Principal Component Analysis (PCA) and Hough Transform as Tool for Simultaneous Localization and Mapping (SLAM) with Sparse and Noisy Sensors

Stephanie Kamarry[1], Raimundo Carlos S. Freire[2], Elyson A. N. Carvalho[3], Lucas Molina [4] and Eduardo Freire[5]

[1,2]Stephanie Kamarry and Raimundo Carlos S. Freire, Federal University of Campina Grande, Paraíba, Brazil
[3,4,5] Elyson A. N. Carvalho, Lucas Molina e Eduardo Freire, Federal University of Sergipe, Sergipe, Brazil
e-mail: stephanie.sousa@ee.ufcg.edu.br

Abstract— **This work proposes a method of handling the difficulties generated by sparse and noisy sensorial output from a small quantity of ultrasonic sensors in order to develop a low cost SLAM system. A pre-processing step of detecting faulty sensors was implemented by applying PCA on the available data in order to extract more reliable baseline features through the Hough Transform. Furthermore, we analyze the influence of odometry errors and failures in the localization of a differential driven mobile robot. This method is suitable for indoor and orthogonal shaped environments, especially for medium and short term tasks, such as exploration, rescue and inspection. The experimental results demonstrate the accuracy and robustness to noise and sensorial failures.**

Index Terms— **Low cost SLAM, Robotics, PCA, Odometry, Sparse Sensors.**

I. INTRODUCTION

The coupling between localization and mapping has been labeled by the scientific community as SLAM (Simultaneous Localization and Mapping), and was originally proposed by Leonard and Durrant-Whyte based on previous works [1].

Many techniques that partially solve SLAM have been proposed since it was first stated as a problem. Most of those approaches, however, rely on the use of precise and dense measurements provided by laser sensors to correctly localize the robot and then construct detailed maps of complex environments [1, 2].

Even though SLAM is a popular theme, there is not much research focusing on low cost, sparse, noisy sensors, with measurement errors and sensor failures, such as ultrasonic sensors [3, 4, 5]. By using low cost sensors instead of those based on high-precision laser measurements, SLAM turns into an even more challenging task [6].

While not as precise or dense as laser sensors, ultrasonic sensors are still attractive due to their much lower price, smaller size, weight and minimal computational requirements. Therefore, ultrasonic sensors are adequate for the end-consumer, due to the necessity of a low product cost, implying less computational capabilities and power availability.

A well known SLAM application is the development of autonomous robots for tasks like safety, cleaning, and entertainment. A popular example is the Roomba Discovery™, which is a commercial autonomous platform created to do domestic activities such as vacuum cleaning the floor. Roomba™ normally uses random exploration in order to clean the whole environment. The combination of SLAM techniques focused on sparse sensors could increase the efficiency of this robot when executing the task.

In this paper, the Hough Transform is used to detect line segments from the information acquired as output of sparse and noisy ultrasonic sensors in order to estimate orientation of a differential driven robot. The orientation was chosen for this analysis because it has the worst error propagation from the sensors for this type of robot, as mathematically proven in [7]. The estimation of the X and Y position is made based on the Extended Kalman Filter (EKF).

The choice of line segments is adequate to estimate orientation, since most of inside environments are made of straight lines, which can be either parallel or perpendicular to each another. As presented in [8], this work is developed based on the assumption that the environment shape is orthogonal. Thus, it can be mapped using only parallel or perpendicular lines. This assumption reduces both algorithm complexity and processing time.

In order to improve the processing time we also used an occupancy grid to represent the map when the information coming from the sensors is transformed to a global coordinate system. Then, less memory is used to store the representation of the environment as demonstrated in [9].

The great challenge in using line segments to estimate an orientation is how to extract them from scarce and often unreliable sensorial data. Therefore, this work aims to solve the SLAM problem in low cost sensors by using PCA to detect faulty and compromised sensors. The choice of PCA is due to the fact that it is processed in similar problems in other engineering areas set this approach as a candidate for the solution of the problem. In addition, its simplicity of implementation and construction based on simple concepts allows the scalability for any problems. Especially for the problem stated in this work, in which few sensors are used, the use of PCA is advantageous in relation to the technique based on autoencoders resignation for being able to have a good performance in environments with little information and no need of first training stage.

This paper is organized as follows: in section II the main works in SLAM with sparse and noisy sensors are presented. The developed mapping system is presented in section III. Different localization approaches are shown in section IV. Results are presented in section V. Finally, conclusions and future works are discussed, followed by references.

