A Bayesian Grid Method PCA-based for Mobile Robots Localization in Unstructured Environments

Fernando Carreira and João M. F. Calado
IDMEC-IST - Technical University of Lisbon
and ADEM-ISEL - Polytechnic Institute of Lisbon
1959-007 Lisboa, Portugal
Email: {fcarreira,jcalado}@dem.isel.pt

Carlos Cardeira
IDMEC-IST
Technical University of Lisbon
1049-001 Lisboa, Portugal
Email: carlos.cardeira@ist.utl.pt

Paulo Oliveira
ISR-IST and IDMEC-IST
Technical University of Lisbon
1049-001 Lisboa, Portugal
Email: p.oliveira@dem.ist.utl.pt

***Abstract*—This paper presents the experimental validation of a new method for mobile robot global self-localization in unstructured environments, i.e. that does not need any beacons or other artifacts structuring the environment. The method resorts to a PCA-based positioning sensor, filtered in a Bayesian probabilistic grid and combined with linear Kalman filters to estimate the global pose of mobile robots. In the implemented system, the information of the environment is captured only with onboard sensors installed in a differential drive robot: encoders, compass, and 2D depth sensor pointed to the ceiling. The use of PCA in a Bayesian probabilistic grid allows to fuse the highly compressed PCA database information, obtained with the low computational effort, in an environment where repetitive scenarios can occur. To avoid the negative impact in the localization estimation caused by the corrupted data existing in the 2D depth sensor, an extension to the classic PCA algorithm is suggested. Thus, the proposed method allows the self-localization of mobile robots in indoor environments with high accuracy and works in a wide range of illumination conditions.**

I. INTRODUCTION

The problem of localization has been a great challenge to the scientific community in the area of mobile robotics; see [12], [3] and the references therein. As happens with persons or animals, for a robot to navigate from a point to another it is of great importance its ability to look at the environment and rapidly answer the following questions: where am I? and what am I facing?

Typically, the localization problem consists in the robot knowing what is its position in a map, based on the knowledge that it can to obtain from the environment, through sensors. However, since usually the robot can to capture similar observations about the environment at different points of the map, the fusion of the sensors with the robot motion model, through a Bayes filter, is very common in localization systems, allowing the global localization of the robot [32], [11].

The Extended Kalman filter (EKF) is one of the most popular Bayes method used in mobile robot localization due its easy implementation [22], [9]. Instead the use of EKF, that is the usual way to apply Kalman filters in mobile robots, in [7] a Linear Parameter Varying (LPV) model of a Dubins car kinematic model being used in a classical Kalman filter (KF) has been developed and experimentally validated. However, as KF (or EKF that is the current practice in mobile robot localization) represents the estimation about the robot position trough a Gaussian function, the probabilistic function is unimodal and, therefore, unable to represent the probability in different localizations at same time [11].

An alternative that allows the tracking of multiple possible localizations is the implementation of a multimodal probabilistic function to represent the probability about the robot localization in the Bayesian filter, using a discretization of the environment in a map. Several systems uses topological information about landmarks with propose of reducing the number of states and, hence, to obtain a more compact representation of the environment [31], [27], [10].

Another approach to discretize the environment with a Bayesian filter is the representation of the robot localization in a probability grid [13], [23]. Although grid maps requires more computational efforts, especially for larger environments, this approach allows better accuracy in mobile robots localization.

In [7], [8], a method for mobile robot localization resorting to Principal Component Analysis (PCA) is implemented allowing the development of a self-localization system with a high compression ration of the environment and low computational efforts [18]. These advantages are due the conversion of the map database into an orthogonal space, allowing to obtain a new database with a high compression ratio, when compared with the amount of captured data, and increase the efficiency of the PCA algorithm. Moreover, the PCA allows to develop localization systems that do not depend on any predefined structure [19], [2], i.e, does not need to identify any specific features about the environment. The PCA is used in [26] for terrain reference navigation of underwater vehicles.

The use of vision systems to obtain information about the environment is very common in robot localization due to the large amount of information that can be extracted from the RGB image [29], [28], [20], [15]. Although the very common solution in the development of the localization systems is to install cameras to look around the mobile robot to obtain its position [30], [15], [19], some robots use a single camera looking upward [17], [14], [33]. The use of vision from the ceiling has the advantage that images can be considered static and without scaling. This solution of a ceiling vision is also successfully implemented in [7], [6], which approach is related with the work presented in this paper.

However, a general handicap of vision sensors is the low robustness working in different environment lighting conditions, decreasing the robustness of the mobile robot localization systems. To avoid this problem, the use of time-to-flight sensors is implemented in same localization systems [21], [1]. Moreover, the time-of-flight has the advantage that allows the capture of depth images with a grid of depth information from the field of view. Recently, due to its low price and a straightforward way to be connected with a computer, the Kinect - device with a RGB and a depth camera developed to video games by PrimeSense and Microsoft - becomes very popular in mobile robotics community, creating some interesting mobile robots applications [4], [16]. Kinect depth images are obtained by a structured-light 3D scanner.

A very common problem in depth sensors, including the Kinect depth sensor, is the existence of missing data in signals, caused by IR beams that are not well reflected, not returning to the depth sensor receiver. In [24], [25], a method using the Principal Component Analysis (PCA) methodology to avoid the problem of missing data in signals is presented and its performance is compared with other state-of-the-art algorithms, concluding that PCA is the algorithm that presents the best results in the corrupted signals recovery process. This method is experimentally validated in self-localization system of mobile robots with corrupted depth images captured by Kinect installed onboard, pointing upwards to the ceiling [6].

