

Autonomously Simultaneous Localization and Mapping Based on Line Tracking in a Factory-Like Environment

Akif DURDU¹, Mehmet KORKMAZ²

¹Department of Electrical and Electronics Engineering, Faculty of Engineering and Natural Sciences, Konya Technical University, Ankara Caddesi No:6, 42030, Karatay/Konya, Turkey

²Department of Civil, Environmental and Geodetic Engineering, The Ohio State University, Columbus, 43210 Ohio, United States of America

akifdurdu@gmail.com, mkorkmazeem@gmail.com

DOI: 10.15598/aeec.v17i1.3048

Abstract. This study is related to SLAM, also known simultaneous localization and mapping which is highly important and an indispensable issue for autonomous mobile robots. Both an environment mapping and an agent's localization are provided with SLAM systems. However, while performing SLAM for an unknown environment, the robot is navigated by three different ways: a user guidance, random movements on an exploration mode or exploration algorithms. A user guidance or random exploration methods have some drawbacks that a user may not be able to observe the agent or random process may take a long time. In order to answer these problems, it is searched for a new and autonomous exploration algorithm for SLAM systems. In this manner, a new kind of left-orientated autonomous exploration algorithm for SLAM systems has been improved. To show the algorithm effectiveness, a factory-like environment is made up on the ROS (Robot Operating System) platform and navigation of the agent is observed. The result of the study demonstrates that it is possible to perform SLAM autonomously in any similar environment without the need of the user interference.

Keywords

Active SLAM, autonomous agents, indoor navigation, simultaneous localization and mapping.

1. Introduction

It is a well-known fact that robotic applications have been increasing day after day and robots have performed an assistance to human from health to indus-

trial applications [8], [12], [14], [18] and [28]. In order for a robot to be able to fulfil a task, it has to know its location and what the world looks like around it. It is agreed that the problem of where the robot is seen as a localization problem. Moreover, the problem of constructing a map of the environment is specified as a mapping one [3] and [25]. Despite the fact that these two issues tackle separately, it may be impossible to give the robot neither location nor map in some cases. Therefore, it is necessary to build a map of an environment while simultaneously localize the robot within this map for such situations. When the literature is scrutinized, it can be seen that this problem is called as simultaneous localization and mapping or acronym of it SLAM [6]. SLAM deals with a construction of a map of an environment in which concurrently localize itself within it. Due to the fact that it could gain an autonomy to the robot, SLAM has been seen a 'holy grail' [6] and, it is an important milestone for the mobile robotic applications. From this point of view, a mobile robot is able to have an information about where it is or where to go by courtesy of SLAM.

Many algorithms have been presented about SLAM from the 2-D maps and metrics maps to 3D or topologic ones, from the filter based approaches to the vision-based ones [1], [11], [19], [28] and [30]. According to the related previous studies, a guidance of robots in an unknown environment could be done in three different ways:

- The first one is a user navigation and it can be thought an ideal and the most efficient solution due to the fact that it is based on human observation. By way of this method, the robot can be navigated to an unmapped area. One of the limitations of this method is that it is not explicit

what happens if the user is not able to observe the robot or the exploration area.

- The second guidance method is the random exploration method. The robot is allowed to run in an exploration mode so that the robot might explore the area moving random orientation and movement. A critical weakness of this method, however, is that it takes a long time to discover all over the area [14], [21], [22], [24] and [29].
- The third method is the special algorithms for autonomous navigation and exploration. When these types of algorithms are combined with SLAM, it is usually called active SLAM approach which ensures the full autonomy for the mobile robot. These approaches generally are benefited from occupancy grid maps. The environment is split into grids and the robot is navigated to the unexplored regions [13], [24] and [31].

The first two methods suffer from some serious disadvantages as mentioned above while the algorithms under the third section generally count on the laser measurements. In this context, the presented study can be discussed under the third class of active SLAM algorithms. Our enhanced approach differs from the existed algorithms because it is image-based instead of laser scanning.

2. Simultaneous Localization and Mapping (SLAM)

The first serious discussions and analyses of SLAM have emerged during the late 1980s. The idea of gathering probability and robotics were the heart of the matter [5] and [23]. There has been notable progress on the solution of the problem after then implementation of Bayes based filters such as Kalman Filter (KF) for linear systems and Extended Kalman Filter (EKF) for non-linear ones [8], [24] and [26].

SLAM is mathematically described in the form of probability in which sensor and control data of the robot are inputs; a map and a pose of the robot are outputs Eq. (1) [6], [8], [21] and [22]:

$$P(x_{0:t}, m_{1:M} | Z_{1:t}, u_{1:t}), \quad (1)$$

where m is a map created with the algorithm and x is pose information of the robot. These two terms also stand for global state parameters of SLAM. On the other hand, z is identified as sensor observations and u is robot control inputs. At the first stage of EKF-SLAM, the state is predicted considering the robot previous state and the control input. At the latter phase,

the prediction has been updated by using sensor observations.

