IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 19, 2018, accepted December 4, 2018, date of publication December 12, 2018,
date of current version December 31, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2885648

Sensor Fusion for Tour-Guide Robot Localization

BIEL PIERO E. ALVARADO VASQUEZ[1], (Student Member, IEEE), RUBEN GONZALEZ[2], FERNANDO MATIA[1], AND PALOMA DE LA PUENTE[1]

[1]Centre for Automation and Robotics, Universidad Politecnica de Madrid, 28006 Madrid, Spain
[2]Escuela Técnica Superior de Ingenieros Industriales, Universidad Politecnica de Madrid, 28006 Madrid, Spain

Corresponding author: Biel Piero E. Alvarado Vasquez (biel.alvarado@upm.es)

This work was supported in part by the DPI2017-86915-C3-3-R project COGDRIVE: Técnicas de Inteligencia Artificial y Ayuda a la Navegación Autónoma, in part by the DPI 2014-53525-C3-1-R project NAVEGASE: Navegación Asistida Mediante Lenguaje Natural that is funded by the Spanish Ministry of the Economy and Competitiveness, in part by the RoboCity2030-III-CM project Robótica aplicada a la mejora de la calidad de vida de los ciudadanos, fase III funded by the Programas de Actividades I+D en la Comunidad de Madrid and co-funded by EU Structural Funds under Grant S2013/MIT-2748.

ABSTRACT Doris, the social robot girl, is under development to be employed in museums and trade fairs as a tour guide. External sensorial information must be inputted so that Doris moves around each new location by using landmark identification points that can improve the real localization of the robot in combination with an extended Kalman filter. Doris is equipped with a semantic map that contains several information points such as the building structure, sites that the robot must pass, features (obstacles) of the built environment, and landmark locations. Three additional sensors were installed on Doris: a laser range finder LMS-200, an omnidirectional Mobotix C25 camera, and an RFID system Speedway Revolution 220 by Impinj. The use of these sensors implies the use of different types of landmarks: 35-cm-high circular landmarks, placed on the ground and covered with a reflective laser-detectable material; markers similar to QR codes placed at 250 cm above the ground level that the omnidirectional camera can identify; and RFID detectable dogbone antennas. One contribution is to prove a simple methodology of localization by using sensor fusion with a semantic map, without mapping the whole environment by creating a point cloud map and without using the SLAM technique. Additionally, another contribution for the research is to define a good methodology for a precise sensors calibration. The initial results showed that each sensor functions efficiently, when using only the laser and the camera, due to the low accuracy of the RFID system alone. The final results show the behavior of the robot localization in the presence of people and different objects when both sensors are working at the same time. Occlusions may affect the reflective landmarks or visual markers. Therefore, the sensor fusion is implemented to achieve better robustness in the location estimation.

INDEX TERMS State estimation, extended Kalman filter, indoor localization, sensor fusion, laser localization, visual localization, RFID localization, social robotics.

I. INTRODUCTION

The presence of people, sculptures, counters, and paintings, among other factors mean that museums and trade fairs are very complex environments. Robot localization is therefore essential for any high-level task that it might perform in that kind of environment. Many studies have been aimed at achieving dependable indoor localization of robots. The study presented by [1] implemented a sensor fusion with a sonar sensor and a Microsoft Kinect for acquiring depth information. Another, in [2], described a robust Multi-sensor fusion localization for indoor mobile robots. It was based on a set-membership estimator, developed by fusing a laser scanner and an odometer, using a point-to-line iterative closest point approach, to match successive environmental data collected by the laser scanner, with which the relative pose transformations of the robot were estimated. A further sensor fusion technique, in [3], proposed a 3D mobile pose estimation system for indoor application, based on the cooperative sensor fusion of radar, ultrasonic, and odometrical data, using an extended Kalman Filter, determining a position error of 15*cm*. Similar work was performed in [4], this time using an Ultra-Wideband (UWB), Indoor Positioning System (IPS) and an Inertial Navigation System (INS) achieving real-time indoor navigation and tracking of automated guided vehicles (AGVs) and mobile robots in factories and warehouses. The sensor fusion algorithm developed in this contribution consisted of delayed compensation, based on the position estimation from the IPS, and a multi-rate extended Kalman

Filter that combined the delay-compensated position from the IPS and the measurements from the INS. The UWB was used to validate the sensor fusion approach, achieving a position error standard deviation of 3.7*cm* for linear movements and 1.7 degrees for rotational movements. A conference paper reported a further work that described a sensor fusion system using odometry and magnetic sensor data for an indoor localization system [5] with the odometry based on the Ackerman steering geometry, and directly calculated from encoder information on two rear wheels and a third wheel for steering. The sensor data fusion was performed with an Unscented Kalman Filter, so that the mean and covariance of the nonlinear system function could be estimated by the Unscented Transformation. A very recent study has been developed in [6], which proposes a geometric feature-based method to solve a Simultaneous Localization And Mapping (SLAM) problem in an unknown structured environment using a Kinect sensor for short range and low Field of View (FoV) measurement. The RANSAC algorithm and a fast SLAM were used for detection and mapping, resulting in an uncertainty reduction of the robot pose in the prediction step that improved pose accuracy where data on geometric features were available. Other work performed by [7] using a Kinect and two other sensors achieved good results by using the WiFi signal for large open areas, and the depth camera and laser for corridors and offices. Reference [8] fused sonar with odometry and the use of particle filter for final localization. Studies in external environments also perform a sensor fusion like in [9] fusing the information provided by a GPS with odometry through an EkF.

Studies by the Intelligent Control Group of the Centre of Automation and Robotics of the UPM-CSIC were performed on sensor fusion, such as [10], which solves the SLAM problem by using the extended Kalman Filter to describe the problems and to find solutions for mapping large environments with excellent results. The same author proposed an efficient geometric approach for Simultaneous Localization And Mapping (SLAM) based on an extended Kalman Filter in [11], where maps are built using orthogonal shape constraints to process inconsistent estimations.

Some years later, the FastSlam, an upgrade to the SLAM problem, was presented in [12] and successfully applied to the construction of feature-based maps in indoor environments. In the same year, [13] used B-Splines as a modeling tool and the set of control points defining their shapes was used to form a complete and compact description of the environment, making it feasible to use an extended Kalman-Filter-based SLAM algorithm. Other SLAM research was performed in [14], by introducing *a priori* knowledge on the latent structure of the environment in SLAM, which can improve the quality and consistency of results and their solutions. It described a general framework for detection, evaluation, incorporation, and removal of structural constraints into a graph formulation of SLAM. The study explained how the inclusion of different kinds and levels of features in a hierarchical order permits the system to discover new structures easily and explains why it makes more sense than other possible representations.

Localization based on Wireless Networks is attributed high importance in recent research, among which the most common system is Radio Frequency Identification (RFID). Different approaches have been developed, such as [15] that introduces a positioning algorithm for RFID tags using two mobile readers that process passive or active tags with randomly distributed known locations. The location was estimated using multilateration with the landmarks and a probabilistic RFID map-based technique with Kalman Filtering. Another study was developed by [16], which examines Direction Of Arrival (DOA) estimation methods and their application to localization and tracking problems of RFID tags. A subsequent RFID positioning system was implemented by [17], based on an object carrying an RFID reader module that reads low-cost passive tags installed next to the object path and that uses a Kalman Filter, the measurements of which are the backscattered signal power propagated from the nearest RFID tag and a tag-path position database. The algorithm is used to produce an estimate of the location of the reader, neglecting tag-reader angle-path loss, where it is compensated by the signal strength information measurement that is received.

The study presented by [18] described the fusion between an RFID System and an ultrasonic sensor that removed the uncertainties of the RFID system. Reference [19] and that was designed to achieve a fast and robust environment for RFID distance estimation by using phase and Received Signal Strength Indicator (RSSI) measurement to estimate distance. More recent research performed in [20] proposed a hybrid multi-sensor fusion strategy for positioning in tunnels where an RFID system was used to achieve a preliminary position, by using the RSSI to estimate distances between the reader and the tags. It then used a Least Mean Square (LMS) federated filter for subsequent global fusion. In [21], a particle filter, laser-based clustering, and an RFID system were used to improve the localization of moving objects by achieving an accuracy of up to 25*cm*.

The main aim of this research is to provide Doris, the social robot, with a multi-sensor architecture based on Laser Range Finder LMS-200, an omnidirectional camera Mobotix C25, and a RFID system Speedway Revolution 220 by Impinj, to complement the internal odometry of the robot. The robot also has an Extended Kalman Filter that will achieve dependable localization with the sensor fusion in built environments such as museums. To that end, Doris will be provided with a semantic map to inform it of the basic structure of the building around which it will move. This map is constructed in XML containing some semantic information and the locations of the sensor detectable landmarks. These landmarks were strategically located in the areas where the robot circulated. Three types of landmarks were considered: the first type was detected by LRF, the second type are the markers that will be seen and decoded by the omnidirectional camera,

and the the tags represent the third type that will be detected by the RFID System.

The benefits and disadvantages of each sensor will be discussed and a procedure will be explained for the estimation of the standard deviation for each variable of each sensor and its integration in the EKF to achieve good convergence. A primary study was done at [22] where the tests were performed in a simple environment with no people and no objects around. This study makes deeper exploration in the technique used by describing a procedure to obtain a good approximation of covariances matrices involved (process and sensors). This work also adds a new sensor and explains why RFID System is not very useful for localization tasks comparing it to the UltraWide Band System and a new test is developed in a much complex environment with presence of objects and people moving around. The structure of the paper is as follows. In section II a brief explanation of Doris' architecture will be explained, detailing how the modules are connected to each other, to perform several tasks. Section III will explain how distance or angular information is obtained from each sensor, so that the estimation error can be also calculated in section IV. In section V, the fusion of sensor information with the EKF will be explained through an example, and finally, in section VI, the performance of the robot will be detailed in evaluation tests at the Higher Technical School of Industrial Engineering of Madrid.

II. DORIS, THE SOCIAL ROBOT GIRL

Doris is a mobile robot conceived to work in a dynamic indoor environments for interacting with people. It builds on the upgrading of Blacky and Urbano, the two previous robots that the group at CAR has been developing over recent years at the Universidad Politécnica de Madrid.

As stated, Doris is expected to work in a system where she interacts with people in public buildings such as museums, theaters, conferences, trade fairs, etc

A. HARDWARE ARCHITECTURE

The hardware consists of three parts:

- The platform, designed by Adept MobileRobots, is a differential steer mobile robot, which has built-in sensors (bumpers, sonars, and laser), all connected to the serial port of the internal computer. This computer has USB ports, ethernet and a WIFI connection. Figure 1a shows the platform.
- The skeleton, attached to the mobile platform, made of methacrylate, holds up speakers, the RFID antennas, a USB HUB, a PoE switch and the robotic head. The torso, is designed to represent a human appearance. In the future, an arm will be attached to develop pick and place tasks.
- The robot user can select different expressions that the head, attached to the skeleton, with 20 DoF will show. The robot is also speech capable and its mouth will synchronize with the sound that is emitted; a combination known as viseme-based acoustic speech, that is shown in figure 1c

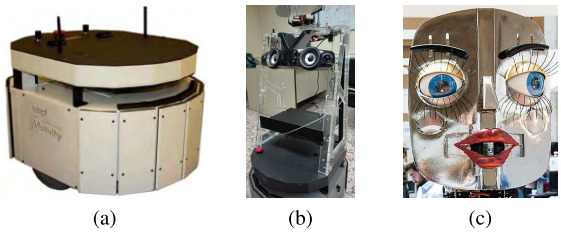

FIGURE 1. Doris hardware overview. (a) Platform. (b) Body. (c) Face.