In the present paper, an adaptation of the self-localization method proposed in [6], considering a Bayesian filter in the PCA position sensor is implemented in a grid map and experimentally validated.

This paper is organized as follows: Section II presents the mobile robot platform and the motivation for the use of Kinect in the proposed localization system; in the Section III, the proposed method that fuses the principal component analysis algorithm for signals corrupted with missing data in a Bayesian filter to obtain the mobile robot position is detailed. For the method validation purpose, Section IV presents experimental results of the proposed method; Finally, some conclusions and topics about future work are presented in Section V.

II. MODEL PLATFORM

The experimental validation of the positioning system proposed in this paper is performed resorting to a low cost mobile robotic platform [5], with the configuration of a Dubins car. A Microsoft Kinect is installed on the platform, pointing upwards to the ceiling, and a digital compass, located on the extension arm (robot rear part) to avoid the motors magnetic interference (see Fig. 1).

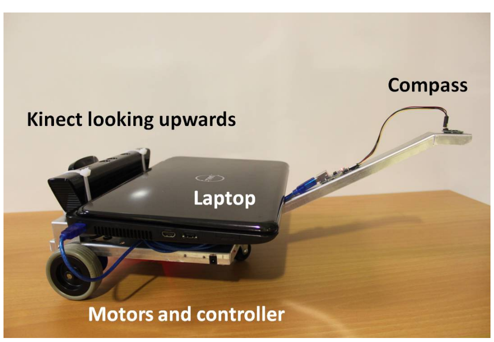

Fig. 1. Mobile platform equipped with kinect sensor and compass

The Kinect includes a RGB camera with a VGA resolution (640×480 pixels) using 8 bits and a 2D depth sensor (640×480 pixels) with 11 bits of resolution. Once the robot moves in an environment indoors in buildings with some information (e.g. building-related systems such as HVAC, electrical and security systems, etc.), it is possible to use the signals captured by a Kinect looking upward (RGB image, depth map or both) by an algorithm that can provide mobile robot global position in the environment.

Due to limitations found in image-based mobile robot localization approaches, regarding illumination changes, and aiming the development of an efficient self-localization solution that can work in places with variation on the level of illumination, only the Kinect depth signal is used, resorting to an adaptation to the method proposed in [26], [8], [7] to the problem at hand.

III. PCA-BASED LOCALIZATION WITH A PROBABILISTIC BAYESIAN GRID

PCA [18] is a methodology based on the Karhunen-Loève (KL) transformation which is often used in applications that need data compression, like image and voice processing, data mining, exploratory data analysis and pattern recognition. The data reduction is obtained through the use of a database eigenspace approximation by the best fit eigenvectors. This technique makes the PCA an algorithm that has a high compression ratio and requires reduced computational resources. The PCA algorithm is successfully used as the mobile robot's position sensor in [6], [8].

A. *PCA for Signals with Missing Data*

In this section, the details about the creation of the PCA eigenspace, compressing the database of the map which captured signals are corrupted with missing data is presented.

The PCA eigenspace is created based on a set of M stochastic signals $\mathbf{x}_i \in \mathbb{R}^N$, $i = 1, \ldots, M$ acquired by a Kinect depth sensor installed onboard the mobile robot, considering an area with N mosaics in two dimensional space, $N = N_x N_y$, where N_x and N_y are the number of mosaics in x and y axis, respectively.

In the common PCA-based approaches, the eigen-space of the set of acquired data is characterized by the corresponding mean $\mathbf{m}_x = \frac{1}{M}\sum_{i=1}^{M}\mathbf{x}_i$ and covariance $\mathbf{R}_{xx} = \frac{1}{M-1}\sum_{i=1}^{M}(\mathbf{x}_i - \mathbf{m}_x)(\mathbf{x}_i - \mathbf{m}_x)^T$. However, the existence of missing data in signal $\mathbf{x}_i$ corrupts the PCA mean value computation creating an orthogonal space with erroneous data.

In the case where missing data occurs, the classical approach to compute the covariance matrix, resorts to a mean substitution operation. Thus, a vector $\mathbf{l}$ with length N consisting of boolean values is used to mark the real and missed data of a signal $\mathbf{x}_i$. Then, considering the j^{th} component of acquired signal $\mathbf{x}_i$, the index $\mathbf{l}_i(j)$ is set to 1 if the signal $\mathbf{x}_i(j)$ is available and it is set to 0 if there is a missing data.

Hence, to avoid the negative impact of the sensor signals missing data in PCA-based approaches performance, an extension to this methodology is proposed in this paper, where instead of considering all values of the M stochastic signals to compute the previously mentioned, only the correct data is used to compute the orthogonal space, characterized by $\mathbf{m}_x$ and $\mathbf{R}_{xx}$, and the value corresponding to missing data is neglected. Thus, the auxiliary counters $\mathbf{c} = \sum_{i=1}^{M}\mathbf{l}_i$ and $\mathbf{C} = \sum_{i=1}^{M}\mathbf{l}_i\mathbf{l}_i^T$ are defined, based on the auxiliary vector $\mathbf{l}$ defined before.

Considering the set with M signals, the mean ensemble for the j^{th} component is given by:

$$\mathbf{m}_x(j) = \frac{1}{c(j)} \sum_{i=1}^{M} \mathbf{l}_i(j)\mathbf{x}_i(j), \;\; j = 1, ..., N \tag{1}$$

and the covariance element $\mathbf{R}_{xx}(j,k), \;\; \{j,k\} = 1, ..., N$ is computed as follows:

$$\mathbf{R}_{xx}(j,k) = \frac{1}{C(j,k)-1} \sum_{i=1}^{M} \mathbf{l}_i(j)\mathbf{l}_i(k)\mathbf{y}_i(j)\mathbf{y}_i(k) \tag{2}$$

where $\mathbf{y}_i = \mathbf{x}_i - \mathbf{m}_x$.