II. SLAM WITH SPARSE AND NOISY SENSORS

Several works in current literature focus on solving different aspects surrounding SLAM, and most of them is derived from the pioneer work done by Smith et al [1].

Digital Object Identifier 10.29292/jics.v15i3.162

Most of SLAM frameworks using proximity sensors concentrate on laser scanners [1, 2, 10, 11]. These sensors perform precise and dense measurements, reducing the negative effect of failures and uncertainties. However, these sensors are not suitable for low cost applications, being the ultrasonic ones demanded for that.

Burguera [3] proposed a localization strategy based on scan matching through and extension of the ICP (Iterative Closest Point) algorithm for ultrasonic sensors. In this approach, the sensors' readings are grouped to overcome their low density and the whole trajectory of the robot during the grouping process is then corrected after. In Burguera's work, however, the low angular resolution and noisy behavior of ultrasonic sensors are not taken into consideration.

The vinySLAM method [4] is used for robots that provide information about proximity to nearby obstacles with a laser scanner and it is supposed to be used in an indoor environment. The algorithm is proposed based on the MonteCarlo scan matcher and the random walk approach.

To extract only reliable information from the sensors, Rencken [2] classifies landmarks as hypothetical, experimental, or confirmed. The biggest drawback of this technique is the amount of memory needed to store all the data and the resulting delay to confirm every information.

Zunino and Christen [12] solved SLAM by using an Extended Kalman Filter (EKF) while presenting a method to detect failures of the EKF and recover from them. Another probabilistic approach was proposed by Tardós [13]. In their work, maps are grouped in such a way that allows for the construction of a limited size sequence of independent stochastic maps, and then combining these maps in a globally consistent manner.

Recently, Schröter, Böhme and Gross [14] presented a probabilistic map correspondence approach using the Rao-Blackwellized Particle Filter (RBPF), which enabled them to solve SLAM using low resolution ultrasonic sensors. The problem of this technique is the fact there is no way to recover information when a particle is lost.

In [15] a mapping system suitable for small mobile robot is presented. They used occupancy grid mapping by an optimized Backtracking Occupancy Grid-Based Mapping. The robots are equipped with infrared proximity sensores, ultrasonic beacon system and encoders. The algorithm also optimizes exploration of the surrounding. A similar approach using occupancy grid was proposed by Khan [16].

Most approaches address the problem of bad sensorial information in SLAM through probabilistic filters or data grouping [5]. In the first case, algorithms tend to be computationally intensive and not produce consistent estimates. For the second case, grouping sensorial data to reduce its sparsity creates an unavoidable delay: information is unavailable for the system during the grouping interval. Considering these facts, this work aims for a simple alternative with low computational and monetary cost to deal with sparse, limited, and faulty sensors with neither probabilistic methods nor data grouping techniques, thus solving the problem and updating the map in real time.

III. PCA BASED MAPPING

The sensorial data quality is a factor with direct influence over a mapping system. Thus, processing this information in order to identify reliable outputs has to be the first step when solving the mapping problem. Then, the way to represent the map must be adequately chosen, since it greatly affects error detection.

As developed in a previous work [17], unreliable measurements can be detected using PCA. This technique is based on the idea that there exists a high correlation between sensor readings, being this correlation modified in case of anomaly. This way, it is possible to determine when anomalies happen in any sensor by observing the correlation matrix of sensorial data over time. Once a faulty sensor is detected, its reading is temporarily ignored during the map construction.

Aiming to detect faulty sensors, the proposed method consists first of finding an unreliable measurement and weighting this problem, then localizing the corresponding sensor and temporarily discarding its outputs, if needed.

In the experiments performed, 32 distance sensors were used to acquire data regarding the position of obstacles in the environment. At each sensor measurement, the 32 samples collected are organized as a vector according 1

$$V[k] = [x_1, \dots, x_n, y_1, \dots, y_n, X, Y, \theta]^T, \tag{1}$$

such that k indicates the time of measurement, x_n represents the coordinate x of sensor measurement n, represented in the robot coordinate system, and y_n represents the coordinate y of sensor measurement n, in the robot coordinate system and the variables X, Y, θ represent the robot pose in the environment coordinate system obtained from the encoder.

