2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET) | 978-0-7381-0527-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/HONET50430.2020.9322812

A Consolidation of SLAM and Signal Reference Point for Autonomous Robot Navigation

I Made Murwantara
Informatics Department
Universitas Pelita Harapan
Tangerang, Indonesia

Benny Hardjono
Informatics Department
Universitas Pelita Harapan
Tangerang, Indonesia
benny.hardjono@uph.edu

Hendra Tjahyadi
Informatics Department
Universitas Pelita Harapan
Tangerang, Indonesia

Alfa Satya Putra
Faculty of Computer Science
Universitas Pelita Harapan
Tangerang, Indonesia

***Abstract*—Robot navigation, heavily, utilizes sensors to identify its current location and to guide forward into its designated position. This movement is achievable, mostly, via semi or fully autonomous navigation method that has a pre-built map. However, we argue that an autonomous robot navigation should not only provide route path to go to their target location, it should also has a feature to return to their starting position or reference location. In this work, an autonomous robot navigation has been built using combination of Simultaneous Localization and Mapping (SLAM) and signal of a device as reference point to find the original position. We make use of Turtlebot2 robot with Kinect sensor to create map, and Infrared beam to create the home signal guidance. In achieving target location, robot uses SLAM that has Odometry to track the distance, where at the same time sensing the presence of homing signal. Then, it returns to its home location by tracking the navigation map and home signal to find the best route path. Our result shows that the combination of SLAM and signal reference point has simplified the autonomous robot navigation management.**

***Index Terms*—robot, autonomous, SLAM, navigation, reference point**

I. Introduction

In general, an autonomous navigation provides capability for a robot to move to their designated location, without human intervention, by utilizing route path of a prebuilt map. Most autonomous navigation also, heavily, depends on technology such as SLAM that has a built map to point out its location in real-time. SLAM uses sensors to build its map and to track its location while moving within the existing map. Robot uses multiple sensing to guide its movement and to avoid surrounding obstacles by utilizing inputs from sensors and cameras. To enable an autonomous robot movement, a navigation map can be predefined using laser-based mapping to provide robot's vision [1], [2]. To reach an end-user's location, a service robot uses predefined navigation map as static information and camera or Lidar [3] to detect the possibility of new objects' location.

Robot technology has also been utilized with machine learning method to have a better performance. Kuhner et. al. [4] have developed a composition of service robot with deep learning method to increase the performance of robot-human interaction. Their approach has been tested in a real-world scenario that included situational changes which are common for human interaction. In developing service robots which is capable to learn human behaviour, Moro et. al. [5] have used machine learning dual-approach which are the Decision Tree and Reinforcement Learning, to support the robot's decision making process. Their approach has engaged human and robot with medium pro-activity to reduce decision space during their interaction.

A robot with autonomous navigation features is able to achieve its mission's goal via instruction, and it needs another set of instructions to accomplish any other missions. However, most autonomous navigation has a very complex management to provide new tasks, such as a returning home instruction and procedure, where a new location should be added to its mission or task. So that, a new navigation guidance or a kind of reference point to help reduce the complexity of autonomous navigation is needed. To our knowledge, only relatively a little research [6] has focused on using signal reference point to provide homing navigation.

In this work, we have provided a signal reference point that spread the beam indicator with range and position. The range provides information of distance between robot and its home location to engage with navigation mapping and location. Moreover, this reference point should also linked to the navigation map and robot movement, such as SLAM. The research questions of our work can be categorized into,

- RQ1 How to make a navigation map that links to a reference point?
- RQ2 Is there any possibility to reuse existing docking (recharge station) system as the reference point?
- RQ3 How to manage the signal reference point as the navigational guidance?

To answer these research questions, some methods should be established, especially, a technique that related to the identification of location and navigation. Moreover, the challenge even come up as the merges of SLAM and Signal Reference Point should not overlap with each other. To address this, we have modified the auto docking movement mechanism in the robot to only identify and stop in a certain distance, as long as the signal can be captured by the robot's sensor.

In the next section, the background knowledge of this paper will be presented. After that, the method to achieve this work will be explicated. In section IV, we will discuss the result of our work, followed then by a section on conclusion and future works.