Considering the new mean ensemble and covariance of the PCA database computed without corrupted data in (1) and (2), the decomposition into the orthogonal space follows the PCA algorithm classical approach, i.e. $\mathbf{v} = \mathbf{U}^T(\mathbf{x} - \mathbf{m}_x)$. The matrix $\mathbf{U} = [\mathbf{u}_1 \, \mathbf{u}_2 \, \ldots \, \mathbf{u}_N]$ should be composed by the N orthogonal column vectors of the basis, verifying the eigenvalue problem:

$$\mathbf{R}_{xx}\mathbf{u}_j = \lambda_j \mathbf{u}_j, \;\; j = 1, ..., N, \tag{3}$$

Assuming that the eigenvalues are ordered, i.e. $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$, the choice of the first $n \ll N$ principal components leads to stochastic signals approximation given by the ratio on the covariances associated with the components, i.e. $\sum_n \lambda_n / \sum_N \lambda_N$.

During the mission, the signal $\mathbf{x}$ is decomposed into the orthogonal space considering only the non-corrupted data. Thus, before the projection of the depth image into the orthogonal space, the mean substitution should be followed, i.e, all j^{th} component of the signal $\mathbf{x}_i$ with corrupted data should be replaced by the corresponding mean value $\mathbf{m}_x(j)$. This method removes the effect of the corrupted data in its decomposition in the orthogonal space $\mathbf{v} = \mathbf{U}^T(\mathbf{x} - \mathbf{m}_x)$.

The robot position is obtained by finding in all PCA eigenspace, the mosaic whose eigenvector is nearest to the acquired signal decomposed into the orthogonal space:

$$r_{PCA} = \min_i \;\; \|\mathbf{v} - \mathbf{v}_i\|_2, i = 1, \ldots, N; \tag{4}$$

Given the mosaic i that verifies this condition, its center coordinates $[x_i \, y_i]^T$ are selected as the robot position, obtained by the PCA-based sensor.

Then, the mean substitution approach is used when there is missing data in the depth signals coming from the Kinect sensor. Just like during the creation of the PCA eigenspace, it must be done before the application of the PCA algorithm, i.e, all j^{th} component of the signal $\mathbf{x}_i$ should be replaced by the corresponding mean value $\mathbf{m}_x(j)$.

However, when the robot moves inside a large area or in scenery with periodic information, the information becomes often ambiguous, resulting frequently in wrong values and consequently, the robot can quickly lose its position. This problem occurs because the use of the PCA for localization as expressed by equation (4) resorts to an unimodal position algorithm, which estimates the position based on the data stored in the database with the eigenvector closest to the acquired data at that instant. To solve this problem, a Markov localization (ML) algorithm is applied, integrating the images eigenvectors distance in a Bayesian probabilistic grid.

B. Bayesian position sensor with PCA-based probabilistic observations

One common method to provide a global localization in robotics is the ML algorithm, that is a straightforward application of a Bayes filter, often used in robot navigation [32], [11]. The ML provides the robot localization through a probabilistic grid map of the environment, that represents the probability knowledge of the robot about its own position. This algorithm applies the Markov assumption, where knowledge of the previous state and current inputs is enough to predict the probability of the current state. Thus, ML can estimate the states $\mathbf{x}$ in instant k based only in instant $k-1$. The Markov assumption make ML a multi-modal algorithm allowing the track of multiple possible positions, disambiguate the periodic information, and find the right robot position.

During an experiment, when the robot is moving, the probability about the robot states is continuously updated with the robot motion prediction and consequent observation, following the Markov algorithm in discrete instant k, where $\mathbf{x}(k-1)$ is the state vector in the previous iteration, $\mathbf{u}(k)$ the action, $\mathbf{z}(k)$ the observation in the current iteration and $\mathbf{m}$ the grip map of the environment (Fig. 2).

Markov localization ($\mathbf{x}(k-1)$,$\mathbf{u}(k)$,$\mathbf{z}(k)$,$\mathbf{m}$)
for i = 1 to N do
  $\bar{P}(\mathbf{x}_i(k)) = \sum_{i=i}^{N} p(\mathbf{x}_i(k)|\mathbf{u}(k), \mathbf{x}_i(k-1))P(\mathbf{x}_i(k-1))$
  $P(\mathbf{x}_i(k)) = p(\mathbf{z}(k)|\mathbf{x}_i(k), \mathbf{m})\bar{P}(\mathbf{x}_i(k))$
return $P(\mathbf{x}(k))$

Fig. 2. Markov localization algorithm

The correction of the prediction step is performed considering that the observation $\mathbf{z}(k)$ is given by the signal captured by the Kinect sensor and processed by the missing data correction:

$$P(\mathbf{x}_i(k)) = p(\mathbf{z}(k)|\mathbf{x}_i(k), \mathbf{m})\bar{P}(\mathbf{x}_i(k)), i = 1, \ldots, N \tag{5}$$

The probability of the robot being in any position of the map is obtained by the distance between the captured image and the corresponding image into the PCA eigenspace:

$$p(\mathbf{z}(k)|\mathbf{x}_i(k), \mathbf{m}) = (1 - \eta\|\mathbf{v} - \mathbf{v}_i\|_2), i = 1, \ldots, N \tag{6}$$

where η is a normalization factor, which ensures $\sum p(\mathbf{z}(k)|\mathbf{x}_i(k), \mathbf{m}) = 1$, $\mathbf{v}$ the eigenvector of the captured image, $\mathbf{v}_i$ the eigenvector of the i image into the eigenspace and $p(\mathbf{z}(k)|\mathbf{x}_i(k), \mathbf{m})$ the probability to observe $\mathbf{z}(k)$ in the state $\mathbf{x}_i(k)$ considering the map m.