However, traditional EKF method suffers from the non-Gaussian noise cases and a big size covariance matrix if the number of landmarks is relatively high [3], [4] and [6]. Because of some downsides of EKF-SLAM, new methods have been improved. The most striking development at this point was the implementation of Particle Filters (PF) to SLAM problem. However, PF also suffers from a big size covariance matrix due to the fact that each particle represents the individual solution. To overcome this problem, the remarkable solution was the implementation of Rao-Blackwellization decomposition method along with the PF. This method is also called FastSLAM. By means of this decomposition, SLAM problem has turned into a classical Monte Carlo Localization (MCL) of the robots and traditional EKF mapping Eq. (2) [1], [9], [15], [16] and [21]:

$$p(x_{0:t}, m_{1:M} | Z_{1:t}, u_{1:t}) = p(x_{0:t} | Z_{1:t}, u_{1:t}) \prod_{i=1}^M p(m_i | Z_{1:t}, X_{0:t}). \quad (2)$$

Assuming to landmark locations independent, localization and mapping can be handled separately. Thanks to this trick, it is computed M times 2 by 2 matrix instead of tackling with the big size of M by M covariance matrix facing on conventional EKF. With this development, computation speed of the algorithm has been notably increased and non-Gaussian distribution representation of the model has been developed. Considering from this point of view, each particle has its belief on the potential solution (map and pose of a robot). The particles have been updated in every iteration when the environment is observed.

3. Implemented System

A factory-like environment has been made up to demonstrate the effectiveness of the improved autonomous algorithm. The lines used to separate the sections from each other are considered to form the working environment (Fig. 1). A pure line follower algorithm is highly likely to produce unstable results considering the robot has to turn when faced rotation points or the endpoint of the lines. In order to get over this problem, a left-orientated follower algorithm has been developed and the movements of the robot have been provided within this framework.

3.1. Evaluation of Images

The robot takes an image (Fig. 2(a)) of the environment via a camera mounted on it. With these images,

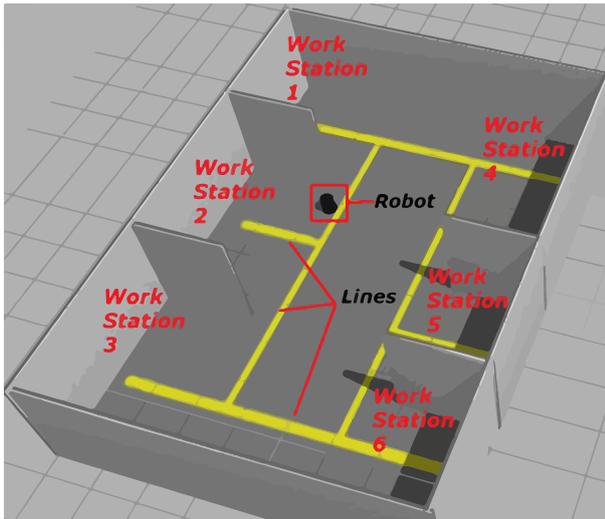


Fig. 1: Environment view, robot and tracking lines.

the navigation path of the robot can be determined by means of image processing algorithms.

The images taken from the camera of the robot are being processed continuously. During this period, several types of images have been produced like HSV and masked ones (Fig. 2 and Fig. 3). Segmented images are used to determine the foreground path.

First of all, images are transformed into HSV images. The aim of this transformation is to provide a more reliable result on the evaluation of the images because of the fact that HSV images are more robust on the change of brightness, shadow effects etc. Otherwise, the areas covered shadows might be misidentified (Fig. 3(b)).

After applying HSV transformation to the image, a novel approach has been thought due to the fact that the assessing of the whole image both will be difficult and increase the computational time. To this end, the image is divided into three sub-regions, x_1 , x_2 , x_3 . x_1 and x_3 parts are multiplied by 0 to exclude the irrelevant regions. Only the region x_2 is let alone and decisions are made based on that region. In this way, only the region to be tracked are extracted from the whole image and the robot is manipulated to the true path. The value of the regions has been calculated by empirically after examining some trials. For this study, the values of the parameters have been computed like $x_1 = 3 \cdot h/4$, $x_2 = 20 \sim 30$ (about 1 meter ahead of the robot), $x_3 = h - (x_1 + x_2)$ (Fig. 4 and Fig. 5) where h is the height of the image, while w is the width of it.

In addition to this, it is needed to use another masked image to determine the left orientation. And therefore, some right part of the original masked image ($w - x_4$) is also covered with zero so that the left orientation can be decided by means of the difference on image moments (Fig. 5 and Subsec. 3.3.).

3.2. The Robot, Sensor and Movement of the Robot

Turtlebot II, which is also known as Kobuki Turtlebot is used to verify the demonstration of the study. This robot is widely accepted and used in academic experiments. It has linear movement in the direction of $+x/-x$ and radial rotation on the $+z/-z$. This robot is preferred to carry out the studies because of the fact that it is cheaper, enabled to assemble a variety of sensors and allowed to use open source material to control [7]. The robot is utilized from a differential drive to steer and detailed analyzing about its kinematic and movement equations are given on Cook [2]. The robot generally comes together with an integrated camera, which is usually Asus Xtion or Microsoft Kinect also used in this study. The Kinect is able to give depth data as well as RGB one.