II. Background

A. Robot Navigation

In many Robotic systems, navigation system is the ultimate aspect of managing robot's movement from current place to a designated area. One of the methods to accomplish this task is the SLAM [7] which consists of processing a map from specific environment and the robot's path from a set of observations.

In building a map, localization method [8] is used to obtain information regarding to robot's position and orientation. An accurate map is needed to provide a good position-estimation. Localization constructs the map of a given location so that a mapping of the robot's location can be inferred via a combination of sensors and map.

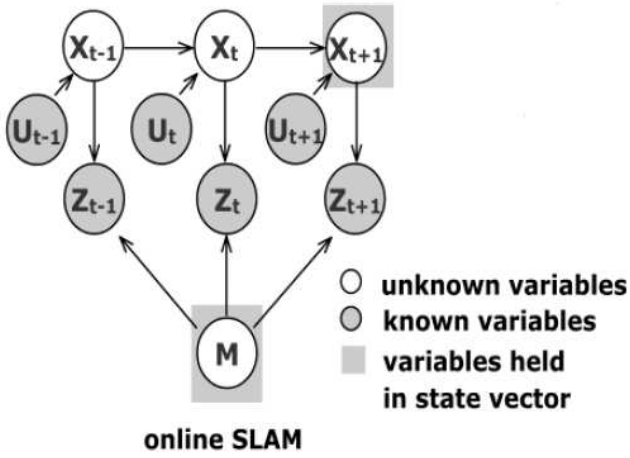

Fig. 1. SLAM - Online and Full [9].

As shown in Figure 1, online SLAM uses information that is obtained from sensors such as Camera and LIDAR. In this method, a map is built by scanning the environment and recapturing the robot's trajectory, then estimates most recent map and pose, where $p(x_t, m|z_{1:t}, u_{1:t}) = \iint ... \int p(x_{1:t}, m|z_{1:t}, u_{1:t})dx_1 dx_2 ... dx_{t-1}$.

This work uses online SLAM with Kinect [10] as its main sensor. We take advantage of a camera (RGB) and an Infrared depth sensing which allows us to have a real-time monitoring and processing within the robot system. Moreover, the Infrared depth sensor has capability to have depth measurement, where the light beams and depth sensor signals are read via emitter when they are reflected back to the sensor. Then, the infrared beam is converted as depth which measures the distance between an object and sensor.

B. Signal Reference Point

A signal reference point provides guidance via signal beam such as ultrasonic or infrared. In providing guidance, the reference point often marks the area with more than one straight line that focuses to a long and specific area. Another method uses wide beam to cover larger space to reduce robot movement and effective findings of the reference point.

As shown in Figure 2, we have splitted the signal beam to have two different ranges, zone1 dan zone 2, and transmitted using three different conditions, that are Right, Centre and Left. The range indicates maximum and minimum distance from the docking station. Originally, the docking station will guide the robot directly into the centre for recharging the robot. In this work, we have modified the reference point to have at least one meter distance from the docking location.

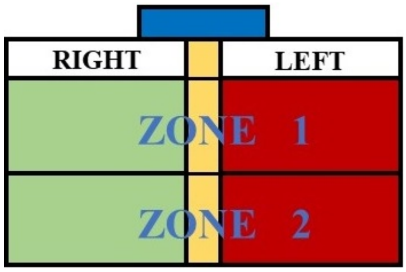

Fig. 2. Docking Signal Guidance.

III. Method

Our method consolidates the SLAM and Signal Reference Point for autonomous robot navigation. This method begins with setting up the reference position. Then, we merge the reference position and our generated navigation map.

A. Reference Position

In order to have reference position within our robot navigation, we have modified the original automatic docking algorithm of Turtlebot2. We have made adjustments and merged the signal from recharge docking station as the reference point for navigation helper. The modified algorithm is shown in algorithm 1. In this algorithm, we assume the navigation map is created based on Grid Mapping (Gmapping) library for SLAM [11] that uses laser scanner and Odometry [12] then they are computed as a map.