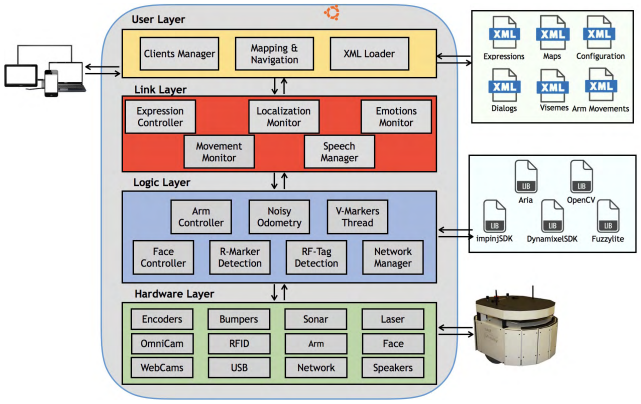

FIGURE 2. Software architecture of Doris.

The information gathered from all sensors, is used for indoor localization. The joint sensor operation will be detailed in section II-B and in section V-C.

B. SOFTWARE ARCHITECTURE

The software works as a Client-Server application, enabling direct communication between Doris (server) and the (client) that controls it. Nowadays, there are three different types of client applications for Doris: Web, Desktop, and Mobile devices. The server application is divided into four layers detailed in figure 2. Each block on the figure details a task, that is asynchronously under execution on a different thread, which means that in the higher level of the architecture, there is a task handler.

- **Hardware Layer**: starts connections with different devices installed on the platform (encoders, bumpers, sonars, cameras, laser, etc), as well as with USB and ethernet devices, at a level with permissions for read/write data provided by superior levels. e.g. the robot position, the speed at which the robot must move, etc.
- **Logic Layer**: at this level, data sent from first layer are processed, for later use by the immediate superior layer, e.g. data from encoders is translated into differential polar coordinates $[\Delta\rho\,(k)\ \ \Delta\phi\,(k)]^T$ for odometry, and then by using the state space equation 27 can be converted into an oriented position $[x\,(k)\ y\,(k)\ \theta\,(k)]^T$. All of the modules in this layer are threads that are launched since the application is executed, for example:

R-Marker is a thread dedicated to detect the reflective landmarks, the V-Markers is other thread detecting the visual markers, and the RF-Tag detection, is a thread for detecting the RFID tags.
It also turns on the server communications with a networked camera and an RF device.

- **Link Layer**: instructions for robotic movements are generated at this level of Doris' programming. The localization task that takes place on this layer processes data from the logic layer, by using odometrical information and data from sensor detector tasks. Emotion and expressive tasks are also performed, where they can command the face and/or the arm. A movement monitor is also included to supervise which actions Doris must perform depending on the localization.
- **User Layer**: Client communications are performed at the user layer along with database loading and the navigational commands are inputted by the user. Doris can manage up to ten users at the same time, but only one can have full access to perform tasks.

III. OBTAINING SENSOR INFORMATION

As shown in section I, there are several studies in which reliable accuracy of robot localization is achieved using different types of sensors, searching for less expensive ones that can perform rapid detection. Three different types of sensor are installed in Doris. The RFID System, the omnidirectional camera, and the laser range finder.

In this section, the method of obtaining data from the sensors, by detecting their respective landmarks and their location by either distance or angle from the robot, are presented.

A. RFID DETECTION

With regard to pose estimation using RFID, the most common method is based on triangulation with the received signal strength indicator (RSSI) as explained in [23], due to its low cost and ubiquity, although those advantages do not imply that it is the best approach to locate objects. RSSI measurements involve high uncertainty margins, according to [24] achieving a mean error from $2.9m$ to $4.3m$ on a floor area of $980m^2$. The most notable contribution was made by [25] in which a tag position error of $1.08m$ was achieved within a limited space.

Other application can be found in [26] which places each the RFID tag on the floor in a fixed and constant distance respect to each other creating a tag matrix and places the reader beneath the robot in order to detect each tag and this fixes the robot position in $X(k)$ and $Y(k)$ and the robot orientation is fixed by using a camera and detecting QR codes.

The applications on guiding robots using RFID systems in narrow areas (through hallways or negotiating a door) are really low, so a certain degree of precision is needed that is not provided by this type of system, but it can be useful for other localization tasks as global positioning.

(a) (b) (c)

FIGURE 3. RFID System installed on Doris. (a) Antenna. (b) Impinj R220 Reader. (c) RFID tag dogbone.

The RFID Impinj R220 reader system (figure 3a) installed in Doris has two right circular polarity antenna S8658PR (figure 3b), from which three different measures can be obtained:

1) **RSSI**: Received Signal Strength Indicator, which provides an estimate of the tag backscatter signal power received P_R in Watts with an RFID reader [27] by using the Friis equation:

$$P_R = \frac{G_T^2 * \lambda^2 * \sigma}{(4 * \pi)^3 * R^4} * P_T \tag{1}$$

where:
P_T is the reader transmitter power at the transmit antenna input (Watts).
G_T is the reader antenna gain.
λ is the carrier wavelength (m), which can be estimated as: $\lambda = \frac{c}{f}$ where, c is the speed of the electromagnetic wave and f is the operational frequency of the antenna.
σ is the Tag Radar Cross Section (m^2), computed as $\sigma = \frac{G_R * \lambda^2}{4 * \pi}$, where G_R is the antenna receiver gain.
R is the distance between reader and tag (m).

2) **RF phase angle**: represents the phase rotation angle over distance [27], which the reader obtains by using the following formula:

$$\theta = 2 * \pi * \left(\frac{2 * R}{\lambda}\right) + \theta_T + \theta_R + \theta_{TAG} \tag{2}$$

where, θ_T, θ_R, θ_{TAG} are the phase angle rotations of the transmitter circuit, the receiver circuit, and the tag reflection characteristic, respectively.
As the phase is a periodic function with a period of $2*\pi$ radians, the values will repeat at distances separated by multipliers of half a wavelength: $R_n = \frac{n*\lambda}{2}$ where, $n = 0, 1, 2, ...$

3) **Doppler Frequency**: is the shift of the received signal at the reader, due to relative motion between the reader and the tag. This feature is helpful for determining the tags that are in motion. This feature will be used for future applications.

Both, RSSI and RF Phase provide information on the distance by computing the equation based on the information from the reader obtained from both (1) and (2):

$$R = \sqrt[4]{\frac{G_T^2 * \lambda^2 * \sigma}{(4 * \pi)^3 * P_R} * P_T} \tag{3}$$

TABLE 1. OpenRTLS UWB price.

Quantity	Description	Price Unit (€)	Subtotal (€)
1	Master anchor	220	220
10	Anchors	220	2200
1	Tag	65	65
1	SDK	100	100
1	Switch	80	80
100	Wiring	3	300
		Total:	2965

where, the antenna operates at a frequency of 865.9 MHz, resulting in a wavelength of $\lambda = 0.3464m$ and the transmitter and receiver gain are also fixed values.

$$R = (\theta - \theta_{TT}) * \frac{\lambda}{4 * \pi} \tag{4}$$

where, θ_{TT} is the total phase rotation of $\theta_T + \theta_R + \theta_{TAG}$ to be determined in section IV-B.

The Ultra Wide-Band range measurement (UWB) is a radio technology similar to the RFID system which operates at higher frequencies. Nevertheless, it was not considered during the selection process due to the high price it has when compared to the RFID System. Although it is well known that it has a good accuracy for tags position estimation and this could be useful for localization of the robots, some issues must be considered:

1) Three anchors (antennas) minimum in each room are needed to achieve a good tag localization.
2) As stated, Doris is intended to work in a complex environment like a museum as a tour-guide, and it will be very difficult to install UWB anchors at these places.
3) Supposing that museums are in the will to install the anchors, another problem may arise because the radio frequency wave of the anchors can penetrate thin walls and some museum have very thick walls and depending on the area of the room, extra anchors might be needed.
4) Costs increase when more anchors are added and a whole UWB system would be very expensive. Table 1 shows the costs of installing an UWB network.

This table shows the price of an UWB network provided by OpenRTLS and using 11 anchors in an area of $10mx10m$, with the purpose to achieve a precise localization of the robot. Compared to the RFID system where antennas can be installed on the robot and tags can be sticked to the walls, the cost of the UWB System is a lot higher.

B. VISUAL DETECTION

Visual detection has been used in mobile robots like used in [28] and [29] which uses a set of QR codes creating a matrix and covering the entire floor to achieve the robot orientation and [30] which is similar to the previous but the camera points to the ceiling instead. The sensor installed on the robot platform is an omnidirectional camera and the fiducial markers are placed on the walls with certain distribution along the room where the robot is going to perform. The reason for installing this type of camera is to provide Doris

FIGURE 4. Omnidirectional camera C25 by Mobotix.

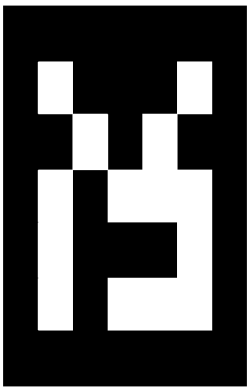

FIGURE 5. Example of a visual landmark.

with a 360 degree field of view, so it can see the whole room and can pick up markers from all directions; for example, in a museum with halls of different sizes. The model used is the Mobotix C25 hemispheric camera, as shown in figure 4. This is an indoor ceiling camera with a diameter of $12cm$ and a weight of $200g$. It includes a light-sensitive 6MP day and night sensor and is powered and communicates via the ethernet.

The camera offers different types of display modes of different quality and resolution. In this case, only the full view is used at a resolution of 1280×960 pixels.

The main goal is to recognize and to estimate the angle of the visual markers. These markers are a QR binary code containing information represented in a 5×5 matrix. This information contains the landmark orientation as well as the map, sector and landmark Id's. The appearance of the marker is in figure 5, printed over an A3 paper size. This code is robust against rotation and perspective views. The markers are located over the wall at 2.5 m above ground level, considering the height of the robot that is ± 1.45 m.

So the problem of locating the robot by using visual landmarks in large areas concerns only the determination of its angle from the center of the camera lens to the center of the marker. This problem can be solved by using OpenCV library, following the steps listed below:

1) Capture a frame from the scene, as shown in figure 6a.
2) The captured frame is converted to grayscale, as in figure 6b.

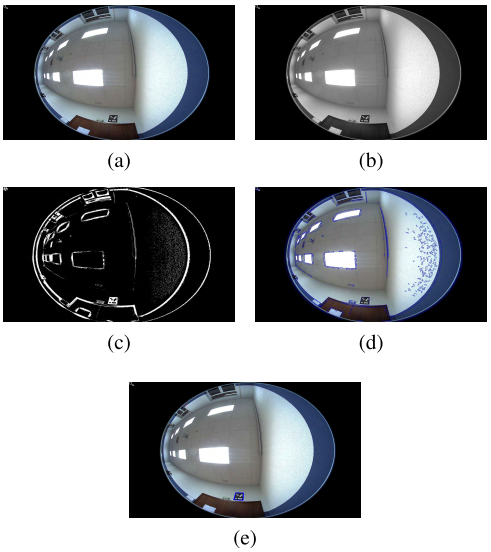

FIGURE 6. Images obtained in the treatment process. (a) Full color scene. (b) Captured scene converted to grayscale. (c) Grayscale thresholded according to environment. (d) Contours detected matching criteria. (e) Markers detected.