It could perform the scanning of an environment via infrared lights on 57 horizontal and 43 vertical degrees. The depth information given sensor is generally accepted reliable up to 5 meters. The different application of robotics has been widely benefited from this sensor because of its cheapness and presenting valuable data, especially in depth since its firstly announcing in 2010. The depth information obtained from the workspace can be converted to a 2 dimension (2-D) laser data. In order to this, the depth information is obtained and smoothed by means of different filters and a single horizontal line excided from the measurement data. Thus, a robot is able to behave like having a 2-D laser scanner on it [29]. Taking into account the bulky structure and expensiveness of laser sensors, this kind of sensors gives a good opportunity on 2-D scanning particularly if the edge determination or similar implementations of an environment is mainly aimed. A comparative work can be seen on [17] about the usage of Kinect -like a laser sensor.

According to the presented left-orientated algorithm, we have manipulated the robot generally in three directions; forward movement and left or reverse turns.

First: Without any orientation decision, the robot goes forward at a rate of $0.2 \text{ m}\cdot\text{s}^{-1}$ and fits the line by means of P-controller considering the deviation from the line center.

Second: Giving a decision on left-orientation, the robot turns left 90 degrees. In this case, forward speed is $0 \text{ m}\cdot\text{s}^{-1}$ while the robot turns fully 90 degrees in the direction of left.

Third: While the robot follows the line, it may encounter the parts that the line finishes. Under these circumstances, the forward speed of the robot is set to $0 \text{ m}\cdot\text{s}^{-1}$ and it turns reverse or 180 degrees.

With the help of these three movements, the robot could be manipulated autonomously with regard to the offered algorithm.

3.3. System Algorithm

Active SLAM is a combination of SLAM and autonomous exploration algorithms. With the presence of the developed algorithm, the robot does not need to navigate by a user or move randomly and SLAM for a given environment could be done autonomously by means of improved left-orientated line follower algorithm. Assuming the robot is started somewhere in the environment, the robot first investigates the existence of line and if there is, it goes forward according to the forward motion. The forward motion continues until any rotation. Having a rotation, the robot asks for it is left or right and then a different type of processes are developed in accordance with the offered algorithm (Fig. 6).

The image moments are thought of as features to determine orientation way or the function of what to do. An image moment is obtained in regard to the investigation of the density function of image pixels. A general physical moment expression can be described in Eq. (3):

$$m_{ij} = \int \int_S x^i y^j f(x, y) dx dy, \quad (3)$$

where S is the domain of workspace, i and j are the degrees of the function of f [15], [20] and [27].

The general moment statement is described for computer vision in Eq. (4) (continues case):

$$m_{ij} = \int \int_{R(t)} x^i y^j I(x, y) dx dy, \quad (4)$$

where $R(t)$ is the observed area by a camera, m_{ij} is the origin moments and $I(x, y)$ is intensity function.