Finally, the robot position given by the PCA-based position sensor with ML is obtained finding the mosaic that has the maximum probability calculated in (5):

$$[\hat{x} \; \hat{y}]^T = \arg\max_{\mathbf{x}_i(k)}(P(\mathbf{x}_i(k))); \tag{7}$$

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

To create the ceiling image database, a set of 1115 snapshots with depth images have been captured at pre-specified grid locations, with the robot in the same attitude, along a grid map with a distance of 0.3 m (in x and y axis) in an area of 18.9 $m \times 9.6$ m (Figs. 3–5). The captured depth images are cropped with a circular mark allowing the rotation and comparison of captured depth images when the robot is in the

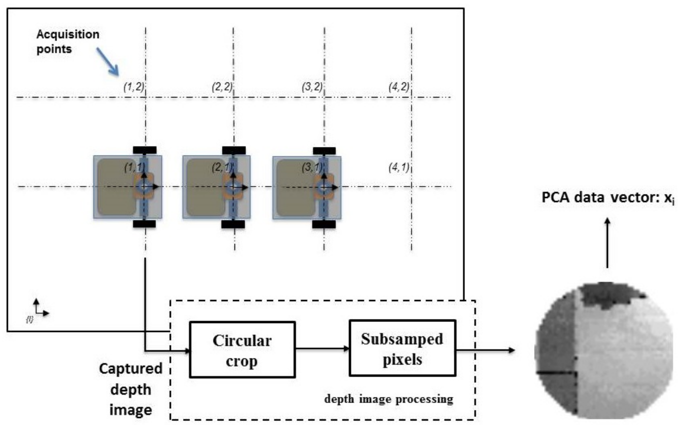

Fig. 3. Grid map and depth image processing to create a PCA eigenspace

Fig. 5. Ceiling view of the environment with periodic elements

same position, but with different attitude, during a mission. In order to compress the amount of data, the depth images are sampled with a compression ratio of $100:1$ and converted into a vector that will be added to PCA eigenspace. Therefore, analyzing the corresponding PCA eigenvalues and selecting the components that explain the images variability in an excess of $85\ \%$, leads to an image database of 9 eigenvectors. This, correspond to a reduction of $99.9\ \%$ in the memory resources when compared with the capacity needed to store the captured database, and $95\ \%$ when compared with the size after the subsample. In [8], [7], the authors followed a similar approach using a RGB camera, but the method revealed to be sensitive to illumination conditions.

During an experiment, it is possible to estimate the robot's attitude and position, as well as the angular motion speed and the robot's angular slippage, using only the signals obtained by the onboard sensors (Kinect, compass and encoders), through a self-localization sensor based in two KF and one PCA algorithm, with an architecture as detailed in Fig. 4.

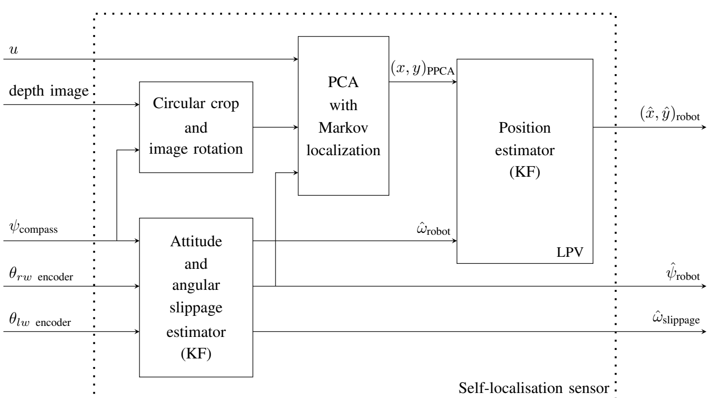

Fig. 4. Architecture of the self-localization sensor

The following notation is used in Fig. 4:

- $\psi_{compass}$ - orientation angle given by the compass;
- $\theta_{rwencoder}$ - angle given by the encoder of the right wheel;
- $\theta_{lwencoder}$ - angle given by the encoder of the left wheel;
- $(x,y)_{PPCA}$ - position coordinate given by the probability grid;
- $(\hat{x},\hat{y})_{robot}$ - estimated robot coordinates in the world referential;
- $\hat{\omega}_{robot}$ - estimated angular speed;
- $\hat{\omega}_{slippage}$ - estimated differential slippage.

Detailing the architecture of the self-localization sensor presented in Fig. 4, the KF depicted on the left of the figure implements the attitude optimal estimator model that is responsible to estimate the mobile robot attitude and the angular slippage. Once all acquired depth images for the PCA database are taken with the same orientation and compressed with a circular crop (Fig. 3), during a mission, the acquired depth images must be rotated to zero degrees of attitude, using the compass angle, and compressed with the same circular crop. The position estimator (on the right of the figure) implements a Linear Parameter-Varying (LPV) model as a function of the estimated angular speed in a KF, fusing it with the position obtained by the PCA with Markov localization algorithm.

Resorting to this architecture, it is possible to estimate the position, attitude and angular slippage of the mobile robot with a global stable error dynamic. For more details about this self-localization architecture see technical report [8].