For PCA application, a matrix *M* is constructed (2) from the samples collected by the sensors (1), as follows

$$M = \begin{pmatrix} V[k] & V[k+1] & \dots & V[k+t] \end{pmatrix} \tag{2}$$

Rewriting according to the elements $m_{i,j}$:

$$M = \begin{pmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,t} \\ m_{2,1} & m_{2,2} & \dots & m_{2,t} \\ \vdots & \vdots & \ddots & \vdots \\ m_{l,1} & m_{l,2} & \dots & m_{n,t} \end{pmatrix} \tag{3}$$

the dimension is $l \times t$, such that t is the number of measurements considered (number of V vectors collected) and l is the size of the V data vector,that is $2n + 3$.

In this work, measurements are analysed using PCA from three distinct moments. This value was determined experimentally as the smallest number of samples that allowed the detection of anomalies in sensory measurements without introducing a delay in the system.

The average array data *M* must be subtract, generating a new data matrix, *G*, with zero mean and same dimension of *M*.

The next step is to calculate the covariance matrix of *G* and then extract their eigenvectors and eigenvalues. The eigenvalues represent the energy of the associated eigenvectors, where each eigenvector represents each component that composes the data.

Calculating the eigenvalues, the highest energy component is then used to construct a matrix $\hat{M}$. This matrix is a more representative and usually more compact transformation of observations, i.e. it is a projection of data in the most relevant direction - the main component.

To quantify the anomalies detected by the PCA, the *MSPE* (Mean Square Prediction Error) (4) of the matrix $\hat{M}$ relative to the original data matrix *M*.

$$MSPE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(m_{i,j} - \hat{m}_{i,j})^2}{nm} \quad (4)$$

In this work, measurements are analysed using PCA on data from three distinct moments. After the PCA step, errors are quantified with MSPE (Mean Square Prediction Error), such that the resulting MSPE from each block influences the calculation of the next MSPE. After three blocks (of three instants each) the anomalies are determined. Corrupt measurements are separated by using an experimentally determined threshold. As a result of this separation, all problematic information is ideally excluded, generating a new matrix from reliable data. This procedure has to happen in real time.

The idea of inserting a pre-processing step is based on the need of identifying faulty sensors such that their outputs wouldn't influence the mapping process.

In the next step, all doubtful information identified is discarded and not used to construct the map. Then, regular grids are used to represent the ambient, discretizing only trustful data. After the first nine time intervals, the proposed method is applied.

IV. LOCALIZATION

Proper localization of a mobile robot consists in finding its position and orientation in space, given a time interval. Mobile robot localization can be divided in two categories: relative or absolute localization [18].

Relative localization methods use previous localization data to estimate the current position. Odometry and inertial navigation are based on this principle [18]. Due to its low cost, odometry is widely used in wheeled mobile robots.

For differential driven robots, two incremental encoders are coupled to each motor in order to measure rotations of the corresponding wheel. By knowing the kinematic equations and parameters, encoder measurements can be used to ideally determine the position and orientation at any time if the starting localization is known [19].

Even though odometry is extensively used due to its simplicity, this technique can be exposed to types of errors that can be identified as either systematic or random.

Systematic errors are those caused by uncertainties in the robot's kinematic model, used to transform encoder measurements in position and orientation. This kind of error can have a significant impact on the behavior of the robot, greatly distorting localization over time, since it accumulates. In the other hand, random errors are caused by unexpected situations such as wheel slippage, ground imperfections or other similar fenomena.

Since it's not only constantly present, but also cumulative, systematic error has the greatest influence over the robot's localization estimate. The accumulation effect is not present in absolute localization methods.

Absolute localization is able to provide a past-independent estimate of the robot's localization, not depending on integration. Different to relative measurements, errors present in this method don't grow indefinitely. In this case, the robot's position is derived from either maps or landmarks.

In the previous work [17], the localization was based on a map, but different to traditional methods, this map was previously unknown. Furthermore, there were no artificial landmarks, facilitating application of the method in any environment. The sensor measurements acquired from each time step were processed by the PCA so that data could be used to construct the global map.