3) Image binarization. Due to different light conditions, the adaptive threshold is chosen for a fixed image so that it is homogenous, which requires two different values: a pixel size, which is invariant between images, and an index obtained by analyzing the histogram with which the mean intensity value is set.
4) Contour detection. The binary image is submitted to find all available contours by retrieving a vector of points. This contour is stored in another vector of contours, only if the contour size is higher or equal than a predefined number of points, which in this case is set at 50 points. This value is used for contours that might be too small to be studied and are considered as noise. Figure 6d shows the contours detected in the scene.
5) Possible marker detection. Consisting of selecting only those contours which are a convex polygon with only 4 sides and discarding everything else. At this point, a list of possible markers is created and refined by keeping only those that match a square or a rectangle.
6) Real marker selection and decoding. As the algorithm must be robust for rotations and transformations of the possible marker that is detected, the perspective transform must be calculated for each one. A transformation that is done by using four pairs of the corresponding points in a real scene. Once the perspective transformational matrix is estimated, a perspective transformation to the image is applied, and the resulting image is sent for decoding.

   As the image contains only black and white cells of a known size, the decoding first consists of building a bitmap based on the information obtained by the image. As it is a binary image, the non-zero counting for each cell is performed to establish every bit, and by applying the Hamming distance, the marker can be recognized and the associated information can be obtained. After the perspective correction, the image can be in 4 positions and the Hamming distance of each one is tested. The zero distance will be chosen as the winning rotation, obtaining a code from the marker, and searching for it in a database that yields a reference for global positioning.
7) Angle estimation. Considering that the camera offers a circular (somewhat similar to fish-eye) image, the angle estimation is done through only one calculation:

$$\phi = tan^{-1}\left(\frac{Y_{ImageCenter} - Y_{MarkerCenter}}{X_{ImageCenter} - X_{MarkerCenter}}\right) - \frac{\pi}{2} \quad (5)$$

   As the figure 6e shows, after applying the previous steps, the markers are detected. As the camera is connected to other computer where the image processing is executed, processing 20 frames per seconds. Then the data is transmitted to the onboard computer every 10 ms via UDP socket.

FIGURE 7. Reflective landmark designed by the group.

C. REFLECTIVE LANDMARKS DETECTION

Laser technology is widely used in robotics, especially for map design using the SLAM algorithm. It is only used for landmark detection in this study with some specifications. These landmarks have a circular shaped reflective cover and were designed ad hoc.

One function of the laser is to add certain properties to each measurement; for example, the reflection intensity when the beam strikes a reflective surface. Taking advantage of this property, each landmark was provided with its respective reflective material.

The reason for their circular shape is because a planar landmark with the same reflective material in earlier tests could only be detected by the laser when the beam struck right in front of the landmark, otherwise, the beam would bounce to another direction and would not return to the laser. Figure 7 shows the prototype of the landmark design. Landmarks are designed to be positioned next to walls, hence their limitation to only 180 degrees.

It is assumed that the center of the reflective landmark may be located with maximum precision where it is a perfect cylinder of a known radius ($r = 0.045m$). The SICK laser adds a reflective intensity to each of the 361 measurements and identifies the landmarks. More than one intensity can belong

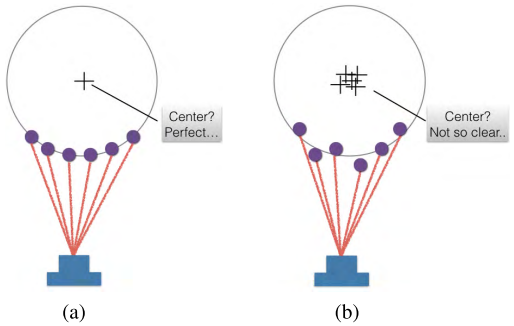

FIGURE 8. Measurement representation obtained by the laser. (a) Ideal laser behavior. (b) Real laser behavior.

to the same landmark. Hence, a classification condition must be implemented, in order to obtain the number of landmarks detected by the laser. First, only those measurements are selected that are of a higher intensity than zero. The condition for classification with the resulting vector of indices is as follows:

$$II\,[i] - II\,[i+1] < Threshold \tag{6}$$

where, II is the intensity index, meaning that the difference between two consecutive indices is lower than the threshold it will classify as the same landmark. Once the set of points associated for each landmark is obtained, the aim is then to obtain the landmark center. This point can be estimated by applying the Law of Sines and an iterative minimum squared error estimation for the error correction. So the aforementioned Law of Sines is used as an observation model.

Each measurement obtained by the laser contains the distance and the angle to the point represented as $[\rho(k)\ \phi(k)]^T$, and a set of these consecutive points represents a single landmark, so that for the *i-th* landmark, there are n points that represent its circumference, where $n \geq 2$.

$$\begin{bmatrix} \hat{\rho}_1^i(k) \\ \phi_1^i(k) \\ \hat{\rho}_2^i(k) \\ \phi_2^i(k) \\ \vdots \\ \hat{\rho}_n^i(k) \\ \phi_n^i(k) \end{bmatrix} \tag{7}$$

and the resulting mean with the radius added to the distance, for each landmark is expressed as:

$$\begin{bmatrix} \tilde{\rho}^i(k) + r \\ \tilde{\phi}^i(k) \end{bmatrix} \tag{8}$$

The Law of Sines is used as the observational model for the iterative minimum squared error estimation that is expressed as:

$$\frac{r}{sin(\tilde{\phi}^i(k) - \phi_j^i(k))} = \frac{\hat{\rho}_j^i(k)}{sin(\alpha)} = \frac{\left(\tilde{\rho}^i(k) + r\right)}{sin(\pi - \alpha - \tilde{\phi}^i(k) + \phi_j^i(k))} \tag{9}$$

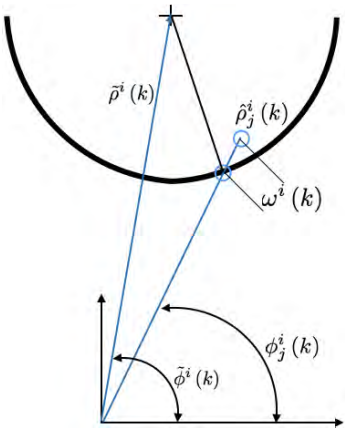

FIGURE 9. Estimation model for landmark center.

Therefore, the distance estimation measured by the laser, for each *j-th* measure, is defined by:

$$\begin{aligned} &h(j, \rho^i(k), \phi^i(k)) \\ &\quad = \hat{\rho}_j^i(k) = \tilde{\rho}^i(k) \cdot cos(\tilde{\phi}^i(k) - \phi_j^i(k)) \\ &\qquad - r \cdot cos\left(sin^{-1}\left(\frac{\tilde{\rho}^i(k)}{r} \cdot sin(\tilde{\phi}^i(k) - \phi_j^i(k))\right)\right) \end{aligned} \tag{10}$$

Figure 9 explains the notation of equation 10, showing the correspondence of each variable involved. So, the value $\rho_j^i(k)$ given by laser should be the sum of the estimation plus a random error $w(k)$, which will be considered with zero mean and $R(k)$ as variance, which reduces the observation model to:

$$\rho_j^i(k) = \hat{\rho}_j^i(k) + w(k) = h(j, \rho^i(k), \phi^i(k)) + w(k) \tag{11}$$

Using equation 10, the Jacobian matrix is computed as shown in equations 12, 13 and 14:

$$H^i(k) = \begin{bmatrix} \frac{\partial h}{\partial \rho} & \frac{\partial h}{\partial \phi} \end{bmatrix} \tag{12}$$

$$\begin{aligned} \frac{\partial h}{\partial \rho} &= cos(\tilde{\phi}^i(k) - \phi_j^i(k)) + \tilde{\rho}^i(k) \cdot \left(cos(\tilde{\phi}^i(k) - \phi_j^i(k))\right)^2 \\ &\quad \cdot r^{-1} \frac{1}{\sqrt{1 - \frac{\tilde{\rho}^i(k) \cdot \left(cos(\tilde{\phi}^i(k) - \phi_j^i(k))\right)^2}{r^2}}} \end{aligned} \tag{13}$$

$$\begin{aligned} \frac{\partial h}{\partial \phi} &= -\tilde{\rho}^i(k) \cdot sin(\tilde{\phi}^i(k) - \phi_j^i(k)) + (\tilde{\rho}^i)^2(k) \cdot sin(\tilde{\phi}^i(k) \\ &\quad - \phi_j^i(k)) cos(\tilde{\phi}^i(k) - \phi_j^i(k)) \\ &\quad \cdot r^{-1} \frac{1}{\sqrt{1 - \frac{\tilde{\rho}^i(k) \cdot \left(cos(\tilde{\phi}^i(k) - \phi_j^i(k))\right)^2}{r^2}}} \end{aligned} \tag{14}$$

By applying these equations to each *j-th* of the n measurement obtained by the laser, a least square method is applied for final correction. Two matrices are used to the achieve a good solution, one of them is the $\mathbf{P}^i(k)$, which indicates the truthfulness of the estimation and a gain matrix $\mathbf{W}^i(k)$, which

is a estimated by using the laser covariance $\mathbf{R}(k)$, the position covariance $\mathbf{P}^i(k)$, and the $\mathbf{H}^i(k)$.

As previously mentioned, the laser error must be inserted, to build the gain Matrix $\mathbf{W}^i(k)$, which generates a diagonal matrix $\mathbf{R}(k)_{n\times n}$ with the typical deviation given by the laser provider: equal to 0.003^2.

$$\mathbf{R}(k) = \begin{bmatrix} 0.003^2 & 0 & \cdots & 0 \\ 0 & 0.003^2 & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots \\ 0 & \cdots & 0 & 0.003^2 \end{bmatrix} \tag{15}$$

This result yields a covariance matrix of the observations that is estimated as:

$$\mathbf{S}^i(k) = \mathbf{H}^i(k) \cdot \mathbf{P}^i(k) \cdot \mathbf{H}^i(k)^T + \mathbf{R}(k) \tag{16}$$

Having the following gain matrix:

$$\mathbf{W}^i(k) = \mathbf{P}^i(k) \cdot \mathbf{H}^i(k)^T \cdot \mathbf{S}^i(k)^{-1} \tag{17}$$

And naming *innovation array* $\boldsymbol{v}^i(k)$, as the difference between measurements obtained by the laser and those obtained by the observation model:

$$\boldsymbol{v}^i(k) = \begin{bmatrix} \rho_1^i(k) \\ \rho_2^i(k) \\ \vdots \\ \rho_n^i(k) \end{bmatrix} - \begin{bmatrix} \hat{\rho}_1^i(k) \\ \hat{\rho}_2^i(k) \\ \vdots \\ \hat{\rho}_n^i(k) \end{bmatrix} \tag{18}$$

The final correction is expressed by:

$$\begin{bmatrix} \tilde{\rho}^i(k) \\ \tilde{\phi}^i(k) \end{bmatrix} = \begin{bmatrix} \tilde{\rho}^i(k) \\ \tilde{\phi}^i(k) \end{bmatrix} + \mathbf{W}^i(k) \cdot \boldsymbol{v}^i(k) \tag{19}$$

This procedure allows to obtain up to 6 landmarks each 10 ms.