B. Bayesian Motion Prediction

The method detailed in Fig. 4 applies the ML algorithm to compute the position of the robot in a probabilistic grid, considering the position of the mosaics captured to create the PCA database. Thus, the state vector of the Bayesian filter is the position of obtained by the PCA with the ML algorithm: $\mathbf{x}_i = [x_{PPCA} \quad y_{PPCA}]^T$. As is presented in Section III, the filtering step of the Bayesian filter is computed through a probability grid based on the PCA algorithm applied to the captured depth image. With the propose of computing automatically the probability of the state transition in the prediction step, and assuming that the robot moves in the direction given by the attitude estimator, the following method is applied.

Considering a grid with N_x and N_y mosaics in x and y axis, respectively, the probability map with $N_x \times N_y$ is

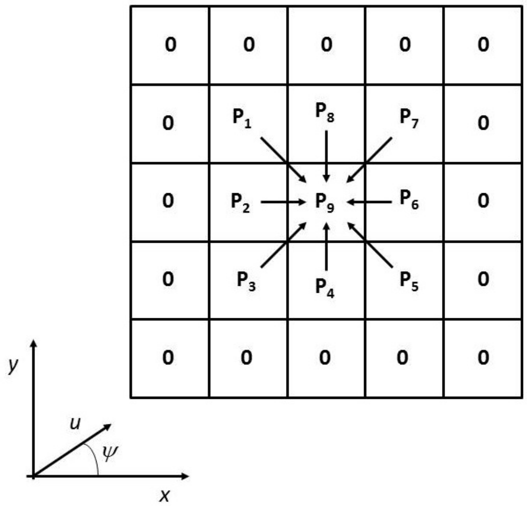

Fig. 6. Probabilities in prediction step

created and initialized with the same probability in all states $P(\mathbf{x}(0)) = \frac{1}{N}$, where $N = N_x \times N_y$.

During a mission, following the classical Markov localization algorithm, the robot position probability map is updated in a prediction step, considering the probability of the robot reaching a new state $\mathbf{x}_i(k)$ with the control $\mathbf{u}(k)$. This probability is obtained by the sum of all possible ways to the robot to reach the mentioned state.

$$\bar{P}(\mathbf{x}_i(k)) = \sum_{i=1}^{N} p(\mathbf{x}_i(k)|\mathbf{u}(k), \mathbf{x}_i(k-1))P(\mathbf{x}_i(k-1)), \quad i = 1, \ldots, N \tag{8}$$

Once that the states are stored in a grid, the probability map computation in the prediction step is performed considering the geometry of the grid and the speed and attitude of the robot, as it is shown in Fig. 6.

Analysing the probability state transition of Fig. 6 and considering that the robot is moving with a speed u in a direction with an attitude ψ, the images of eigenspace are captured with a distance d between mosaics and the localizer has been computed in a digital processor with a sampling time T, being the probability states transition is given by:

$$\bar{P}(\mathbf{x}_i(k)) = \eta \sum_{l=1}^{9} p_l P(\mathbf{x}_l(k-1)) \tag{9}$$

where η is a normalizer, which ensures $\sum \bar{P}(\mathbf{x}_i(k)) = 1$. Thus, considering that the robot is moving in a direction ψ, a possible way to compute the state transition probability from each 8 neighbors ($l = 1 \ldots 8$) is:

$$p_l = \begin{cases} (\cos(\psi + \tan^{-1}\frac{y_l - y_9}{x_9 - x_l}))^q & if p_l > 0 \\ 0 & if p_l <= 0 \end{cases} \tag{10}$$

$$p_l = \eta \cdot p_l(1 - p_9) \tag{11}$$

with η a normalization factor, which ensure $\sum p_l = 1$, and p_9 is the probability that the robot is kept in the same state.

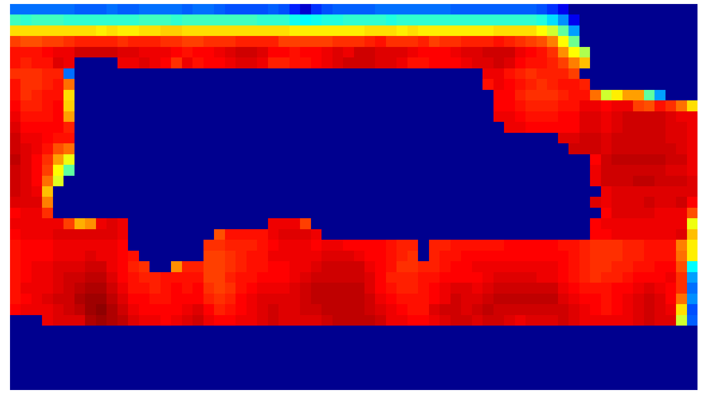

Fig. 7. Probability grid after 10 observations

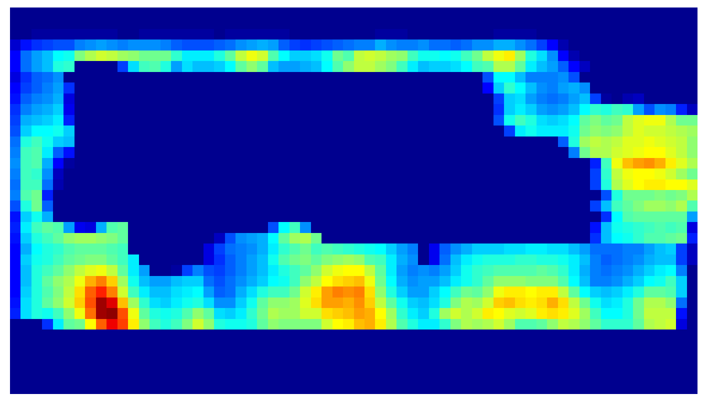

Fig. 8. Probability grid after 10 observations

Once $p_i < 1, i = 1 \ldots 8$, the exponent q allows to increase the probability between the states in the same direction of the robot attitude, reducing the probability between the states whose transition is caused by angular motion uncertainty (diagonal direction in state transition).