Initially, the first representation of the global map is defined, by transforming the outputs of ultrasonic sensors (sparse and noisy) in the first time step. For that purpose, the information is pre-processed using the technique explained in the previous section. When measurements are tagged as reliable, data is transformed to the global coordinate frame (5) in $t = 1$, through a simple change of coordinate system. At this moment, all odometry is reliable since there was no error accumulation and the initial pose is known.

$$Data_{global}(t) = R \times \begin{bmatrix} X_{data}(t) \\ Y_{data}(t) \end{bmatrix} + \begin{bmatrix} X_{odom}(t) \\ Y_{odom}(t) \end{bmatrix} \quad (5)$$

such that $Data_{global}$ represents positions in the global reference frame, X_{data} and Y_{data} are X and Y positions in the robot's reference frame, X_{odom} and Y_{odom} together form the robot's position in the environment measured from odometry, and R is the rotation matrix obtained from (6).

$$R = \begin{bmatrix} \cos(\theta_{odom}(t)) & \sin(\theta_{odom}(t)) \\ -\sin(\theta_{odom}(t)) & \cos(\theta_{odom}(t)) \end{bmatrix} \quad (6)$$

After transformed to the global coordinate system, this data is stored in an occupation grid with $1cm$ resolution. As shown in [17], the proposed method is not influenced by the grid resolution. Therefore, the choice of resolution for this work was done experimentally. The occupation is binary, that is, if there is a point inside a cell, it is considered as occupied (1). Otherwise, it is empty (0).

In possession of an initial representation of the map, the SLAM process starts. In each step, measurements from sensors are transformed to the global reference frame by knowing the current position (5). However, instead of using (6) to find the rotation, the system of absolute localization is used to estimate R_{esti} (7).

$$R_{esti} = \begin{bmatrix} \cos(\theta_{esti}(t-1)) & \sin(\theta_{esti}(t-1)) \\ -\sin(\theta_{esti}(t-1)) & \cos(\theta_{esti}(t-1)) \end{bmatrix} \quad (7)$$

where θ_{esti} the estimated orientation. To calculate it, the Hough Transform is used to find lines from both the global map and sensor measurements.

The orientations of these lines are compared to each other. The least change in orientation ($\Delta\theta$) gives the variation from previous to current orientation (8).

$$\theta_{esti}(t) = \theta(t-1) + \Delta\theta \tag{8}$$

As proven in [7], orientation is the measurement with greatest error propagation for odometry, in the case of a differential track robot. This can be corrected with the developed technique.

To assure there are no rough estimations, a safety step verifies if $\Delta\theta$ is lesser than a threshold defined using the maximum robot speed. When this condition is not satisfied, the previous orientation is maintained (9).

$$\theta(t) = \theta(t-1) \tag{9}$$

In order to reduce processing time, the whole map is not used to compare the data, but only a portion of it closest to the robot, defined as a circle with radius r. This can be done since the robot navigates in a limited speed and ultrasonic sensors have a limited range (10).

$$r = V_{mx} \times \triangle t + \alpha \tag{10}$$

such that V_{mx} is the maximum speed of the robot, Δt is the time between samples, and α is the range of the ultrasonic sensors.

The proposed method was tested together with a simple exploration strategy, with results presents in the next section.

V. RESULTS

Simulation tests were developed to validate the method proposed in this work. In these experiments, a differential drive robot with 32 sparse and noisy ultrasonic sensors navigated with limited speed in an unknown environment. In this specific case, all measured data was noisy and the sensors saturated in 300 cm. Additionally, 8 sensors would randomly experience faults: two of them being always saturated, two with no measurements, and the other four with big measurement noise (between 20 and 80 cm).

Considering no error in odometry, the map obtained after exploration is shown by Fig. 1. Measurements with arc patterns occur due to sensory saturation.

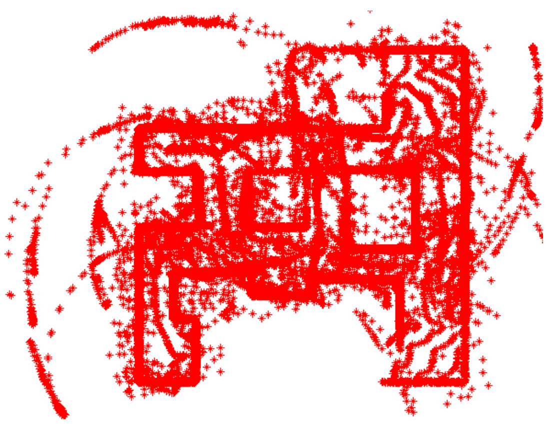

Fig. 1 Noisy data obtained from ultrasonic sensors.

By estimating the robot's position from optical encoders in each wheel, both X and Y estimates can be compared to the real positions over the experiment. Even though there is an incremental error in odometry for X and Y, it accumulates slowly when compared to the orientation measurement.