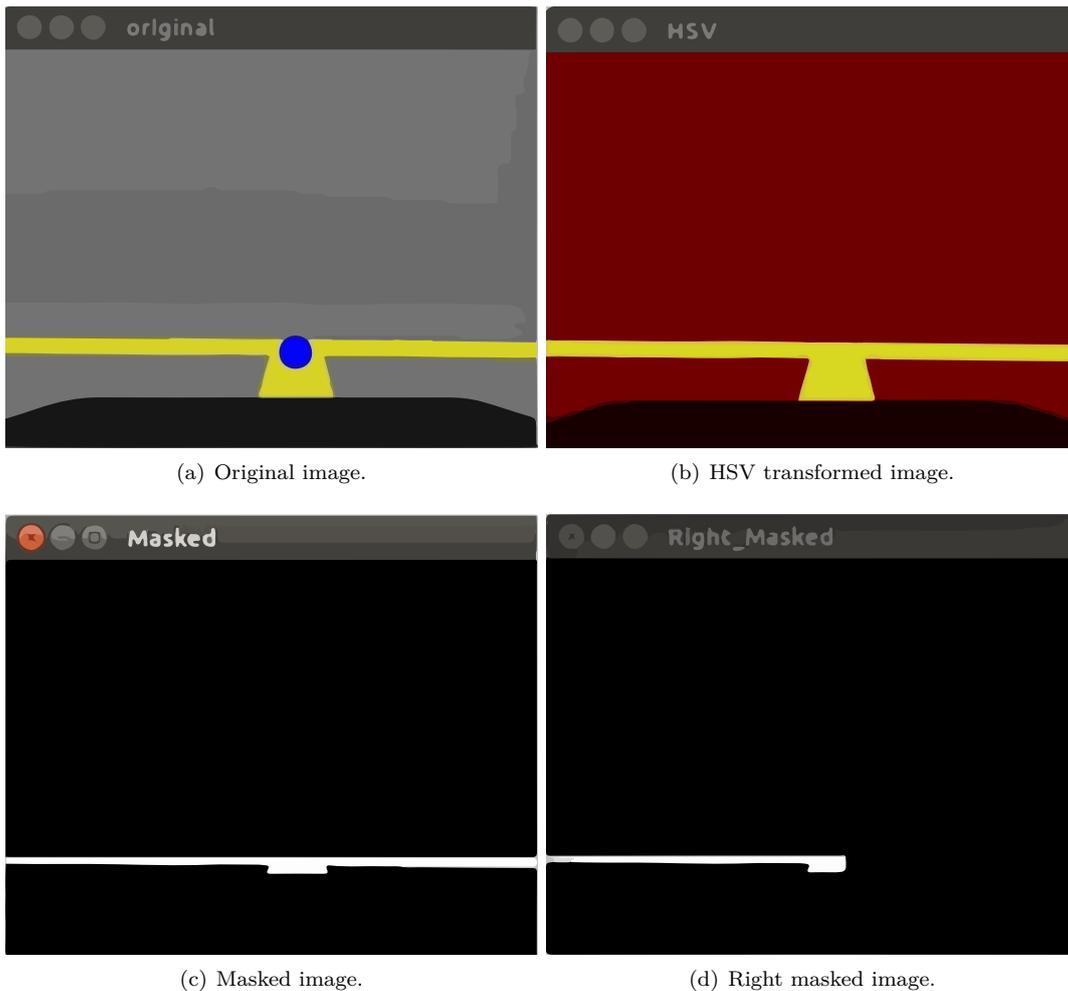


Fig. 2: Robot's views.

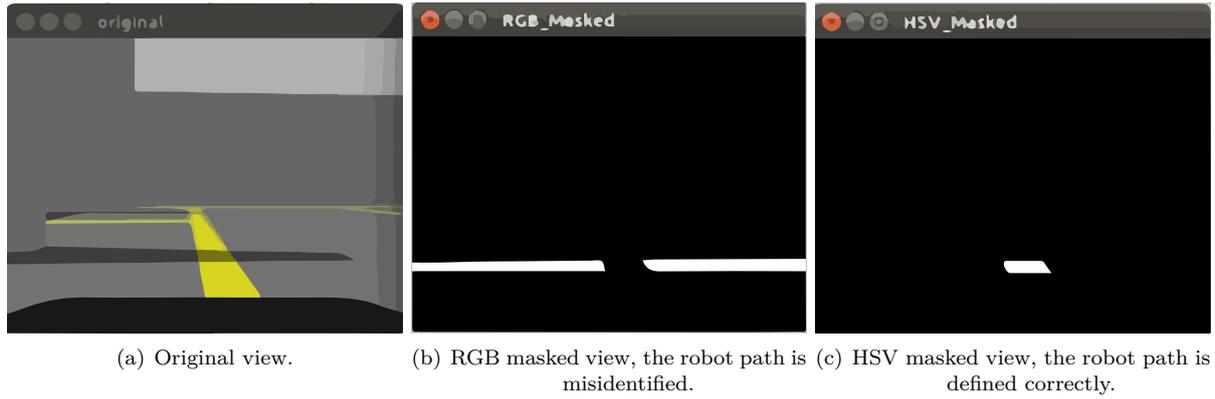


Fig. 3: Shady regions.

Algorithm 1 Overall algorithm scheme of the system

```

1: function NAVIGATION( ) ▷ See Fig. 6
2: end function
3: function GLOBAL MAP ← SLAM( )
4:   while depth image is acquired do
5:     Get the depth image
6:     Transform it into laser data
7:     Match the sequential scan
8:     Produce new particles  $\{x_t^{(i)}\}$ , using the  $x_{t-1}^{(i)}$  and  $\pi(x_t | Z_{1:t}, u_{0:t})$  ▷  $\pi(\cdot)$ : proposal distribution
9:     Weight the particles using the equation  $w^{(i)} = \frac{p(x_t^{(i)} | Z_{1:t}, u_{0:t})}{\pi(x_t^{(i)} | Z_{1:t}, u_{0:t})}$  ▷ importance weighting
10:    Replace the particles having lower to the higher ones ▷ resampling
11:    Estimate the map  $m_t^{(i)}$  for each particle pose  $x_t^{(i)}$  corresponding to  $p(m_t^{(i)} | X_{1:t}^{(i)}, Z_{1:t})$  ▷ map estimating
12:  end while
13: end function
14: while keeprunning do
15:   SLAM() ▷ part-1: constructing a map of the environment
16:   NAVIGATION() ▷ part-2: autonomous navigation
17: end while

```

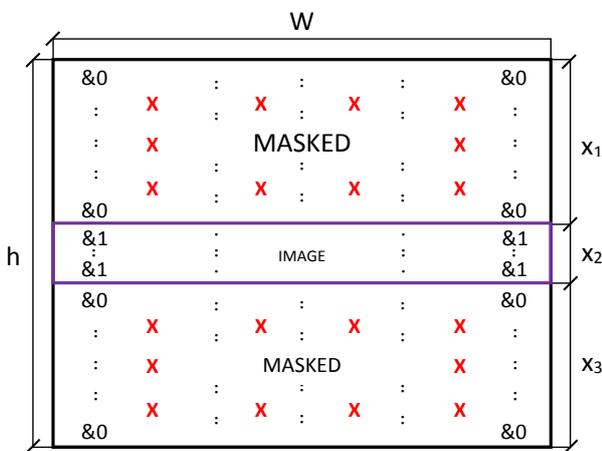


Fig. 4: Masked image.