Finally, the new predict states probability map $\mathbf{x}_i(k)$ is obtained:

$$\begin{aligned} \bar{P}(\mathbf{x}_i(k)) = \; & \eta(\textstyle\sum_{l=1}^{8} (p_l \frac{u \cdot T}{d} P(\mathbf{x}_l(k-1))) + \\ & + p_9(1 - \frac{u \cdot T}{d}) \cdot P(\mathbf{x}_9(k-1))) \end{aligned} \tag{12}$$

C. Results for 2D localization

To test the mobile robot self-localization performance of the proposed approach, several tests have been performed in an environment with repeatability along a predefined path with 93 m, combining both straight lines and curves and travelling two laps inside the mapped area. During the experiment, the robot is moving with $0.1\ \mathrm{m} \cdot \mathrm{s}^{-1}$ robot speed and 2.5 Hz of sampling frequency and the real mobile robot trajectory has been measured allowing the comparison of the estimated position with the real one (ground truth test).

Figures 7–10 shows the probability grid performed by ML with a filtering step based on the PCA algorithm, where the "hot colors" (dark red) represents high localization probability and "cold colors" (blue) the low probability. The blue areas in the center, bottom and top in Fig. 7 represents unmapped areas which do not exist in the PCA eigenspace and, so, have null probability.

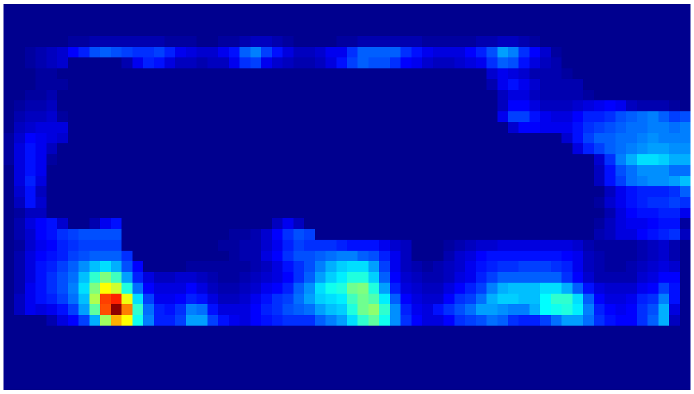

Fig. 9. Probability grid after 20 observations

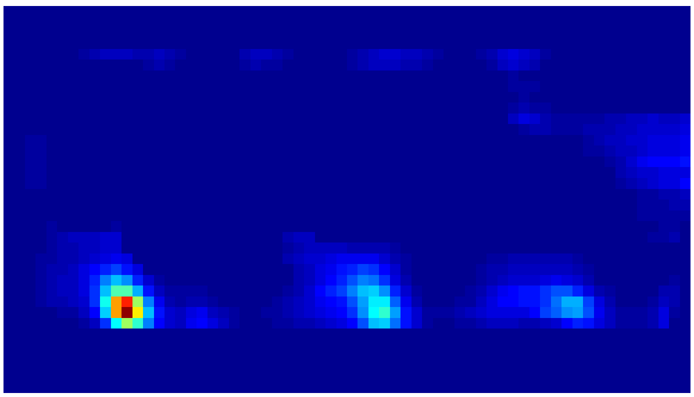

Fig. 10. Probability grid after 30 observations

Figures 8–10 show that ML is able to quickly disambiguate possible repeatability of the scenery and find the right position. Figure 10 shows that after 30 samplings (12 seconds), the position sensor has a high belief about the right position of the robot, which can be validated with the results presented in Fig. 11.

Figures 11 and 12 present the results of the self-localization sensor, comparing the position obtained by the ML with PCA, the KF position estimator and the real path of the robot, measured in the ground. The results of Fig. 11 show that the ML with PCA approach is able to achieve an accurate position of the robot, allowing good performance on the global self-position sensor. Notice that the precision of the grid with depth images has $0.3\ m$ of distance.

Figure 12 shows that the self-localization system estimates the path correctly while the path estimated by odometry diverges completely from the ground truth path. The blue circles represents the position uncertainty obtained by the Kalman filter. As it is possible to observe, the uncertainty increases when the robot are in the top of the Fig 12. This happen because in this area the celling has less information. Nevertheless, Fig 11 and Fig 12 show that the robot is still able to estimate the position of the robot, even when uncertainty increases.

Analyzing the results of the attitude estimator in Fig. **??**, it is possible to observe that the estimated attitude is very close to the ground truth, allowing to conclude that this Kalman filter provides results with good accuracy. For more details about