This is because the error propagation is indeed much more severe for orientation measurements, as proven in [20] and [7].

With such corrupt data, it would be impossible to recover any information from the map, hindering any attempt to navigation. To solve the problem of cumulative orientation error, the technique proposed by this article was then used. The results after applying the orientation estimation methodology are presented in Fig. 2.

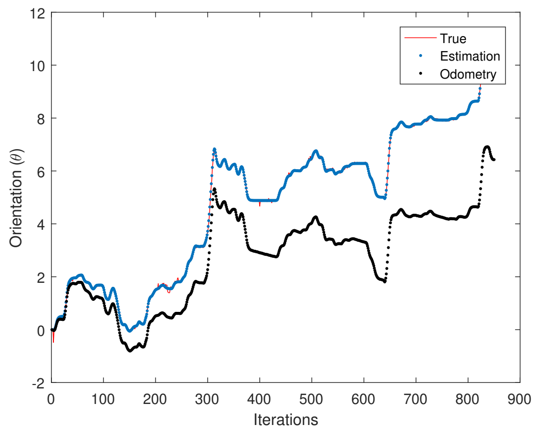

Fig. 2 Real θ (blue), Technique proposed (red) and measured value (black).

The estimation of the X and Y position is made through the EKF. The results after applying the EKF are presented in Fig. 3 and Fig. 4.

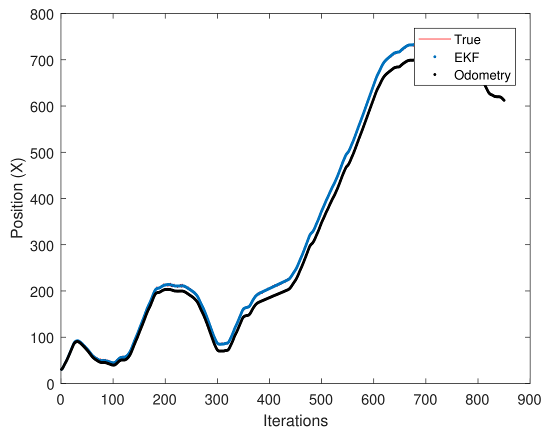

Fig. 3: Real X position (blue), EKF estimation (red) and measured value (black).

After the orientation error was under control, it was possible for the robot to navigate in an unknown space, localizing itself and then constructing a map. By representing this map in an occupation grid, obtained through a pre-processing step using PCA together with the orientation estimation technique, a final result was achieved, presented by Fig. 5.

One of the key performance parameters is the processing time. So a SLAM algorithm should handle scans as fast as possible to operate in real time. Mean scan processing time (MSPT) of the considered algorithms was tested on laptop (Core i7 2.6 GHz CPU, 8 GB RAM).

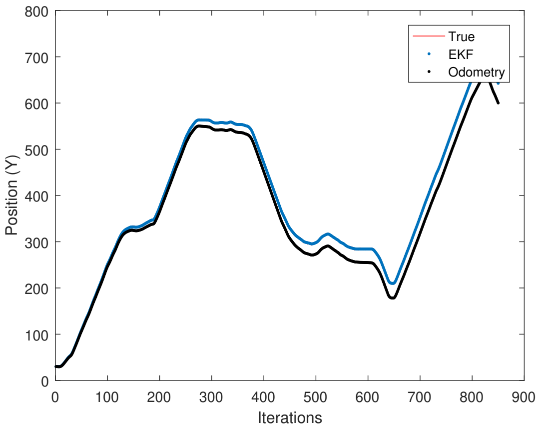

Fig. 4: Real Y position (blue), EKF estimation (red) and measured value (black).

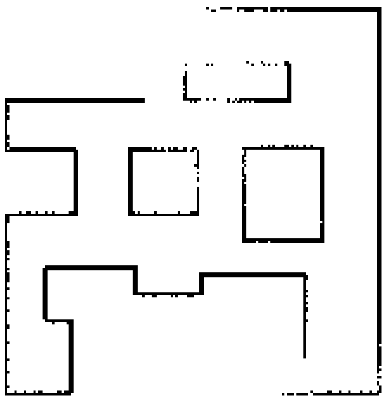

Fig. 5 Final map after the SLAM process.

The results of testing was 2.95 ms which corresponds to 19.93% of the MSPT of the [4], cited in the literature review.