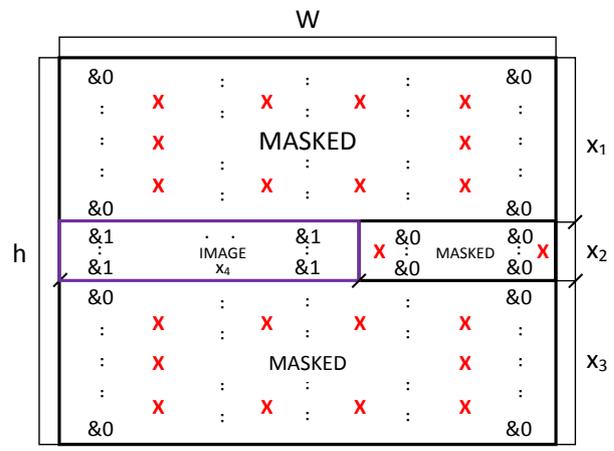


Fig. 5: Left masked image.

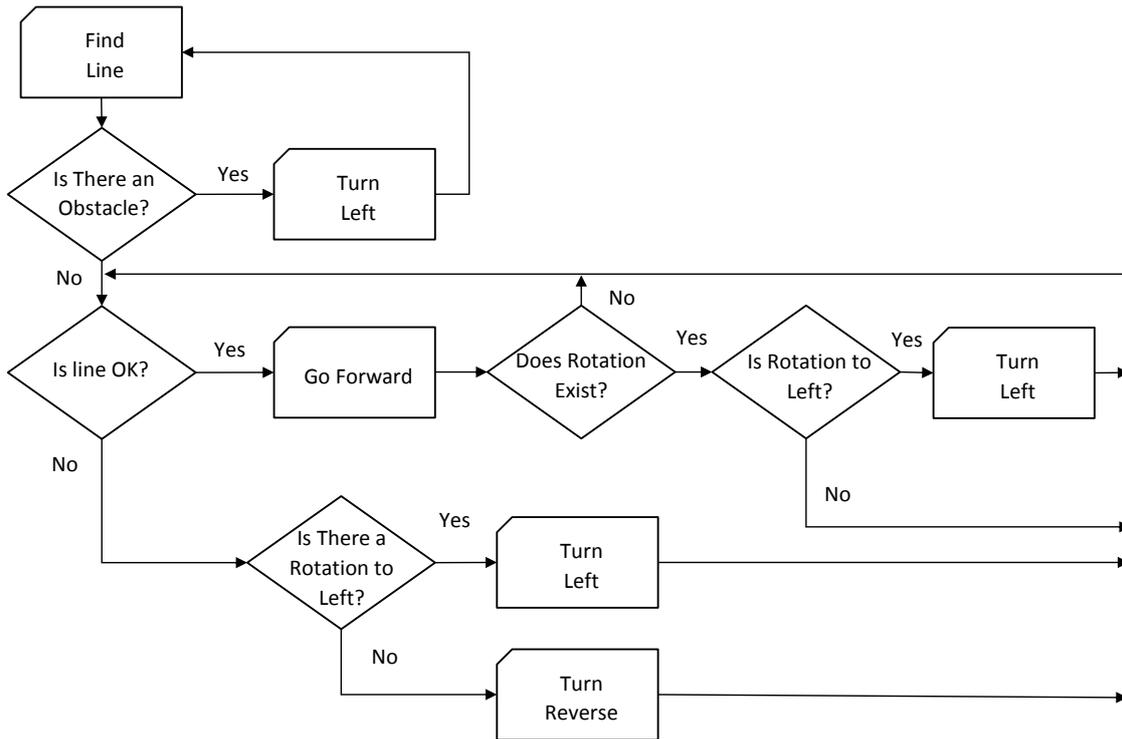


Fig. 6: Navigation flowchart snippet.

The centered moments of order $i + j$ as regards the centroid of the object is expressed by Eq. (5):

$$\mu_{ij} = \int \int_{R(t)} (x - c_x)^i (y - c_y)^j I(x, y) dx dy, \quad (5)$$

where, $c_x = m_{10}/m_{00}$, $c_y = m_{01}/m_{00}$ are the centroids of the 2-D object, (c_x, c_y) . For the discrete case, the integrals have been replaced by summations.

The line center is determined using the above-mentioned equations from the masked images (Fig. 4 and Fig. 5). This center is updated as the image is renewed. The deviation from this line center could be thought of an error Eq. (6):

$$e = c_x - c_i, \quad (6)$$

where, c_x is the image center on x-direction and c_i is the center of the line. The computed error is applied to the robot to correct the drift from the line. This scheme can also be regarded as a P-controller structure owing to the non-existence of neither past error nor predicted one. The experiments are also tested with the PID controller. However, it is observed that a P-controller body is sufficient for following the line smoothly [27].