IV. SENSORS ERROR ESTIMATION

The measurement error must first be computed, before starting to work with the inner odometry of the robot, the RFIDs, the laser, and the camera. This error, present in every single measurement, is denoted as white noise, and is time variant. The aim of these experiments is to achieve a Gaussian distribution from all sensors for each variable, both in distance and in angle; in other words, the uncertainty of each measurement will be computed.

A. ODOMETRY STANDARD DEVIATION

The robot drive system uses high-speed, high-torque, brushless reversible-DC motors. Each drive motor includes a high-resolution optical quadrature shaft encoder that provides 195 ticks per millimeter of wheel rotation for precise position and speed sensing and advanced dead-reckoning [31]. Although this system might imply reliable robot odometry, it still fails to yield the position estimation error, which has to be used in the Kalman Filter as the process error $Q(k)$.

This error term needs to be estimated, in order to be represented as a covariance matrix, where its principal diagonal

TABLE 2. Process noise standard deviation.

Distance (m)	Angle (rad)
0.029444	0.04019

TABLE 3. Standard deviation of RSSI and RF Phase.

Description	Min	Max	Unit
RSSI		1	dB
RF Phase	-0.1	0.1	rads

are the variances of the error in distance and yaw angle $(heading, \theta)$ of the robot.

$$Q(k) = \begin{pmatrix} \sigma_{dr}^2 & 0 \\ 0 & \sigma_{\theta r}^2 \end{pmatrix} \tag{20}$$

Doris was intended to move forward and backwards in any direction and to spin in circles, to obtain the values of σ_{dr}^2 and $\sigma_{\theta r}^2$. Every position reported by the robotic encoders was registered in a file and the difference between the position provided by the encoders versus the distance and angle measured with external sensors is taken as the error. When the robot is moving forward or backwards it tends to lean to the right, due to wheel pressure differences, adding a bias. One thing to be considered is that the robot receives linear and angular speed commands as inputs and the model in use is referenced in drive and steer increments as inputs, so the robots always moves at the same speed. The results are shown in the table below:

$$Q(k) = \begin{pmatrix} 0.000867 & -0.00049 \\ -0.00049 & 0.001616 \end{pmatrix} \tag{21}$$

B. RFID STANDARD DEVIATION

The RFID system can be powerful for many applications, such as identifying people, logistic and supply chain visibility, race timing, access control, among others, yet it is not very powerful for the purposes of localization. Designing a Real-Time Location System (RTLS) using RFIDs is no easy job. Environment and other effects assume an important place when the reader estimates RSSI and phase angles. Propagation effects such as absorption and scattering can reduce the power observed at the reader receiver, as well as multipath propagation and undesired signals in the environment, as mobile networks and Wi-Fi can combine with primary backscatter, increasing or decreasing the RSSI value.

Phase angle is estimated by using the RSSI as a function of the Signal to Noise Ratio (SNR), so the more noise energy within the receiver bandwidth, the greater the phase standard deviation. In addition, the reader receives signal processing that introduces π radian ambiguity, so that the reported phase can be the true phase (θ) or the true phase plus π radians $(\theta + \pi)$.

According to [27], the standard deviation of the RSSI System is:

These values were obtained from mean value over 1000 tags in an anechoic chamber. However, the robot will not

TABLE 4. Values of RSSI and distance in a complex environment.

Real distance (m)	RSSI 0 rad (dB)	Distance 0 rad (m)	RSSI $\frac{\pi}{4}$ rad (dB)	Distance $\frac{\pi}{4}$ rad (m)
1.00	-37	0.929964	-47.5	1.702025
1.50	-46	1.561227	-53.5	2.404175
2.00	-40	1.965468	-59.5	3.395987
2.50	-43	1.351967	-53	2.335963
3.00	-56	2.776298	-64	4.400135
3.50	-56	2.776298	-61	3.702253
4.00	-63	4.275295	-57	2.940804

TABLE 5. RFID standard deviation.

Distance error @ 0 rad(m)	Distance error @ $\frac{\pi}{4}$ rad (dB)
0.496876	0.892177

move around in anechoic spaces. The typical surroundings will be, for example, in a museum where several people will interact with Doris at different times. Moreover, objects from a museum like sculptures will represent obstacles for the robot, affecting every measurement taken by the RFID system. The following table shows how this system works in an indoor environment where the robot will perform, calculating the distance from antenna to tag by using equation (1). Due phase is a periodical function and given the great difficulty in determining the distance from tag to reader by using only one phase, the phase equation method was discarded:

Table 4 shows the real distance from the antenna to the tag in the first column, the reported RSSI with the tag in front of the antenna in the second column, where the phase is around 0 *rads*, and the RSSI value converted to distance in the third column. Columns 4 and 5 are the same as the two previous ones, but with the tag placed at 45 degrees from the antenna. The results show that if the tag is placed right in front of the antenna (Line of Sight), then the distance will be more feasible, but with a small deviation from the center, the reader will report a higher RSSI and the distance will be erroneous.

As can be seen in previous table, the error is too high, added to which the EKF will lead to undesired results and the phase will be useless, because it will depend on the RSSI value. Hence, the RFID system will be excluded from the localization system and will not be used for future implementations.

C. LASER STANDARD DEVIATION

In section III-C, the operation of the laser range device was briefly explained. However, its behavior tends to add uncertainty to the measurement, for which reason a procedure was followed to obtain the mean and variance values for distance and angular uncertainty.

1) The robot was placed in an area where the landmarks can be freely moved.
2) The area is approximately $1.6m$ wide and $4.0m$ heigh and was divided into small areas of $0.16m^2$
3) A measurement was taken in four different orientations, in order to estimate the distance and angle and to compare them with the real measurements

TABLE 6. Reflective landmark real position vs estimated position.

Real distance (m)	Real angle (rad)	Estimated distance (m)	Estimated angle (rad)
0.8944	0.4636	0.8957	0.4916
0.8000	0.0000	0.7948	-0.0038
0.8944	-0.4636	0.8844	-0.4886
1.2649	0.3217	1.2656	0.3272
1.2000	0.0000	1.1872	0.0002

TABLE 7. Reflective landmark errors at different distances.

Error distance (m)	Error angle (rad)
-0.0013	-0.0279
0.0051	0.0038
0.0099	0.0250
-0.0007	-0.0054
0.0127	-0.0002

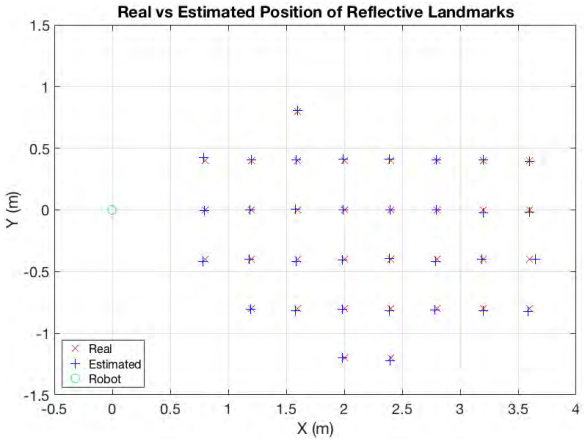

FIGURE 10. Real vs estimated position of reflective landmarks.

4) This procedure was repeated 25 times for each orientation.

A sample iteration is shown in the following table:

where, columns 1 and 2 are the real distance and angle from the sensor to the landmark, columns 3 and 4 are the estimated distance and angle provided by the laser. In table 7 columns 1 and 2 are the error between the real and the estimated values. In figure 10 can be seen the previous table. This experiment was performed up to $5m$ in x-coordinate and $5m$ in y-coordinate, but only few results are shown in the tables and the respective figure.

Where the green 'o' is the location of the robot, the red 'x' represent the real position of the landmark (columns 1 and 2 of table 6) and the blue '+' the position estimated and fixed by the detector explained in section III-C (columns 3 and 4 of table 6).

as can be seen in table 7, the error is a random value. The standard deviations for the resultant distance and angle are shown in table 8:

$$R(k+1) = \begin{pmatrix} 0.000083 & 0 \\ 0 & 0.000027 \end{pmatrix} \quad (22)$$

TABLE 8. Reflective landmark standard deviation.

Distance (m)	Angle (rad)
0.00911	0.005196

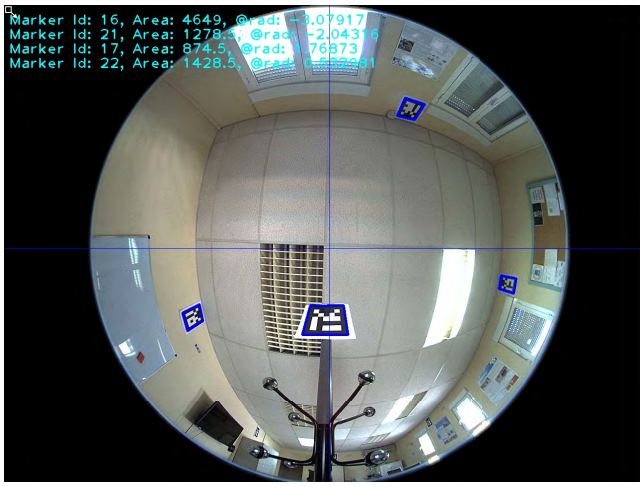

FIGURE 11. Visual marker placed at zero degrees.

TABLE 9. Angle error at different distances.

Distance (cm)	0 (rad)	$\frac{\pi}{2}$ (rad)	π (rad)	$\frac{3\pi}{2}$ (rad)
20	0.06088			
40	0.06776	0.03297		
60	0.05457	0.03258	0.07663	0.13908
80	0.05994	0.03031	0.07577	0.10758
100	0.0595	0.02915	0.05832	0.09835
120	0.06258	0.03183	0.07348	0.07687
140	0.06045	0.03431	0.06854	0.08102
160	0.06138	0.04255	0.05873	0.07578
180	0.06005	0.04744	0.06511	0.06926
200	0.05909	0.04658	0.06011	0.06911

The covariance between laser distance and angle is around 10^{-7}, which is practically negligible.

D. CAMERA STANDARD DEVIATION

Using a similar procedure as in IV-C, a marker that is $2.5m$ above ground level (shown in figure 11) was placed at different positions and distances from the robot. The error of different angles from the camera at different distances is shown in the following table.

As can be seen in table 9, the error is higher when the landmark is closer to the camera, which makes sense because the closest marker to the camera has more pixel variations. The standard deviations for each angle are shown in table 10 where the highest deviation for the EKF filter is selected by the following equation:

$$R(k+1) = \left(0.00045 rad^2\right) \tag{23}$$

V. KALMAN FILTER AND INTEGRATION

Each sensor described above appears to work quite well alone, but one of the main purposes of this research is to ensure that they all work in synergy with the odometry of the robot by means of sensor fusion. One way to achieve this aim is by using the Kalman Filter. As stated, the position of the state vector of the robot is defined as:

TABLE 10. Angle standard deviation.

