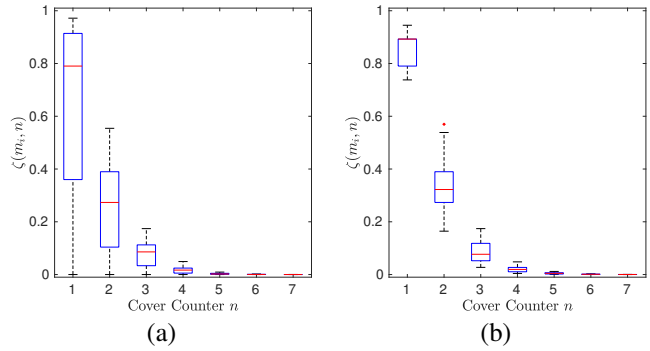

Fig. 4. Box-and-whiskers plots of the probabilities $\zeta(m_i, n)$ of not detecting a landmark along the corridor shown in Figure 3 with (a) and without (b) using the front RGB-D camera.

TABLE III. NUMBER OF LANDMARKS PLACED IN THE CORRIDOR SHOWN IN FIGURE 3 WITH AND WITHOUT USING THE FRONT RGB-D CAMERA.

	Cover Counter n						
	1	2	3	4	5	6	7
no RGB-D	12	24	36	45	57	66	78
RGB-D	4	9	14	18	22	27	32

Again, red and green dots are obtained including or excluding the data from the RGB-D camera, respectively. If $\gamma_f = 0.4$ and $\nu_d = 0.95$, at least $n = 5$ landmarks per clause are needed to satisfy (8) for all the trajectories, resulting in the deployment shown in Figure 5(c). Notice that the landmarks are placed mainly along the corridors, since the density of possible paths is higher than in the rooms. Moreover, when the RGB-D camera is used, the number of landmark clusters as well as their size are smaller. This is confirmed by the results in Table IV, which reports the amount of landmarks actually deployed. The results on the probability of failure $\zeta(m_i, n)$ are not reported because they are similar to those shown in Figure 4. As a final comment, it is worth emphasising that, even if the number of trajectories and potential landmark locations is quite large, the computation time of the greedy placement algorithm implemented in Matlab and running on a 3.50 GHz Intel Core i7 with 8 GB of RAM is about 45 minutes.

V. CONCLUSIONS

This paper presents a greedy algorithm solving the problem of minimal landmark placement for indoor localisation based on the combination of dead-reckoning, visual landmark detection and contextual information. The problem is constrained by the desired location-dependent target uncertainty and by is affected by the probability of missing a landmark. The effectiveness of the proposed approach is demonstrated with simulations in a realistic environment. Future developments will focus on actual experiments on the *FriWalk*, on classification of the trajectories (core and optional trajectories, with probabilities) and on the usage of heterogeneous landmarks, e.g. visual Aruco codes and RFID tags.

ACKNOWLEDGMENT

The activities described in this paper have received funding from the European Union Horizon 2020 Research and Innovation Programme - Societal Challenge 1 (DG CONNECT/H) under grant agreement no. 643644 for the project ACANTO - A CyberphysicAl social NeTwOrk using robot friends.

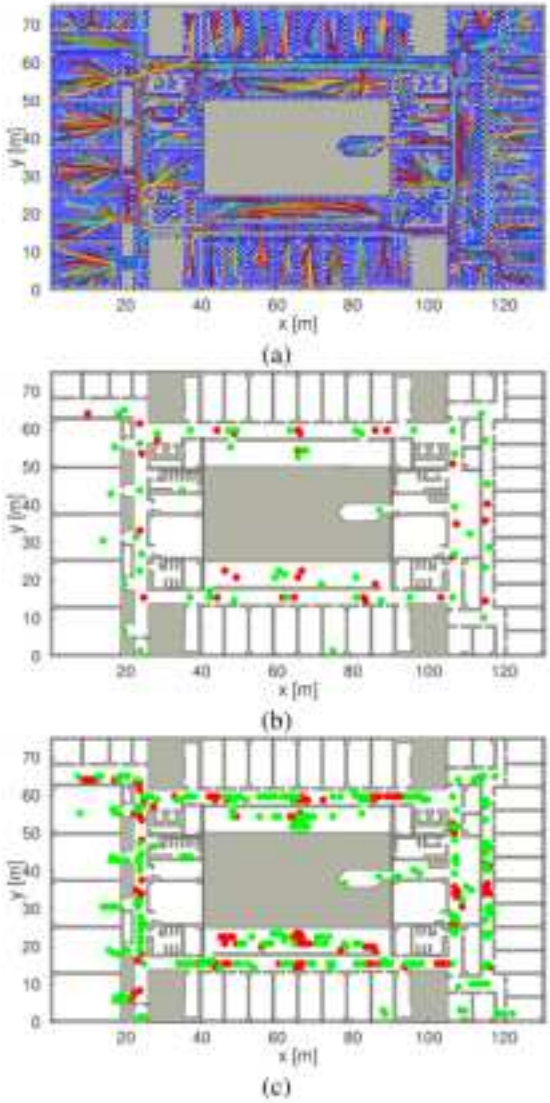

Fig. 5. Simulation results in a realistic scenario. In (a) the *FriWalk* trajectories generated by a path planner are shown on the DISI map. In (b) and (c) the results of landmark placement with (red dots) and without (green dots) using the RGB-D camera are plotted for $n = 1$ with $\gamma_f = 0$ and $n = 5$ with $\gamma_f = 0.4$, respectively.