This algorithm uses SLAM location as input and then produces output as reference location. In order to initialize the process, the turning activity sets up as 0^o as to avoid unexpected activity, and not in docking position, also battery status should be higher than the recharging threshold. The process begins when SLAM navigation has brought the robot into a position where the Signal of Reference Point is detected, then the signal will navigate the robot into the reference point. Otherwise, if there is no signal identified then the navigation will always enable the SLAM.

When Reference Signal has been identified, the location will be included as $ZoneID$ where in this work we have splitted between Zone1 and Zone2 based on its nearest position to the Infrared source. To avoid complexity for this navigation helper, we do not move the robot once it has identified the zone location. Previous to that, in the making of the navigation map, we had included the two location of docking stations as part of the map.

Algorithm 1 Reference Point Guidance

Input: SLAM location
Output: reference location

$turned \leftarrow 0$
$docking \leftarrow$ false
$battery \leftarrow$ ok

while No Reference Signal **do**
  $Navigation \leftarrow ReadSLAM()$
  $IRSignal \leftarrow ReadIR()$
  $ZoneID \leftarrow Zone()$
  if IRSignal detected **then**
    $ZoneID \leftarrow IdentifyLocation$
    $IRSignal \leftarrow NavigationAlign$
    if $turned \geq 360^0$ **then**
      $ObstacleExist \leftarrow ObsDetected(IRSignal)$
    else
      $ObstacleExist$
      $FindNewPath(IRSignal)$
      $Reset(ZoneID)$
      if $MoveFind(IRSignal)$ **then**
        $FindLeft(IRSignal)$
        $FindRight(IRSignal)$
      else
        $Move \leftarrow ReadSlam()$
      end if
    else
      $ReturnError$
    end if
    $Turn(10^o)$
    $turned \leftarrow turned + 10^o$
  end if
  $ResetRequest(PressButton(B0))$
end while

When the robot has arrived in the designated location, and if according to the navigation map this location is not far from a reference point, it will search the signal by turning 360^o clockwise with stepping each 10^o. If there is no signal detected then the system will declare an assumption that there is an obstacle which has blocked the signal. To make sure the signal exists or not, robot will move to its left and make another turned 360^o, If it cannot find the reference signal then move to the right position and repeat the same procedure.

In this work the default navigation map is always on. In doing so, when the reference signal cannot be reached, the robot will move according to its task's mission using SLAM. Other than that, robot will stop and turned 10^o for each movement in a clockwise direction when it will try to find any indication of home location on its map based on the reference signal.

This reference point guidance will stop to operate when an intervention has been made by pressing the Button B0 on the robot's front panel, which will stop any current task. After that, the robot is ready to serve a new request from its web based application.

B. Merging the SLAM and Multi Signal Reference Position

We also have developed an initial merging scenario between SLAM and Multi Signal Reference Point. We report this as work in progress. This technique is an enhancement for the Reference Point Guidance as explained in Algorithm 1. In this method we have utilized several Signal Reference Points (SRPs) within our Navigation map. This algorithm calls the Reference Point Guidance as to resolve the signal movement and to get the functionality to reveal the signal reference point id from our two reference devices (robot's recharge stations).

Algorithm 2 Global Reference Point

Input: GlobalMap
Output: UpdatedRoutePath

$StepsToUpdate \leftarrow 0$
$NumberofSRP \leftarrow$ N
$RPG() \leftarrow$ ReferencePointGuidance

while Not Arrived to Destination **do**
  $ReadAlgRPG \leftarrow ReadMap()$
  $IRSignal \leftarrow ReadIR()$
  $Navigation \leftarrow ReadSLAM()$
  for $n = l$ to N **do**
    $SRP_{id} \leftarrow IdentifySRPidN_{id}$
    if IRSignal detected **then**
      $ZoneID \leftarrow IdentifyLocation$
      $IRSignal \leftarrow NavigationAlign$
      $SetAsNearest \leftarrow SRP_{id}$
    else
      $Move \leftarrow ReadSLAM()$
      $MoveNext \leftarrow RPG()$
    end if
  end for
  $ResetRequest(PressButton(B0))$
end while

The global reference point in algorithm 2, it begins by registering the number of signal reference point and importing the functionality within the algorithm of reference points 1. During the robot movement, the sensors try to capture all reference signal point along the path. When a reference point identified, the reference point ID will be added as the nearest point of reference. Then, the robot will keep moving to finish its task. After that, when it return to its nearest home location, the reference point will act as navigation points within the SLAM navigation map.