0 (rad)	$\frac{\pi}{2}$ (rad)	π (rad)	$\frac{3\pi}{2}$ (rad)
0.00339	0.00954	0.00740	0.02122

$$\mathbf{x}(k) = \begin{bmatrix} x(k) \\ y(k) \\ \theta(k) \end{bmatrix} \tag{24}$$

The main idea is to minimize the odometrical error that can be caused by cumulative errors in encoders, slippery wheels, etc. An adaptation of the Kalman Filter can achieve this goal.

A. KALMAN FILTER

Proposed by [32], the filter consists of a set of mathematical equations to arrive at an effective solution for the least squares problem, making it robust and powerful, due to the fact that it is based on past, present, and future data. It is summarized in the form of a recursive least square algorithm for dynamic systems.

The algorithm approaches the problem for state estimation of a discrete lineal process through data sets which can have noisy values. Based on this filter, other versions for nonlinear systems are the Unscented Kalman Filter (UKF) and the Extended Kalman Filter (EKF). The latter has been chosen to be used in this research due to the computational cost of the UKF. Although it is better for strong nonlinearities, previous works in robot localization showed similar results in practice [33], [34]. Regarding the use of the EKF instead of a particle filter, the main reason for this choice is that the designed landmarks are highly distinctive and data association errors are not very likely.

The first step for the EKF is the prediction model:

$$\mathbf{x}(k+1|k) = \mathbf{f}(k, \mathbf{x}(k|k), \mathbf{u}(k) + \mathbf{v}(k)) \tag{25}$$

$$\tilde{\mathbf{x}}(k+1|k) = \mathbf{f}\left(k, \tilde{\mathbf{x}}(k|k), \mathbf{u}(k)\right) \tag{26}$$

Based on this equation, Doris uses the following values for the prediction:

$$\begin{bmatrix} \tilde{x}(k+1|k) \\ \tilde{y}(k+1|k) \\ \tilde{\theta}(k+1|k) \end{bmatrix} = \begin{bmatrix} \tilde{x}(k|k) \\ \tilde{y}(k|k) \\ \tilde{\theta}(k|k) \end{bmatrix} + \begin{bmatrix} \Delta\rho(k) \cos\left(\tilde{\theta}(k|k) + \frac{\Delta\phi(k)}{2}\right) \\ \Delta\rho(k) \sin\left(\tilde{\theta}(k|k) + \frac{\Delta\phi(k)}{2}\right) \\ \Delta\phi(k) \end{bmatrix} \tag{27}$$

where, $[\Delta\rho(k) \;\; \Delta\phi(k)]^T$ is the system input defined as the drive $\Delta\rho(k)$ and the incremental values of the steering

encoders $\Delta\phi\,(k)$. And for upgrading the position uncertainty, the following is used:

$$\mathbf{P}\,(k+1|k) = \mathbf{A}\,(k)\,\mathbf{P}\,(k|k)\,\mathbf{A}^T\,(k) + \mathbf{G}\,(k)\,\mathbf{Q}\,(k)\,\mathbf{G}^T\,(k) \tag{28}$$

where, $\mathbf{P}\,(k|k)$ is the current state covariance matrix and $\mathbf{Q}\,(k)$ is the process covariance matrix. $\mathbf{A}\,(k)$ and $\mathbf{G}\,(k)$ are the Jacobian matrices that are computed as:

$$\mathbf{A}\,(k) = \nabla\tilde{\mathbf{f}}_{\tilde{\mathbf{x}}(k|k)}; \qquad \mathbf{G}\,(k) = \nabla\tilde{\mathbf{f}}_{\tilde{\mathbf{u}}(k)} \tag{29}$$

yielding (30) and (31), as shown at the top of the next page:

The prediction of the observations and their covariance matrix have to be computed:

$$\tilde{\mathbf{z}}\,(k+1) = \mathbf{h}\left(k+1, \tilde{\mathbf{x}}\,(k+1|k)\right) \tag{32}$$

$$\mathbf{S}\,(k+1) = \mathbf{H}\,(k+1)\,\mathbf{P}\,(k+1|k)\,\mathbf{H}^T\,(k+1) + \mathbf{R}\,(k+1) \tag{33}$$

where, $\mathbf{R}\,(k+1)$ is the measurement covariance matrix, where every landmark is inputted with its Cartesian coordinates in a database:

$$\mathbf{x}_L = \begin{bmatrix} x_L \\ y_L \end{bmatrix} \tag{34}$$

so the observation is the relative position of the landmark with respect to the robot, in other words, the model used in this case is:

$$\begin{aligned} &\tilde{\mathbf{z}}\,(k+1) \\ &= \begin{bmatrix} \rho\,(k+1) \\ \phi\,(k+1) \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{(x_L - \tilde{x}\,(k+1|k))^2 + (y_L - \tilde{y}\,(k+1|k))^2} \\ tan^{-1}\left(\dfrac{y_L - \tilde{y}\,(k+1|k)}{x_L - \tilde{x}\,(k+1|k)}\right) - \tilde{\theta}\,(k+1|k) \end{bmatrix} \end{aligned} \tag{35}$$

the matrix $\mathbf{H}\,(k+1)$ is the Jacobian matrix computed as follows:

$$\mathbf{H}\,(k+1) = \nabla\mathbf{h}_{\tilde{\mathbf{x}}(k+1|k)} \tag{36}$$

which is computed for each landmark in the database, resulting in (37), as shown at the top of the next page:

And the $\mathbf{W}\,(k+1)$ which is the gain of the Kalman Filter, is computed as:

$$\mathbf{W}\,(k+1) = \mathbf{P}\,(k+1|k)\,\mathbf{H}^T\,(k+1)\,\mathbf{S}^{-1}\,(k+1) \tag{38}$$

The next step is to obtain every **observation** from every sensor:

$$\mathbf{z}\,(k+1) \tag{39}$$

Then the **matching** between every observation is performed. For reflective landmarks, an Euclidean distance between the polar coordinates of each landmark stored in the database $\mathbf{p}_1^i = \left(\rho_L^i, \phi_L^i\right)$ and the observation $\mathbf{p}_2 = (\rho_z, \phi_z)$ is computed by using:

$$\mathbf{d}^i\,(\mathbf{p}_1, \mathbf{p}_2) = \sqrt{\rho_L^2 + \rho_z^2 - 2 * \rho_L * \rho_z \cos\,(\phi_L - \phi_z)} \tag{40}$$

where a distances set is obtained and the smallest distance of that set is chosen as the matching landmark observation.

With regard to the visual markers, it is easier to perform the matching, because it is a QR code containing information. As it is a 5×5 matrix, the first and the last row are used for orientation, and the 3 middle rows are used to identify the map (2^{nd} row), the sector (3^{rd} row), and the landmark (4^{th} row). The first two rows will be used in the future for general localization, not implemented in this research, but the other row of the visual marker will be used to perform the matching between the observation and the database.

Once the landmarks have been matched, the measurement innovation is obtained:

$$\boldsymbol{\nu}\,(k+1) = \mathbf{z}\,(k+1) - \tilde{\mathbf{z}}\,(k+1) \tag{41}$$

but this innovation can be deleted, if the landmark does not satisfy the Mahalanobis distance acceptance test, computed as:

$$\mathbf{d}_m^i = \frac{\boldsymbol{\nu}\,(k+1)}{\sigma_s} \tag{42}$$

where, σ_s is the standard deviation of the sensor. If the $\mathbf{d}_m^i$ is lower than a maximum Mahalanobis distance, then the measurement innovation is accepted, otherwise it is rejected.

The final position corrected by the EKF is computed as:

$$\tilde{\mathbf{x}}\,(k+1|k+1) = \tilde{\mathbf{x}}\,(k+1|k) + \mathbf{W}\,(k+1)\,\boldsymbol{\nu}\,(k+1) \tag{43}$$

And the uncertainty $\mathbf{P}\,(k+1|k+1)$ is calculated as:

$$\mathbf{P}\,(k+1|k+1) = [I - \mathbf{W}\,(k+1)\,\mathbf{H}\,(k+1)]\,\mathbf{P}\,(k+1|k) \tag{44}$$

B. SEMANTIC MAP DEFINITION

The world where Doris is going to perform actions needs to be represented. An XML file is a common solution for world description. This world is represented by some important items as features (tables, chairs, doors, chargers, objects...), sites where the robot needs to go to, and the ways to arrive to those sites and the landmarks located on each room which help Doris with locatilization.

In listing 1 there is sample of one of the locations where doris is intended to move around, remarking the set of landmarks, features, sites and ways. This map is built with a priori knowledge of the user.

Features, sites and landmarks locations are referenced in the map. Sites have a description indicating if the site is connected to a "DOOR", belongs to a "WAY" or the site is connected with an important item as a "FRAME". Also the landmarks contain the type of beacon placed on the room, indicating if it is "laser" or "camera" type.

C. INTEGRATING SENSORS

At this point, all sensors must be fused into the EKF, in order for the odometrical system to achieve better positioning of the robot. This operation is achieved by modifying some equations in the filter where sensor fusion takes place. In one

$$\mathbf{A}(k) = \begin{bmatrix} 1 & 0 & -\Delta\rho(k)\sin\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) \\ 0 & 1 & \Delta\rho(k)\cos\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) \\ 0 & 0 & 1 \end{bmatrix} \tag{30}$$

$$\mathbf{G}(k) = \begin{bmatrix} \cos\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) & -0.5 * \Delta\rho(k)\sin\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) \\ \sin\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) & 0.5 * \Delta\rho(k)\cos\left(\theta(k|k) + \dfrac{\Delta\phi(k)}{2}\right) \\ 0 & 1 \end{bmatrix} \tag{31}$$

$$\mathbf{H}(k) = \begin{bmatrix} -\dfrac{x_L - \tilde{x}(k+1|k)}{\sqrt{(x_L - \tilde{x}(k+1|k))^2 + (y_L - \tilde{y}(k+1|k))^2}} & -\dfrac{y_L - \tilde{y}(k+1|k)}{\sqrt{(x_L - \tilde{x}(k+1|k))^2 + (y_L - \tilde{y}(k+1|k))^2}} & 0 \\ \dfrac{y_L - \tilde{y}(k+1|k)}{(x_L - \tilde{x}(k+1|k))^2 + (y_L - \tilde{y}(k+1|k))^2} & -\dfrac{x_L - \tilde{x}(k+1|k)}{(x_L - \tilde{x}(k+1|k))^2 + (y_L - \tilde{y}(k+1|k))^2} & -1 \end{bmatrix} \tag{37}$$

of them, (35), every observation must be computed and every landmark position must be read from the database and referenced to the robot position. This operation will result in a matrix $2 * rl + vm$, where rl is the number of reflective landmarks and vm is the number of visual markers in the room, which is the result of the information given by each sensor (distance and angle to the landmark by the laser and angle to the marker by the camera).

The sensor type is very important, as it will set the order in which each sensor is taken into account in the filter. The main reason is to set an order for all matrices. The established order is:

1) Laser Range Finder (rl).
2) Omnidirectional Camera (vm).