It is benefited from ROS (Robot Operating System) to implement the study. ROS is a widely accepted and used framework because it can support the representation of the many real-time parameters. ROS-gMapping package is used to build a map of the environment and localize the robot within it [7], [9], [10], [13] and [32].

This method is a kind of grid-based SLAM that uses the Rao-Blackwellized particle filter. Particles hold the location of the robot and the map of the environment. Getting new observations lead to updated states in every iteration. A scan matching method is used for a measure of distance. According to technique, the robot location is calculated using the matching of sequential observations (see Alg. 1). The output of this method is the occupancy grid mapping of the environment. The algorithm mentioned in the study generally consists of two main blocks that one of them is the executing SLAM and another one is the navigation the robot autonomously. The pseudo-code of the algorithm is as in Alg. 1.

4. Results

In this study, a factory-like environment has created to carry out SLAM autonomously and to investigate the mentioned algorithm effectiveness. For this purpose, it is desired the robot to start a point referred in the map and followed the line in accordance with the offered algorithm. During the autonomous run, the robot has performed four main actions: find line, left, reverse, forward.

The frequency of the seen actions is pointed out in Fig. 7 as in the form of a histogram. As it is expected, the main action for the robot is the forward movements

while the less one is “find line”. This result also shows that when the robot finds the path, it sticks to the developed algorithm.

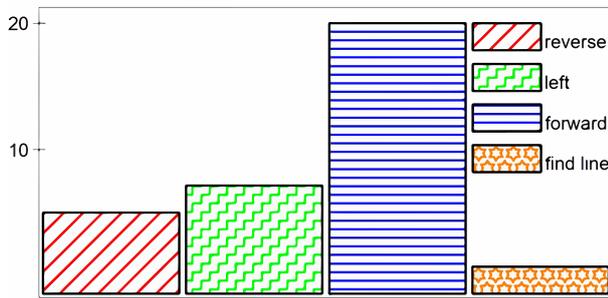


Fig. 7: The histogram bar graphic of the movements during the experiment.

According to the result of the trials, the robot can successfully build the map of the environment autonomously and robustly via the improved algorithm free from the effects of brightness, shadow etc. Figure 8 shows the map of the environment using the method and Fig. 9 points out the map created by user observed navigation.

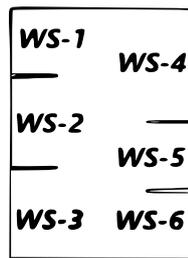


Fig. 8: The environment map via the offered method (WS: Workstations).

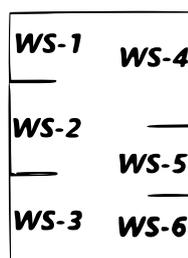


Fig. 9: The environment map via user navigation (WS: Workstations).

5. Conclusion

This paper has investigated an autonomous SLAM approach. SLAM is a highly important issue for mobile robots to implement their duties. A mapping of an environment might be done by user guidelines, random exploration or systematic algorithms. However,

the first two mentioned methods suffer from some serious limitations such as the absence of a user or not suitable observations for the user. In addition to this, randomly exploration could lead to a loss of time.

As to our knowledge, there is no comprehensive overview of recent research of vision-based for active SLAM. This study is designed to fill this gap by presenting a left-orientated navigation algorithm. Within this scope, a map of the environment has been built autonomously via the presented method without any necessity of user input or random process. In order to validate the presented method results, a factory-like environment is made up. The environment has the lines which describe the individual parts in a factory. The robot can manage to build a map of the environment by way of the mentioned algorithm. The evidence from the study indicates that the robot is able to succeed autonomous SLAM without a need of an external intervention. With the help of this method, a created map done by one robot can easily be shared with the other robots to save time or collaborate with each other.

This study has thrown up some questions in need of further investigation. For example, a future study investigating on different decision process using artificial intelligence would be very interesting and might improve the decision operation. On the other hand, further research might combine the other types of SLAM methods with this presented algorithm.

Acknowledgment

Authors are thankful to RAC-LAB (www.rac-lab.com) for providing the trial version of their commercial software.

References

- [1] BRAND, C., M. J. SCHUSTER, H. HIRSCHMULLER and M. SUPPA. Stereo-vision based obstacle mapping for indoor/outdoor SLAM. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Chicago: IEEE, 2014, pp. 1846–1853. ISBN 978-1-4799-6934-0. DOI: 10.1109/iros.2014.6942805.
- [2] COOK, G. *Mobile robots: navigation, control and remote sensing*. Hoboken: Wiley, 2011. ISBN 978-0-470-63021-1.
- [3] CRUZ, S., D. M. MUNOZ, M. E. CONDE, C. H. LLANOS and G. A. BORGES. A Hardware Approach for Solving the Robot Localization