TABLE IV. NUMBER OF LANDMARKS PLACED WITH AND WITHOUT USING THE RGB-D CAMERA IN THE REALISTIC CASE SHOWN IN FIG. 5.

	Cover Counter n						
	1	2	3	4	5	6	7
no RGB-D	53	100	155	203	256	310	360
RGB-D	33	67	98	120	158	188	216

REFERENCES

[1] D. Dardari, P. Closas, and P. M. Djuri, "Indoor tracking: Theory, methods, and technologies," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1263–1278, Apr. 2015.

[2] L. Mainetti, L. Patrono, and I. Sergi, "A survey on indoor positioning systems," in *Proc. International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Croatia, Sep. 2014.

[3] A. Colombo, D. Fontanelli, D. Macii, and L. Palopoli, "Flexible indoor localization and tracking based on a wearable platform and sensor data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 4, pp. 864–876, Apr. 2014.

[4] P. Nazemzadeh, F. Moro, D. Fontanelli, D. Macii, and L. Palopoli, "Indoor positioning of a robotic walking assistant for large public environments," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 11, pp. 2965–2976, Nov. 2015.

[5] D. Ayllón, H. A. Sánchez-Hevia, R. Gil-Pita, M. U. Manso, and M. R. Zurera, "Indoor blind localization of smartphones by means of sensor data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 4, pp. 783–794, Apr. 2016.

[6] N. Simon, J. Bordoy, F. Höflinger, J. Wendeberg, M. Schink, R. Tannhäuser, L. Reindl, and C. Schindelhauer, "Indoor localization system for emergency responders with ultra low-power radio landmarks," in *Proc. of IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2015, pp. 309–314.

[7] P. Nazemzadeh, D. Fontanelli, and D. Macii, "Optimal Placement of Landmarks for Indoor Localization using Sensors with a Limited Range," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Madrid, Spain, Oct. 2016, pp. 1–8.

[8] H. Strasdat, C. Stachniss, and W. Burgard, "Which landmark is useful? Learning selection policies for navigation in unknown environments," in *IEEE Intl. Conf. on Robotics and Automation*. IEEE, 2009.

[9] Y. Wang, Z. Tu, Q. Wang, Y. Shen, and J. Li, "Flocking based distributed deployment for target monitoring in mobile sensor networks: Algorithm and implementation," in *Proc. of IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2012, pp. 2472–2477.

[10] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson, "Landmark selection for vision-based navigation," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 334–349, Apr. 2006.

[11] M. Beinhofer, J. Müller, and W. Burgard, "Near-optimal landmark selection for mobile robot navigation," in *IEEE Intl. Conf. on Robotics and Automation*. IEEE, 2011, pp. 4744–4749.

[12] A. A. Khaliq, F. Pecora, and A. Saffiotti, "Inexpensive, reliable and localization-free navigation using an RFID floor," in *European Conf. on Mobile Robots (ECMR)*, Lincoln, United Kingdom, Sep. 2015.

[13] F. Zanatti, D. Fontanelli, L. Palopoli, D. Macii, and P. Nazemzadeh, "Optimal Placement of Passive Sensors for Robot Localisation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and System*. Daejeon, South Korea: IEEE/RSJ, Oct. 2016, pp. 4586–4593.

[14] M. Beinhofer, J. Müller, and W. Burgard, "Effective landmark placement for accurate and reliable mobile robot navigation," *ROBOT. AUTON. SYST.*, vol. 61, no. 10, pp. 1060–1069, Oct. 2013.

[15] V. Magnago, L. Palopoli, R. Passerone, D. Fontanelli, and D. Macii, "A Nearly Optimal Landmark Deployment for Indoor Localisation with Limited Sensing," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Sapporo, Japan: IEEE, Sept. 2017, pp. 1–8.

[16] P. Nazemzadeh, D. Fontanelli, D. Macii, and L. Palopoli, "Indoor Localization of Mobile Robots through QR Code Detection and Dead Reckoning Data Fusion," *IEEE Trans. on Mehcatronics*, 2017, available on line.

[17] "ACANTO: A CyberphysicAl social NeTwOrk using robot friends," http://www.ict-acanto.eu/acanto, February 2015, EU Project.

[18] L. Palopoli et al., "Navigation Assistance and Guidance of Older Adults across Complex Public Spaces: the DALi Approach," *Intelligent Service Robotics*, vol. 8, no. 2, pp. 77–92, 2015.

[19] P. Bevilacqua, M. Frego, E. Bertolazzi, D. Fontanelli, L. Palopoli, and F. Biral, "Path Planning maximising Human Comfort for Assistive Robots," in *IEEE Conference on Control Applications (CCA)*. Buenos Aires, Argentina: IEEE, Sept. 2016, pp. 1421–1427.

[20] F. Moro, A. D. Angeli, D. Fontanelli, R. Passerone, D. Prattichizzo, L. Rizzon, S. Scheggi, S. Targher, and L. Palopoli, "Sensory stimulation for human guidance in robot walkers: A comparison between haptic and acoustic solutions," in *IEEE International Smart Cities Conference (ISC2)*, Trento, Italy, Sept. 2016, pp. 1–6.

[21] M. Andreetto, S. Divan, D. Fontanelli, and L. Palopoli, "Harnessing Steering Singularities in Passive Path Following for Robotic Walkers," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*. Singapore: IEEE, May 2017, pp. 2426–2432.

[22] ——, "Path Following with Authority Sharing between Humans and Passive Robotic Walkers Equipped with Low-Cost Actuators," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2271–2278, Oct. 2017.