Example: Suppose the robot in a certain position has the two sensors enabled and the room where it is located has three reflective landmarks and two visual markers. The predicted position is:

$$\tilde{\mathcal{X}}(k+1|k) = \begin{bmatrix} 5.2152 \\ 2.3489 \\ 3.1054 \end{bmatrix} \tag{45}$$

and the observation matrix $\tilde{\mathbf{z}}(k+1)$ will be:

$$\tilde{\mathbf{z}}(k+1) = \begin{bmatrix} \rho_{rl1} \\ \phi_{rl1} \\ \rho_{rl2} \\ \phi_{rl2} \\ \rho_{rl3} \\ \phi_{rl3} \\ \hline \phi_{vm1} \\ \phi_{vm2} \end{bmatrix} = \begin{bmatrix} 2.9530 \\ 0.6492 \\ 1.5731 \\ 0.7242 \\ 0.8512 \\ -1.5767 \\ \hline 0.0793 \\ -1.0485 \end{bmatrix} \tag{46}$$

The camera is not located at the exact center for the robot, so the deviation must be added to the observation, for it to be correctly computed.

Once the order in the matrices is defined, the other equation which is affected by the sensor fusion is (33); the $\mathbf{R}(k+1)$ is a square diagonal matrix, with the covariance values of every sensor that is enabled. This will result in the following covariance matrix (47), as shown at the bottom of the next page: where, the values of $\sigma^2_{\rho rl}$, $\sigma^2_{\phi rl}$ and $\sigma^2_{\phi vm}$ are constant and were obtained along with each error term in a large-scale trial.

And, finally, the matching step plays an important role, because the vector from equation (41) can be correctly built according to the match. The Mahalanobis distance estimates achieved correct matching. Although it was used for every sensor, the matching for the visual markers can be easily performed by matching the visual code identified by the camera.

VI. TESTS AND RESULTS

Some tests were performed after programming the filter and the detectors in each task. These tests were performed in a sector located at the Higher Technical School of Industrial Engineering in Madrid as depicted in figure 12. This operational area has an approximate width and height of $25.66m \times 5.85m$, with a hallway that accesses other rooms that it is about $1.73m$ high.

For the initial position, the robot was placed near one door of the hallway, at the position $[1.4m, 3.9m, 0rad]^T$, and the covariance matrix for the initial position $\mathbf{P}(0)$ was:

$$\mathbf{P}(0) = \begin{bmatrix} 0.000567 & 0 & 0 \\ 0 & 0.000567 & 0 \\ 0 & 0 & 0.010417 \end{bmatrix} \tag{48}$$

These covariances matrices, $\mathbf{P}(0)$ and $\mathbf{Q}(k+1)$, should adapted to reality, otherwise the EKF will take a higher amount of time to converge.

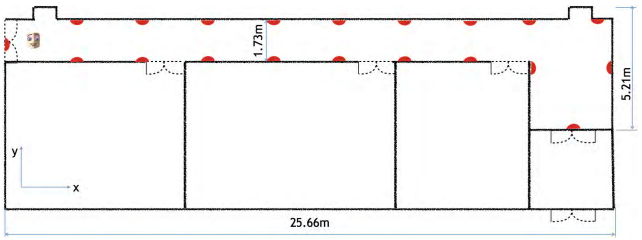

FIGURE 12. Distribution of visual and reflective landmarks.

A. LASER TESTS

Landmarks were placed in the hallway which was used for the experiment, at a distance of 275 *cm* from each other. The reason for this distance was due to the classification landmarks algorithm. If two consecutive landmarks are too close, this algorithm might recognize and classify both as one single landmark. Neither can the landmarks be too far from the LRF, otherwise it will not perceive them, and although the laser can reach up to 32*m*, reflectiveness is lost over distances. Figure 12 represents how the reflective landmarks are distributed along the corridor.

After several tests developed in a narrow hallway with this configuration of landmarks, so that the filter can converge with only the laser readings the following covariance matrix $\mathbf{R}(k+1)$ might represent a good approximation:

$$\mathbf{R}(k+1) = \begin{bmatrix} 0.000083 & 0 & 0 & \cdots & 0 \\ 0 & 0.000027 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0.000083 & 0 \\ 0 & \cdots & 0 & 0 & 0.000027 \end{bmatrix} \quad (49)$$

Then, using this variance for each measure, the Mahalanobis distance will reject all those measures which are over $0.02m$ for the distance and $0.01rad$ in case of the angle. In case both measures given by laser are above this tolerance threshold, the measures will be rejected and the filter will not be used for correction.

The operation of the filer is shown in figure 13. The behavior of the EKF shows how it has converged. One line represents the real odometry provided by the robot, and the other line represents the real pose of the robot provided by the EKF. Figure 14 shows the evolution of the position covariance matrix at each step, where each line represents

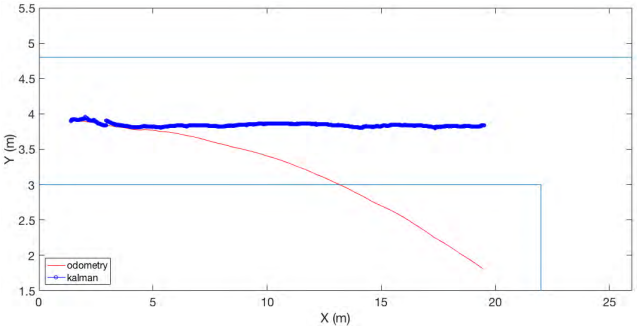

FIGURE 13. Odometrical position of the robot vs. EKF using only LMS-200.

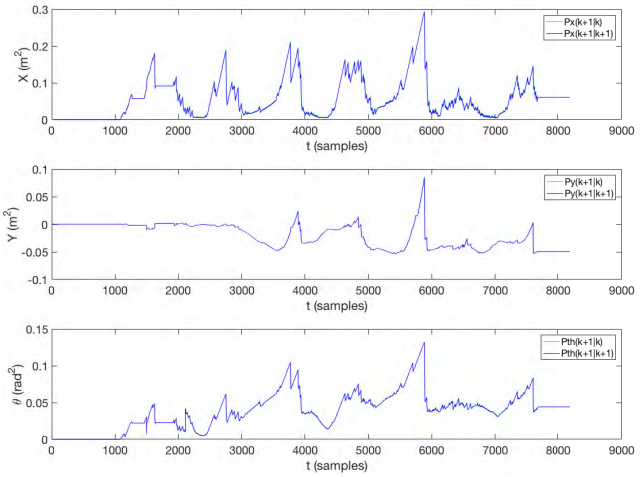

FIGURE 14. Estimated position covariance of the robot using only LMS-200.

the evolution of the prediction $\mathbf{P}(k+1|k)$and the correction $\mathbf{P}(k+1|k+1)$. It can be seen that there are some points where the filter increments the uncertainty, which happens because the observations are provided from one to zero landmarks that are not accurate enough for the EKF, while the robot keeps moving forward. Despite this inconvenience, the filter manages to converge rapidly when a new landmark is seen.

Supposing that the landmarks might not be seen at any one time, in crowded places where people are walking around the robot, the laser will not be very useful. Hence the need to add a further sensor that can register other landmarks at all times, which explains the need for a camera.

$$\mathbf{R}(k+1) = \left[\begin{array}{cccccc|cc} \sigma^2_{\rho rl} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{\phi rl} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{\rho rl} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{\phi rl} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{\rho rl} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^2_{\phi rl} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{\phi vm} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma^2_{\phi vm} \end{array}\right] \quad (47)$$

```
<sector id="1" name="Computer Vision" width="1030" height="1146" reference="2" polygon="m 0 0 l 0 1146 l 1030 0 l 0 −1146 z" adjacency="0,2,7,8">
  <landmarks>
    <landmark id="0" type="laser" x="33" y="30" z="0"/>
    <landmark id="1" type="laser" x="52" y="230" z="0"/>
    <landmark id="2" type="laser" x="200" y="110" z="0"/>
    <landmark id="3" type="laser" x="200" y="390" z="0"/>
    <landmark id="4" type="laser" x="360" y="110" z="0"/>
    <landmark id="11" type="camera" x="0" y="713" z="275"/>
    <landmark id="12" type="camera" x="0" y="985" z="275"/>
    <landmark id="0" type="camera" x="90" y="1146" z="275"/>
    <landmark id="1" type="camera" x="464" y="1146" z="275"/>
  </landmarks>
  <features>
    <feature id="0" name="armario 1" width="60" height="670" x="0" y="300"/>
    <feature id="1" name="mesa 1" width="80" height="179" x="160" y="778"/>
    <feature id="2" name="mesa 2" width="80" height="179" x="160" y="957"/>
    <feature id="3" name="mesa 3" width="80" height="179" x="323" y="778"/>
    <feature id="4" name="door" width="137" height="10" x="10" y="1146"/>
    <feature id="5" name="door" width="137" height="10" x="80" y="0"/>
    <feature id="6" name="door" width="137" height="10" x="0" y="65"/>
  </features>
  <sites sequence="5,3,2,0">
    <site id="0" name="door" x="115" y="1174" linked−sector−id="0" xcoord="18" ycoord="−1146"/>
    <site id="1" name="way" x="115" y="713"/>
    <site id="2" name="way" x="464" y="713"/>
    <site id="3" name="frame" x="115" y="209"/>
    <site id="4" name="door" x="0" y="209" linked−sector−id="7" xcoord="−18" ycoord="1180"/>
    <site id="5" name="way" x="115" y="80"/>
    <site id="6" name="door" x="0" y="80" linked−sector−id="8" xcoord="−18" ycoord="1180"/>
    <site id="7" name="door" x="115" y="0" linked−sector−id="2" xcoord="0" ycoord="1059"/>
  </sites>
  <ways>
    <way st="0" adjacency="1"/>
    <way st="1" adjacency="0,2,3"/>
    <way st="2" adjacency="1"/>
    <way st="3" adjacency="1,4,5"/>
    <way st="4" adjacency="3"/>
    <way st="5" adjacency="3,6,7"/>
    <way st="6" adjacency="5"/>
    <way st="7" adjacency="5"/>
  </ways>
</sector>
```

Listing 1. Semantic map sample

B. CAMERA TESTS

Visual markers were placed at the same position above the reflective landmarks, in order to compare the results provided by the EKF working only with laser LMS-200 in section VI-A. On this occasion, the EKF was only functioning with the camera. This sensor had to be connected to another computer in order for the image processing to function as quickly as the laser.

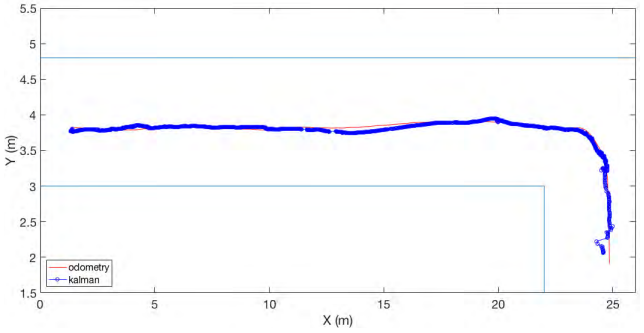

FIGURE 15. Odometrical position of the robot v. EKF using only the Mobotix Camera.