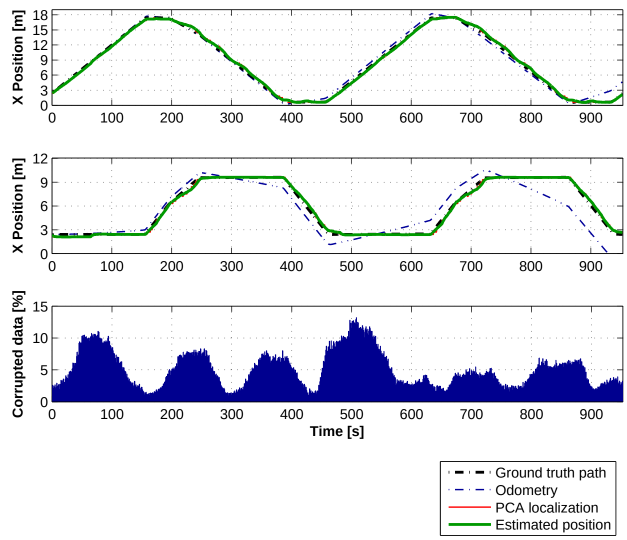

Fig. 11. Estimated position along time

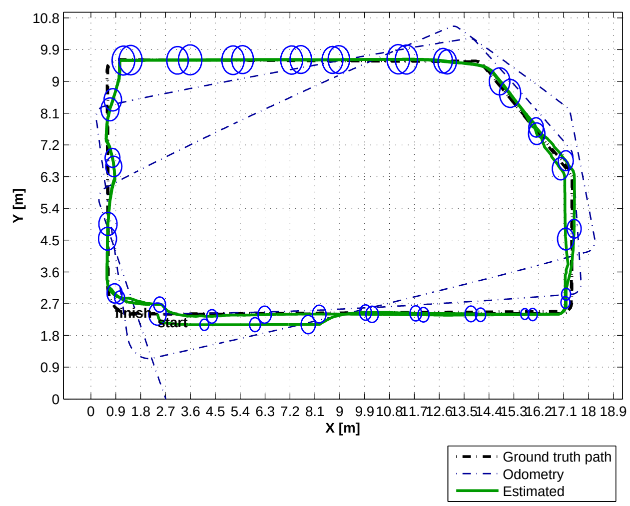

Fig. 12. Map with estimated position considering a ground truth path

Kalman filters design, see technical report [8].

Finally, analyzing the distribution of the estimated position error in Fig. 14 can be concluded that it is approximately Gaussian with a mean close to zero. The distribution is non-zero mean Gaussian because the trajectory is not random and due to the finite resolution of the PCA probabilistic grid ($0.3\ m$). However the mean error in both axis is very close to zero being $\bar{e_x} = 5.6 \times 10^{-2}\ m$ and $\bar{e_y} = -1.0 \times 10^{-2}\ m$.

V. CONCLUSIONS

The existence of scenarios with periodic elements or very similar in different points of the environment is usual and it can induce the position systems to a erroneous localization of mobile robots.

In this paper a Bayesian method using a grid map with an

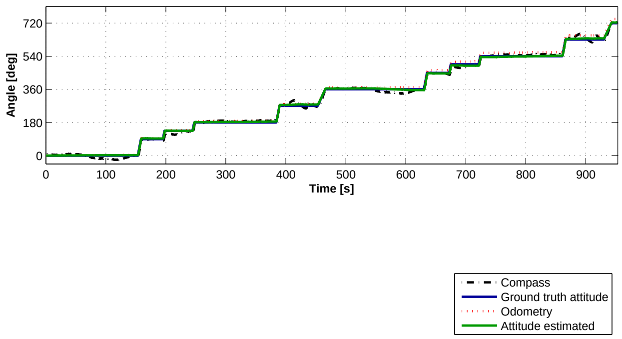

Fig. 13. Estimated attitude along time

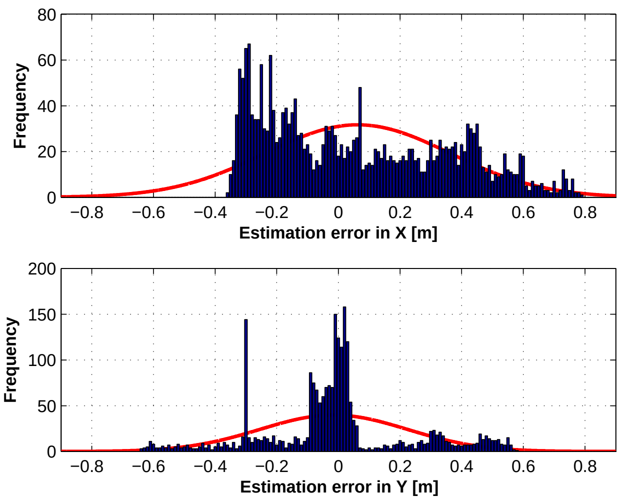

Fig. 14. Distribution of the estimated position error

observation belief based on PCA algorithm has been developed and experimentally validated. The proposed self-localization system is based on depth images of a Kinect sensor installed onboard of a mobile robot, looking upwards to the ceiling. Once the depth sensor often provides signals corrupted with missing data, caused by reflections problems in the IR beams, the proposed method uses an extension of a PCA algorithm to avoid the negative impact of missing data.

Thus, the proposed method allows the implementation of a mobile robot localization system to work in unstructured environments with periodic elements, and without the need of landmarks or the extraction of specific features. Moreover, the use of the PCA algorithm allows to compress the initial database with a high ratio and, during the experiences, to obtain the observation probability with a low computation efforts.

The integration of the Bayesian PCA-based position sensor with a linear Kalman filter allows to obtain an optimal an globally stable localization of the mobile robot.

In the future, the proposed localization method will be implemented in a path following control approach, where the self-localization system will be integrated in a control close loop performing different tasks like obstacle avoidance and human-robot interaction.

ACKNOWLEDGMENTS

The authors thank the valuable assistance of João Rodrigues in the development of the work reported in this paper.

This work was partially supported by Fundação para a Ciência e a Tecnologia, through IDMEC under LAETA.

REFERENCES

[1] Sergio Almansa-Valverde, José Carlos Castillo, and Antonio Fernndez-Caballero. Mobile robot map building from time-of-flight camera. *Expert Systems with Applications*, 39(10):8835 – 8843, 2012.

[2] Matej Artač, Matjaž Jogan, and Aleš Leonardis. Mobile robot localization using an incremental eigenspace model. In *IEEE International Conference on Robotics and Automation*, pages 1025–1030, Washington, DC, USA, May 2002.

[3] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): part II. *Robotics Automation Magazine, IEEE*, 13(3):108 – 117, September 2006.