- Problem Using a Sequential EKF. In: *International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum*. Cambridge: IEEE, 2013, pp. 306–313. ISBN 978-0-7695-4979-8. DOI: 10.1109/ipdpsw.2013.139.
- [4] DISSANAYAKE, G., S. HUANG, Z. WANG and R. RANASINGHE. A review of recent developments in Simultaneous Localization and Mapping. In: *6th International Conference on Industrial and Information Systems*. Kandy: IEEE, 2011, pp. 477–482. ISBN 978-1-4577-0032-3. DOI: 10.1109/iciinf.2011.6038117.
- [5] DURRANT-WHYTE, H. Uncertain geometry in robotics. *IEEE Journal on Robotics and Automation*. 1988, vol. 4, iss. 1, pp. 23–31. ISSN 0882-4967. DOI: 10.1109/56.768.
- [6] DURRANT-WHYTE, H. and T. BAILEY. Simultaneous localization and mapping: part I. *IEEE Robotics & Automation Magazine*. 2006, vol. 13, iss. 2, pp. 99–110. ISSN 1070-9932. DOI: 10.1109/mra.2006.1638022.
- [7] GALLARDO, N., K. PAI, B. A. EROL, P. BENAVIDEZ and M. JAMSHIDI. Formation control implementation using Kobuki TurtleBots and Parrot Bebop drone. In: *World Automation Congress (WAC)*. Rio Grande: IEEE, 2016, pp. 1–6. ISBN 978-1-8893-3551-3. DOI: 10.1109/wac.2016.7582996.
- [8] GRASA, O. G., E. BERNAL, S. CASADO, I. GIL and J. M. MONTIEL. Visual SLAM for Handheld Monocular Endoscope. *IEEE Transactions on Medical Imaging*. 2014, vol. 33, iss. 1, pp. 135–146. ISSN 0278-0062. DOI: 10.1109/tmi.2013.2282997.
- [9] GRISETTIYZ, G., C. STACHNISS and W. BURGARD. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Barcelona: IEEE, 2005, pp. 2432–2437. ISBN 0-7803-8914-X. DOI: 10.1109/robot.2005.1570477.
- [10] GRISETTI, G., C. STACHNISS and W. BURGARD. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. *IEEE Transactions on Robotics*. 2007, vol. 23, iss. 1, pp. 34–46. ISSN 1552-3098. DOI: 10.1109/tro.2006.889486.
- [11] HORNUNG, A., K. M. WURM, M. BENNEWITZ, C. STACHNISS and W. BURGARD. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*. 2013, vol. 34, iss. 3, pp. 189–206. ISSN 0929-5593. DOI: 10.1007/s10514-012-9321-0.
- [12] HU, G., W. TAY and Y. WEN. Cloud robotics: architecture, challenges and applications. *IEEE Network*. 2012, vol. 26, iss. 3, pp. 21–28. ISSN 0890-8044. DOI: 10.1109/mnet.2012.6201212.
- [13] JORGE, V. A. M., R. MAFFEI, G. S. FRANCO, J. DALTROZO, M. GIAMBASTIANI, M. KOLBERG and E. PRESTES. Ouroboros: Using potential field in unexplored regions to close loops. In: *IEEE International Conference on Robotics and Automation (ICRA)*. Seattle: IEEE, 2015, pp. 2125–2131. ISBN 978-1-4799-6923-4. DOI: 10.1109/icra.2015.7139479.
- [14] LEE, W. C., A. S. A. SALAM, M. F. IBRAHIM, A. A. A. RAHNI and A. Z. MOHAMED. Autonomous industrial tank floor inspection robot. In: *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*. Kuala Lumpur: IEEE, 2015, pp. 473–475. ISBN 978-1-4799-8996-6. DOI: 10.1109/icsipa.2015.7412238.
- [15] LONG, P., W. KHALIL and P. MARTINET. Robotic cutting of soft materials using force control & image moments. In: *13th International Conference on Control Automation Robotics & Vision (ICARCV)*. Singapore: IEEE, 2014, pp. 474–479. ISBN 978-1-4799-5199-4. DOI: 10.1109/icarcv.2014.7064351.
- [16] MONTEMERLO M., S. THRUN, D. KOLLER and B. WEGBREIT. Fast-SLAM:A factored solution to the simultaneous localization and mapping problem. In: *Proceeding Eighteenth national conference on Artificial intelligence*. Edmonton: AAAI, 2002, pp. 593–598. ISBN 0-262-51129-0.
- [17] PARK, C., S. KIM, D. KIM and S. OH. Comparison of plane extraction performance using laser scanner and Kinect. In: *8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. Incheon: IEEE, 2011, pp. 153–155. ISBN 978-1-4577-0723-0. DOI: 10.1109/ur.2011.6145951.
- [18] ROJAS, D., G. MILLAN, F. PASSOLD, R. OSORIO, C. CUBILLOS and G. LEFRANC. Algorithms for Maps Construction and Localization in a Mobile Robot. *Studies in Informatics and Control*. 2014, vol. 23, iss. 2, pp. 189–196. ISSN 1220-1766. DOI: 10.24846/v23i2y201407.
- [19] SCHMUCK, P., S. A. SCHERER and A. ZELL. Hybrid Metric-Topological 3D Occupancy Grid Maps for Large-scale Mapping. *IFAC-PapersOnLine*. 2016, vol. 49, iss. 15, pp. 230–235. ISSN 2405-8963. DOI: 10.1016/j.ifacol.2016.07.738.