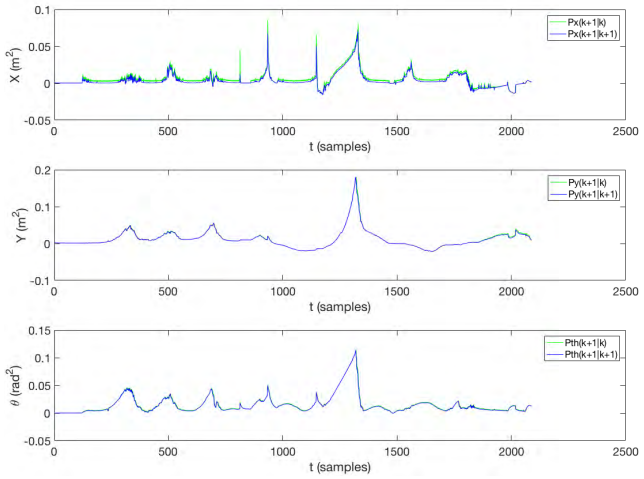

FIGURE 16. Estimated position covariance of the robot using only the Mobotix Camera.

Several tests were performed with these markers and the following covariance matrix $\mathbf{R}(k+1)$ represents a good approximation of the filter only converging with the camera:

$$\mathbf{R}(k+1) = \begin{bmatrix} 0.000045 & 0 & 0 & \cdots & 0 \\ 0 & 0.000045 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0.000045 & 0 \\ 0 & \cdots & 0 & 0 & 0.000045 \end{bmatrix} \quad (50)$$

Comparing the covariances matrices (49) of the laser and (50) the camera, it can be seen that the variance error of the camera angle was lower than the variance error of the laser angle. This arrangement yielded a better EKF performance that may be seen in figure 15.

Comparing the results obtained in section VI-A, it can be seen that the camera, which has minor marker estimation errors and a 360 degree field of view, can register more

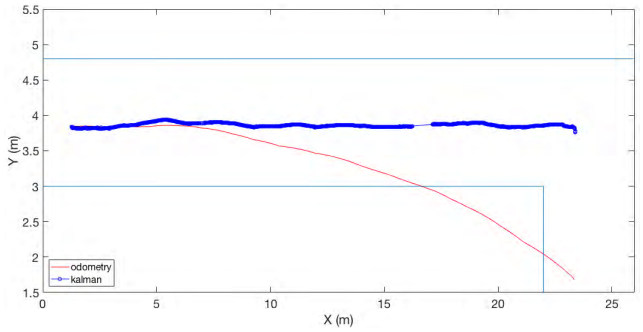

FIGURE 17. Odometrical position of the robot versus EKF using both LMS-200 and the Mobotix Camera.

markers than the laser sensor. While the laser can see at least 2 landmarks, the camera can see 5 or more markers.

When peaks appear in the uncertainty graphs in figure 16, it means that no marker is received in the CPU of the robot, due to communication problems (cable, wifi, etc) or a malfunction of the detector where there is too much light, resulting in noise for the adaptive threshold. The most notorious example is the peak between the samples 1000 and 1500, where the robot did not received any visual landmark in a long period.

C. LASER AND CAMERA

Figure 17 shows how the EKF works much better when both sensors are combined. The uncertainty achieved with both sensors is lower than the uncertainty achieved with only the laser or only the camera. The uncertainty achieved by each sensor is shown in figures 14 and 16, where the laser sensor showed a mean uncertainty in positions at around $0.06m^2$ and at an orientation of 0.04 rad^2, and the camera presented a mean uncertainty at a position of $0.004m^2$ and at an orientation of $0.01rad^2$, underlining the superior performance of the camera compared to the laser. A combination of the results of both sensors in figure 18 showed a mean uncertainty in positions around $0.02m^2$ and at an orientation of $0.02rad^2$. Comparing the results, better operational performance was achieved with only the camera, although the laser can help at certain times, when visual markers are unavailable. For this reason the robot will work with both sensors for positioning.

D. ANOTHER EXPERIMENT WITH PEOPLE MOVING AROUND

A new experiment with obstacles and people passing by was developed in the Center of Automation and Robotics located at Universidad Politécnica de Madrid. The trajectory is shown in figure 19.

The robot goes from the charging station located at $[2m\ 16.7m\ -1.57rad]^T$, it moves backwards, then spins 90 degrees and keeps moving backwards to the point $[3.65m\ 17.6m\ -3.14rad]^T$. A reactive controller is used to follow the trajectory while people is passing around the robot. This trajectory consists on a set of three points that the robot must pass through. The controller moves the robot from the current position to the point $[1.25m\ 17.6m\ -3.14rad]^T$ and then the controller tries to keep the robot in a straight line until it reaches the second point $[1.25m\ 11.0m\ -1.57rad]^T$ and then the controller moves the robot up to the final point $[1.0m\ 2.3m\ -1.57rad]^T$.

In the same figure, the boxes represent objects placed along the environment where the tests are being performed.

This experiment has important issues to highlight. The first one is the landmarks visibility. In the largest area there is no

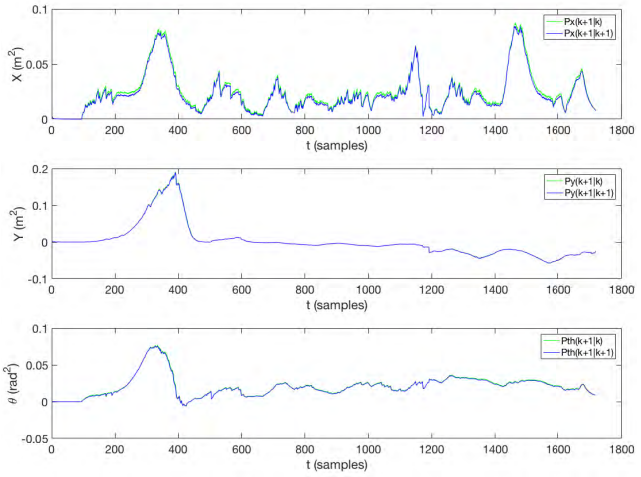

FIGURE 18. Estimated position covariance using both LMS-200 and the Mobotix Camera.

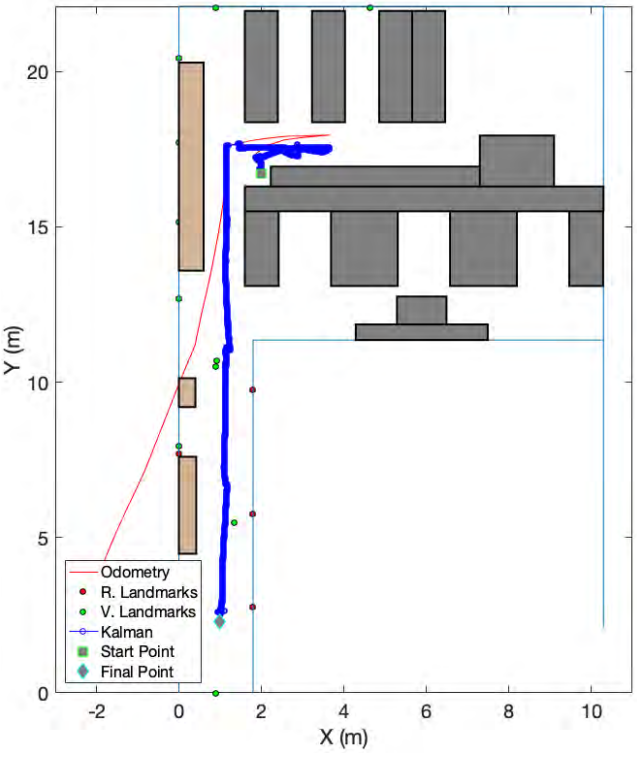

FIGURE 19. Odometrical vs EKF trajectory of the robot using both sensors.

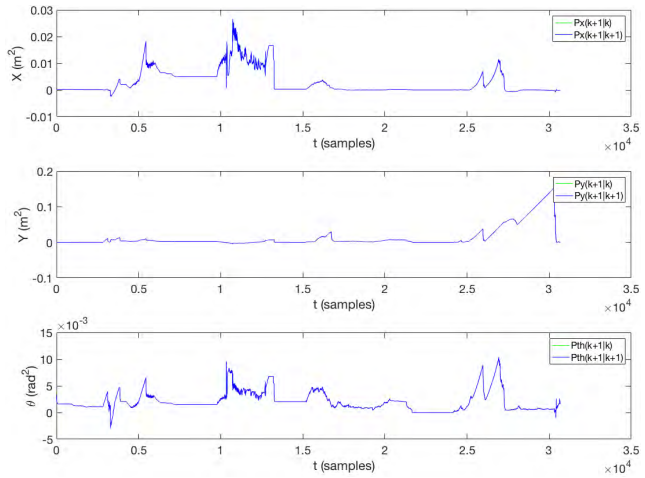

FIGURE 20. Odometrical covariance position using both sensors.

TABLE 11. Ground truth Robot position vs. Real Position.

X_{real} (m)	Y_{real} (m)	θ_{real} (rad)	X_{est} (m)	Y_{est} (m)	θ_{est} (rad)
2	16.7	-1.57	1.9469	16.6882	-1.5571
3.65	17.60	-3.14	3.6231	17.4974	-3.0939
1.25	17.60	-3.14	1.3018	17.5492	-3.1081
1.25	11.00	-1.57	1.2396	11.0382	-1.4213
1.05	10.49	-1.57	1.1407	10.5053	-1.5887
1.00	2.30	-1.57	1.0140	2.5391	-1.5149

space enough to put some reflective landmarks on the ground, so the first part of the experiment only uses seven visual markers placed on the walls. Then in the hallway located next, there are fewer objects placed on the ground, therefore, five reflective landmarks and five visual markers can be placed. The second issue is that there are static objects and people walking in front of the robot.

The evolution of the covariance matrix of the position is shown in figure 20 and the benefits of having both types of landmarks instead of having only one can be seen. The first 15000 samples shown in the $X(m^2)$ subgraph the variance of the position with the help of the visual markers. A good localization is achieved but the variance of the estimation is increased. On the other hand, the next samples show the variance when the reflective landmarks are detected improving a lot the localization of the robot.

While executing the trajectory, one of the reflective landmarks was suddenly occluded, so the localization was only based on the visual landmarks, as can be seen in the same figure from the sample 25000, where the variances of the state is increasing until the last point is reached.

Next table shows a comparison between each point given by the trajectory and the position obtained by the EKF:

As it can be observed in the table 11 the first three columns are the points where the robot must pass through, the next three columns are the position provided by the Kalman Filter and in table 12 the columns are the errors between real

TABLE 12. Ground truth error.

Error X (m)	Error Y (m)	Error θ (rad)
0.0531	0.0118	0.0129
0.0269	0.1026	-0.0461
-0.0518	0.0508	-0.0319
0.0104	- 0.0382	-0.1487
-0.0907	-0.0153	0.0189
-0.0140	-0.2391	-0.0551

TABLE 13. Sensor list prices.

Quantity	Description	Price Unit (€)	Subtotal (€)
1	LRF SICK LMS-200	2276	2276
1	Omnidirectional Camera	428	428
1	RFID System	1773.81	1773.81
		Total:	4477.81

and estimated position. Marks were located on the ground to ensure the robot pass over these points. In the fourth row, a significant deviation of the robot can be observed. This is due to people passing in front of the robot and it started to deviate, trying to arrive at the desired position from other point. The same behavior is presented when robot is in the hallway between y-coordinates 6.00 and 8.00 shown in figure 19. It deviates a little but it instantly returns to the trajectory.