VI. CONCLUSION

This work proposes a new method for low cost SLAM, based on using the PCA and Hough Transform. Since it is not based on any probabilistic approach, this proposal differs in concept to traditional methods, besides having a lower computational cost.

As expected, directly using odometry to localize the robot results in a map full of errors. Because of this, it is important to use localization tools that don't accumulate error over time. Considering this problem, a localization method was developed by using the same map the robot was constructing to aid in finding its position on each step.

When representing the map in an occupation grid, it was possible to use the Hough Transform to search for lines, reducing the magnitude of measurement errors from sensors. The developed localization system was greatly improved by correction of odometry error through online map usage, solving all presented problems.

It is important to emphasize that this method has a low computational cost, it's easy to implement and doesn't require expensive laser sensors.

REFERENCES

[1] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty," *The international journal of Robotics Research*, vol. 5, no. 4, pp. 56–68, 1986.

[2] W. D. Rencken, "Concurrent localisation and map building for mobile robots using ultrasonic sensors," in *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93)*, vol. 3. IEEE, 1993, pp. 2192–2197.

[3] A. Burguera, G. Oliver, and J. D. Tardos, "Robust scan matching localization using ultrasonic range finders," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 1367–1372.

[4] A. Huletski, D. Kartashov, and K. Krinkin, "Vinyslam: an indoor slam method for low-cost platforms based on the transferable belief model," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6770–6776.

[5] N. M. Yatim and N. Buniyamin, "An overview of simultaneous localization and mapping using low-cost mini robot platform," *Advanced Science Letters*, vol. 23, no. 11, pp. 11 053–11 056, 2017.

[6] T. N. Yap and C. R. Shelton, "Slam in large indoor environments with low-cost, noisy, and sparse sonars," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 1395–1401.

[7] J. G. N. De Carvalho Filho, L. Molina, E. Olveira, and E. Á. N. Carvalho, "The impact of parametric uncertainties on mobile robots velocities and pose estimation," *IEEE Access*, vol. 7, pp. 69 070–69 086, 2019.

[8] V. Nguyen, A. Harati, A. Martinelli, R. Siegwart, and N. Tomatis, "Orthogonal slam: a step toward lightweight indoor autonomous navigation," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 5007–5012.

[9] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[10] A. Eliazar and R. Parr, "Dp-slam: Fast, robust simultaneous localization and mapping without predetermined landmarks," in *IJCAI*, vol. 3. Acapulco, Mexico, 2003, pp. 1135–1142.

[11] T. Duckett *et al.*, "A genetic algorithm for simultaneous localization and mapping," 2003.

[12] G. Zunino and H. I. Christensen, "Navigation in realistic environments," in *9th Intl. Symp. on Intelligent Robotic Systems, Toulouse, France*, 2001.

[13] L. M. Paz, J. D. Tardós, and J. Neira, "Divide and conquer: Ekf slam in $o(n)$," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1107–1120, 2008.

[14] C. Schröter, H.-J. Böhme, and H.-M. Gross, "Memory-efficient gridmaps in rao-blackwellized particle filters for slam using sonar range sensors." in *EMCR*, 2007.

[15] D. Gonzalez-Arjona, A. Sanchez, F. López-Colino, A. De Castro, and J. Garrido, "Simplified occupancy grid indoor mapping optimized for low-cost robots," *ISPRS International Journal of Geo-Information*, vol. 2, no. 4, pp. 959–977, 2013.

[16] M. S. Khan, S. S. Chowdhury, N. Niloy, F. T. Z. Aurin, and T. Ahmed, "Sonar-based slam using occupancy grid mapping and dead reckoning," in *TENCON 2018-2018 IEEE Region 10 Conference*. IEEE, 2018, pp. 1707–1712.

[17] S. Kamarry, L. Molina, J. R. Montalvao Filho, E. Olveira, and E. Á. N. Carvalho, "Mapeamento utilizando análise de componentes principais," *CBA*, 2012.

[18] J. Borenstein, H. Everett, and L. Feng, *Navigating mobile robots: Systems and techniques*. AK Peters, Ltd., 1996.

[19] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT press, 2011.

[20] J. G. N. De Carvalho Filho, E. Á. N. Carvalho, L. Molina, and E. O. Freire, *Propagação de incertezas estruturais no modelo cinemático de robôs móveis*, Campina Grande, 2010, vol. 1.