- [20] SHAO, Q., J. HU, Y. FANG, W. LIU, J. QI and G. ZHU. Image moments based visual servoing of robot using an adaptive controller. In: *Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*. Tokyo: IEEE, 2016, pp. 57–61. ISBN 978-1-5090-1362-3. DOI: 10.1109/acirs.2016.7556188.
- [21] SIM, R., P. ELINAS, M. GRIFFIN, A. SHYR and J. J. LITTLE. Design and analysis of a framework for real-time vision-based SLAM using Rao-Blackwellised particle filters. In: *The 3rd Canadian Conference on Computer and Robot Vision*. Quebec: IEEE, 2006, pp. 21–21. ISBN 0-7695-2542-3. DOI: 10.1109/crv.2006.25.
- [22] SIM, R. and N. ROY. Global A-Optimal Robot Exploration in SLAM. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Barcelona: IEEE, 2005, pp. 661–666. ISBN 0-7803-8914-X. DOI: 10.1109/robot.2005.1570193.
- [23] SMITH, R. C. and P. CHEESEMAN. On the Representation and Estimation of Spatial Uncertainty. *The International Journal of Robotics Research*. 2016, vol. 5, iss. 4, pp. 56–68. ISSN 0278-3649. DOI: 10.1177/027836498600500404.
- [24] TAO, T., Y. HUANG, F. SUN and T. WANG. Motion Planning for SLAM Based on Frontier Exploration. In: *International Conference on Mechatronics and Automation*. Harbin: IEEE, 2007, pp. 2120–2125. ISBN 978-1-4244-0827-6. DOI: 10.1109/icma.2007.4303879.
- [25] THRUN, S., W. BURGARD and D. FOX. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In: *International Conference on Robotics and Automation*. San Francisco: IEEE, 2000, pp. 321–328. ISBN 0-7803-5886-4. DOI: 10.1109/robot.2000.844077.
- [26] THRUN, S., W. BURGARD and D. FOX. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*. 1998, vol. 31, iss. 1, pp. 29–53. ISSN 0885-6125. DOI: 10.1023/A:1007436523611.
- [27] WANG, J. and H. CHO. Micropeg and Hole Alignment Using Image Moments Based Visual Servoing Method. *IEEE Transactions on Industrial Electronics*. 2008, vol. 55, iss. 3, pp. 1286–1294. ISSN 0278-0046. DOI: 10.1109/tie.2007.911206.
- [28] WULF, O., D. LECKING and B. WAGNER. Robust Self-Localization in Industrial Environments Based on 3D Ceiling Structures. In: *International Conference on Intelligent Robots and Systems*. Beijing: IEEE, 2006, pp. 1530–1534. ISBN 1-4244-0258-1. DOI: 10.1109/iros.2006.282300.
- [29] ZUG, S., F. PENZLIN, A. DIETRICH, T. T. NGUYEN and S. ALBERT. Are laser scanners replaceable by Kinect sensors in robotic applications? In: *IEEE International Symposium on Robotic and Sensors Environments Proceedings*. Magdeburg: IEEE, 2012, pp. 144–149. ISBN 978-1-4673-2706-0. DOI: 10.1109/rose.2012.6402619.
- [30] JAFRI, S., J. IQBAL, H. KHAN and R. CHELLALI. A Unified SLAM Solution Using Partial 3D Structure. *Elektronika ir Elektrotechnika*. 2014, vol. 20, iss. 9, pp. 3–8. ISSN 2029-5731. DOI: 10.5755/j01.eee.20.9.8707.
- [31] YAMAUCHI, B. A frontier-based approach for autonomous exploration. In: *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*. Monterey: IEEE, 1997, pp. 146–151. ISBN 0-8186-8138-1. DOI: 10.1109/cira.1997.613851.
- [32] BERGEON, Y. and V. KRIVANEK. Design of an Open Hardware Bridge Between Robots and ROS/non-ROS Environments. *Advances in Electrical and Electronic Engineering*. 2018, vol. 16, iss. 2, pp. 199–204. ISSN 1804-3119. DOI: 10.15598/aeec.v16i2.2439.

About Authors

Akif DURDU has been an Assistant Professor with the Electrical-Electronics Engineering Department at the Konya Technical University (KTUN) since 2013. He earned a Ph.D. degree in Electrical-Electronics Engineering from the Middle East Technical University (METU), in 2012. He received his B.Sc. degree in Electrical-Electronics Engineering in 2001 at the Selcuk University. His research interests include mechatronic design, search & rescue robotics, robot manipulators, human-robot interaction, multi-robots networks and sensor networks. Dr. Durdu is teaching courses in control engineering, robotic and mechatronic systems.

Mehmet KORKMAZ received the B.Sc. and M.Sc. degrees from Department of Electrical and Electronics Engineering, Selcuk University, Konya, Turkey, in 2009 and 2013. Mr. Korkmaz has been currently studying for his Ph.D. at the same university and working as a visiting scholar at the Department of Civil, Environmental and Geodetic Engineering, The Ohio State University, Columbus, Ohio, USA. His main studies are fractional control theory and mobile robotics.