VII. CONCLUSIONS

Indoor position estimation is extremely useful for guiding a robot in museums and trade fairs. The robot in this study has shown that its position can be established, especially when interacting with people, by using laser, camera, and RFID sensors. The first step when working with a laser range finder was to build the reflective landmarks; a complicated process requiring expensive materials. The process of finding the best place to locate the reflective landmark was important, so that the classification algorithm could detect different landmarks, despite which it is a good method to achieve localization without mapping the whole environment.

An omnidirectional camera is a very powerful sensor, because it provides a 360 degree field of view of the built environment and it achieves reliable angle estimations. The visual landmarks, however, had to be printed on an A3 size paper, so that it could be easily detected by the camera at long distances $(5-6m)$. An A4 size marker was used at first, achieving only a detection range of $0-3m$. One big disadvantage of working with an omnidirectional camera is that a high resolution image is needed for detection that requires a fast processing speed. At present, Doris is not equipped with sufficient processing capacity and the image processing has to be performed by another computer.

The RFID system cannot be used for localization due to the high uncertainty that it presents. This is essentially caused by the low operating frequency of RFID. Ultra Wide Band systems work at very high frequencies that increase the efficiency of the performance. However, RFID can be used for

other purposes such as general localization, quality navigation or objects which can provide some semantic information.

Regarding the cost of each sensor, the following table portraits a balance of the prices of each one:

A procedure to obtain the standard deviation for each sensor has been presented that processes an initial estimation for each one. Once integrated with the EKF some adjustments have successfully been applied for faster convergence of the EKF.

Experiments demonstrate that in even with obstacles or people passing by, or even the absence of one type of the landmarks, a good localization is achieved. It can be achieved by using only the omnidirectional camera, but can be refined by including the reflective landmarks.

REFERENCES

[1] J.-D. Lee and Z.-Y. Dang, "Dual-sensor fusion for obstacle avoidance in indoor environment," in *Proc. Int. Conf. Adv. Robot. Intell. Syst. (ARIS)*, May 2015, pp. 1–5.

[2] B. Zhou, K. Qian, F. Fang, X. Ma, and X. Dai, "Multi-sensor fusion robust localization for indoor mobile robots based on a set-membership estimator," in *Proc. IEEE Int. Conf. Cyber Technol. Automat. Control Intell. Syst. (CYBER)*, Jun. 2015, pp. 157–162.

[3] Y. Dobrev, S. Flores, and M. Vossiek, "Multi-modal sensor fusion for indoor mobile robot pose estimation," in *Proc. IEEE/ION Position Location Navigat. Symp. (PLANS)*, Apr. 2016, pp. 553–556.

[4] R. G. Yudanto and F. Petré, "Sensor fusion for indoor navigation and tracking of automated guided vehicles," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Banff, AB, Canada, Oct. 2015, pp. 1–8.

[5] D. Jeon, H. Choi, and J. Kim, "UKF data fusion of odometry and magnetic sensor for a precise indoor localization system of an autonomous vehicle," in *Proc. 13th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI)*, Xi'an, China, Aug. 2016, pp. 47–52.

[6] G. Brahmanage and H. Leung, "A kinect-based SLAM in an unknown environment using geometric features," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst.*, Daegu, South Korea, Nov. 2017, pp. 570–575.

[7] J. Biswas and M. Veloso, "Multi-sensor mobile robot localization for diverse environments," in *Robot Soccer World Cup* (Lecture Notes in Computer Science), vol. 8371. Springer, 2013, pp. 468–479.

[8] A. Burguera, Y. González, and G. Oliver, "Sonar sensor models and their application to mobile robot localization," *Sensors*, vol. 9, no. 12, pp. 10217–10243, Dec. 2009.

[9] R. Thrapp, C. Westbrook, and D. Subramanian, "Robust localization algorithms for an autonomous campus tour guide," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2001, pp. 2065–2071.

[10] D. Rodriguez-Losada, F. Matia, A. Jimenez, and R. Galán, "Local maps fusion for real time multirobot indoor simultaneous localization and mapping," *J. Field Robot.*, vol. 23, no. 5, pp. 209–291, 2006.

[11] D. Rodríguez-Losada, F. Matía, and R. Galán, "Building geometric feature based maps for indoor service robots," *Robot. Auton. Syst.*, vol. 54, no. 7, pp. 546–558, Jul. 2006.

[12] D. Rodríguez-Losada, P. S. Segundo, F. Matía, and L. Pedraza, "Dual fastSLAM: Dual factorization of the particle filter based solution of the simultaneous localization and mapping problem," *J. Intell. Robot. Syst.*, vol. 55, nos. 2–3, pp. 109–134, Jul. 2009.

[13] L. Pedraza, D. Rodríguez-Losada, F. Matía, G. Dissanayake, and J. Valls, "Extending the limits of feature-based SLAM with B-splines," *IEEE Trans. Robot.*, vol. 25, no. 2, pp. 353–366, Apr. 2009.

[14] P. de la Puente and D. Rodriguez-Losada, "Feature based graph-SLAM in structured environments," *Auton. Robots*, vol. 37, no. 3, pp. 243–260, Oct. 2014.

[15] A. Bekkali, H. Sanson, and M. Matsumoto, "RFID indoor positioning based on probabilistic RFID Map and Kalman filtering," in *Proc. 3rd IEEE Int. Conf. Wireless Mobile Comput. Netw. Commun. (WiMob)*, Oct. 2007, p. 21.

[16] Y. Zhang, M. G. Amin, and S. Kaushik, "Localization and tracking of passive RFID tags based on direction estimation," *Int. J. Antennas Propag.*, vol. 2007, Oct. 2007, Art. no. 17426.

[17] S. S. Saab and Z. S. Nakad, "A standalone RFID indoor positioning system using passive tags," *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 1961–1970, May 2011.

[18] B.-S. Choi, J.-W. Lee, J.-J. Lee, and K.-T. Park, "A hierarchical algorithm for indoor mobile robot localization using RFID sensor fusion," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2226–2235, Jun. 2011.

[19] A. Whitney, J. Parker, Z. Kratzer, Z. Kratzer, and J. Fessler, "RSSI informed phase method for distance calculations," in *Proc. IEEE Int. Conf. Acanced Intell. Mechatron. (AIM)*, Busan, South Korea, Jul. 2011, pp. 1138–1142.

[20] X. Song, X. Li, W. Tang, W. Zhang, and B. Li, "A hybrid positioning strategy for vehicles in a tunnel based on RFID and in-vehicle sensors," *Sensors*, vol. 14, no. 12, pp. 23095–23118, Dec. 2014.

[21] Y. Fu, C. Wang, R. Liu, G. Liang, H. Zhang, and S. U. Rehman, "Moving object localization based on UHF RFID phase and laser clustering," *Sensors*, vol. 18, no. 3, p. 825, Mar. 2018.

[22] B. P.-E. Alvarado Vásquez, F. Matia, and R. Galan, "Improving indoor robots localisation by fusing different sensors," in *Proc. IEEE Int. Conf. Intell. Robots Syst. (IROS)*, 2018.

[23] J. Zhou and J. Shi, "RFID localization algorithms and applications—A review," *J. Intell. Manuf.*, vol. 20, pp. 695–707, Dec. 2009.

[24] A. D. Koutsou *et al.*, "Preliminary localization results with an RFID based indoor guiding system," in *Proc. IEEE Int. Symp. Intell. Signal Process.*, Oct. 2007, pp. 1–6.

[25] J. Wu, "Three-dimensional indoor RFID localization system," Ph.D. dissertation, Univ. Nebraska-Lincoln, Lincoln, NE, USA, 2012.

[26] 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.

[27] *Speedway Revolution Reader Application Note Low Level User Data Support*, Impinj, Seattle, WA, USA, 2013.

[28] G. Cimini, F. Ferracuti, A. Freddi, S. Iarlori, and A. Monteriú, "An inertial and QR code landmarks-based navigation system for impaired wheelchair users," in *Ambient Assisted Living*. Berlin, Germany: Springer-Verlag, 2013.

[29] P. Nazemzadeh, D. Fontanelli, D. Macii, and L. Palopoli, "Indoor localization of mobile robots through QR code detection and dead reckoning data fusion," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 6, pp. 2588–2599, Dec. 2017.

[30] H. Zhang, C. Zhang, W. Yang, and C.-Y. Chen, "Localization and navigation using QR code for mobile robot in indoor environment," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2015, pp. 2501–2506.

[31] *PatrolBot(R) Operations and Technical Manual*, MobileRobots, Amherst, NH, USA, Sep. 2008.

[32] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[33] F. Martinelli, "Robot localization: Comparable performance of EKF and UKF in some interesting indoor settings," in *Proc. 16th Medit. Conf. Control Automat.*, Jun. 2008, pp. 499–504.

[34] L. D'Alfonso, W. Lucia, P. Muraca, and P. Pugliese, "Mobile robot localization via EKF and UKF: A comparison based on real data," *Robot. Auton. Syst.*, vol. 74, pp. 122–127, Dec. 2015.

BIEL PIERO E. ALVARADO VASQUEZ received the graduate degree in computer engineering from the Universidad De Oriente, Venezuela, in 2006, and the M.Sc. degree in automation and robotic from the Universidad Politecnica de Madrid, Spain, in 2014, where he is currently pursuing the Ph.D. degree in automation and robotics by developing a framework for social robots. He worked for five years in PDVSA which is the main oil industry of Venezuela, developing process control applications. He has also worked as a backend developer for other industries, such as FLIR Networked Systems and Auxitec. His main research interests include sensors data procession, robots navigation, human–robot interaction, process control engineering, and fuzzy control.

RUBEN GONZALEZ received a degree in industrial engineering, focused on industrial electronics and automation. He is currently pursuing the master's degree in automation and robotics with the Universidad Politecnica de Madrid. He is interested in computer vision and artificial intelligence for robotic applications.

FERNANDO MATIA is currently a Full Professor with the Universidad Politecnica de Madrid. His teaching activity covers control, robotics, and programming, being the author of three textbooks and having supervised more than 10 final degree and final master theses. His research is focused on fuzzy control and autonomous robots, having being an advisor of eight doctoral theses and the author of many articles in indexed journals, book chapters, and conference papers. He has edited one book on fuzzy control. Additionally, he has participated in many research projects and contracts with companies, and has a national accreditation of 24 research years. Regarding his academic management activities, he has been the Deputy Vice-Director of Studies, the Vice-Director of Quality Management, and the Vice-Director of Studies at the High School of Industrial Engineering, UPM, and he is currently the Director of the Department of Automatic Control, Electrical and Electronics Engineering and Industrial Computing.

PALOMA DE LA PUENTE received the Engineering degree in automatic control and electronics and the Ph.D. degree in robotics and automation from the Universidad Politecnica de Madrid (UPM), in 2007 and 2012, respectively. She had a research stay as a Pre-Doctoral Visitor at Caltech. After finishing her Ph.D., she was a Postdoctoral Researcher at DISAM-UPM and at the ACIN Institute of Automation and Control, Vienna University of Technology. She also had professional experience at Ixion Industry and Aerospace, and she is currently an Assistant Professor at UPM. Her main research interests are navigation, mapping, SLAM, spatial cognition, sensor data processing, human–robot interaction for service robotics, and systems engineering. She has participated in several national and European projects and also in international robotics competitions.