[4] J. Biswas and M. Veloso. Depth camera based indoor mobile robot localization and navigation. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1697–1702. IEEE, 2012.

[5] C. Cardeira and J. Sá da Costa. A low cost mobile robot for engineering education. In *Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE*, pages 2162–2167, Raleigh, USA, 2005.

[6] F. Carreira, J. M. F. Calado, C. Cardeira, and P. Oliveira. Enhanced pca-based localization using depth maps with missing data. In *Robotica - 13th International Conference on Autonomous Robot Systems and Competitions*, pages 56–63, Lisboa, Portugal, april 2013.

[7] F. Carreira, C. Christo, D. Valério, M. Ramalho, C. Cardeira, J. M. F. Calado, and P. Oliveira. 2d pca-based localization for mobile robots in unstructured environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, pages 3767–3868, Vilamoura, Portugal, October 2012.

[8] F. Carreira, C. Christo, D. Valério, M. Ramalho, C. Cardeira, J. M. F. Calado, and P. Oliveira. Experimental validation of a PCA-based localization system for mobile robots in unstructured environments. IDMEC/CSI Internal Report, http://www1.dem.ist.utl.pt/cardeira/Mesh/Ca12b.pdf, June 2012.

[9] L. Chen, H. Hu, and K. McDonald-Maier. Ekf based mobile robot localization. In *Emerging Security Technologies (EST), 2012 Third International Conference on*, pages 149–154. IEEE, 2012.

[10] H. Cheng, H. Chen, and Y. Liu. Topological indoor localization & navigation for autonomous industrial mobile manipulator. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, volume 1, pages 238–243. IEEE, 2012.

[11] Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A Kantor, Wolfram Burgard, Lydia E. Kavraki, and Sebastian Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, Cambridge, MA, June 2005.

[12] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part I. *Robotics Automation Magazine, IEEE*, 13(2):99 –110, June 2006.

[13] D. Fox, W. Burgard, and S. Thrun. Active markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(3):195–207, 1998.

[14] Yasuaki Fukutani, Tomoyuki Takahashi, Masahiro Iwahashi, Tetsuya Kimura, Samsudin Siti Salbiah, and Norrima Binti Mokhtar. Robot vision network based on ceiling map sharing. In *11th IEEE International Workshop on Advanced Motion Control (AMC 2010)*, pages 164–169, Nagaoka, Niigata, March 2010.

[15] Arturo Gil, Oscar Mozos, Monica Ballesta, and Oscar Reinoso. A comparative evaluation of interest point detectors and local descriptors for visual slam. *Machine Vision and Applications*, 21:905–920, 2010.

[16] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *The International Journal of Robotics Research*, 31(5):647–663, 2012.

[17] WooYeon Jeong and Kyoung Mu Lee. CV-SLAM: a new ceiling vision-based SLAM technique. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3195–3200, 2005.

[18] I. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 2002.

[19] Ben Kröse, Roland Bunschoten, Stephan Ten Hagen, Bas Terwijn, and Nikos Vlassis. Household robots look and learn: environment modeling and localization from an omnidirectional vision system. *IEEE Robotics & Automation Magazine*, 11:45–52, December 2004.

[20] Bor-Woei Kuo, Hsun-Hao Chang, Yung-Chang Chen, and Shi-Yu Huang. A light-and-fast slam algorithm for robots in indoor environments using line segment map. *Journal of Robotics*, 2011:12, 2011.

[21] U. Larsson, J. Forsberg, and A. Wernersson. Mobile robot localization: integrating measurements from a time-of-flight laser. *IEEE Transactions on Industrial Electronics*, 43(3):422–431, June 1996.

[22] J.J. Leonard and H.F. Durrant-Whyte. *Directed sonar sensing for mobile robot navigation*, volume 448. Kluwer Academic Publishers Dordrecht, 1992.

[23] Z. Liu, W. Chen, Y. Wang, and J. Wang. Localizability estimation for mobile robots based on probabilistic grid map and its applications to localization. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2012 IEEE Conference on*, pages 46–51. IEEE, 2012.

[24] P. Oliveira. Interpolation of signals with missing data using pca. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, volume 3, page III, Toulouse, France, May 2006.

[25] P. Oliveira and L. Gomes. Interpolation of signals with missing data using principal component analysis. *Multidimensional Syst. Signal Process.*, 21(1):25–43, March 2010.

[26] Paulo Oliveira. MMAE terrain reference navigation for underwater vehicles using PCA. *International Journal of Control*, 80(7):1008–1017, July 2007.

[27] A. Ranganathan, E. Menegatti, and F. Dellaert. Bayesian inference in the space of topological maps. *Robotics, IEEE Transactions on*, 22(1):92–107, 2006.

[28] Davide Scaramuzza, Friedrich Fraundorfer, and Roland Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. In *IEEE International Conference on Robotics and Automation (ICRA2009)*, pages 4293–4299, Kobe, Japan, May 2009.

[29] Wolfram Burgard Sebastian Thrun and Dieter Fox. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents. MIT Press, August 2005.

[30] C. Siagian and L. Itti. Biologically inspired mobile robot vision localization. *IEEE Transactions on Robotics*, 25(4):861–873, August 2009.

[31] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *International Joint Conference on Artificial Intelligence*, volume 14, pages 1080–1087. LAWRENCE ERLBAUM ASSOCIATES LTD, 1995.

[32] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[33] De Xu, Liwei Han, Min Tan, and You Fu Li. Ceiling-based visual positioning for an indoor mobile robot with monocular vision. *IEEE Transactions on Industrial Electronics*, 56(5):1617–1628, May 2009.