Another point of this algorithm is to provide a new mission insertion. As the robot returns to its home location, we can have new task inserted into the robot while it has reached one of the reference points.

IV. RESULTS AND ANALYSIS

In our experiment to run the reference point algorithm, we have built a navigation map using Gmapping and added an auto detection of the infra red signal as Signal Reference Point. The result of our work has not been very smooth when the robot has to make a straight movement, it always has to turn for 360^o to find the reference signal. However, the basic functionality to finish its task and return to home location have been achieved for one reference point. In this proof of concept, we make use of a single docking device as our Signal Reference Point.

It begins with position (a) where the robot is at standby mode and in its standby reference location, waiting for an incoming task. In this location, the robot has tracked the signal reference from the docking station, shown by a rectangle in red. As described in the algorithm 1, to update its current location, the robot will make a round movement which shows its current scanning results against the existing map. It is worth noticing that the current reference signal is not saved as part of the map, it is only a procedure on location checking. Then, in (b), after receiving a task to move to the target location, the robot predicts the distance and checks its battery status. When it detects the battery is in low level according to the power threshold level, the task will be delayed and the robot will recharge its battery into full level first. In this experiment, the battery power is well above the threshold level. The green line in Figure 3 (b) shows the path of prediction. When there is no location-change of objects within the reference map, the robot will move using the existing Odometry and map guidance, as in (c) and (d). The robot at the same time monitors the reference signal along the path. If there is a reference signal detected, then this location will be enabled as a new reference point. If there are a number of changes in a certain objects' location, the robot will compare its reference map and current camera and sensor information. Then, in (e), the robot stops in the front of the recipient or requester (the blue box), with gap of about 50 centimeters away from the requester-location.

When the robot has already reached the target, robot turns 10^o clockwise to facing toward to the target. It will stay at the last condition until button B0 on the robot is pressed. However, in this work we have created a rule where the robot will go back to its predestined location if for 5 minutes no one has pressed the work-completed button. As shown in Figure 3 (f), the robot autonomously prepares its path back to its standby location by making a feasible navigation path, as shown in green line. The first attempt (f) to create navigation path has failed as the robot has found the best path to use, which is the previous going path. After that, the robot returns back to its standby location, as shown in (h). When (i) it identifies the Signal Reference location, the robot aligns its position to move into the docking zone as described in Figure 2 .

V. CONCLUSION AND FUTURE WORKS

In this work, we have built a consolidated robot navigation using SLAM and Signal Reference Point. Our result shows that a robot may have capability to accomplished a task and go back to its home location without human intervention. This navigation method has reduced the complexity of robot's task, to use only one command to achieve its goal and go back to home location. Moreover, this method has also open the possibility to combine a navigation map with external devices, such as robot docking (charging) station.

For our future work, we would like to extend this method to handle multiple robots in a navigation map and to have a better fail-over mechanism.

ACKNOWLEDGMENT

This work has been funded by the Indonesian Ministry of Research, Technology and Higher Education (RistekDikti) Grant No. 26/AKM/PNT/2019. It is also supported by the research body (LPPM) of Universitas Pelita Harapan Grant No. 184/LPPM-UPH/V/2019, in which Benny Hardjono is the Team Leader and Corresponding Author.

REFERENCES

[1] F. Nobis, J. Betz, L. Hermansdorfer, and M. Lienkamp, "Autonomous racing," in *Proceedings of the 2019 3rd International Conference on Virtual and Augmented Reality Simulations - ICVARS '19*. ACM Press, 2019. [Online]. Available: https://doi.org/10.1145/3332305.3332319

[2] Stanford Artificial Intelligence Laboratory et al., "Robotic operating system," 2018. [Online]. Available: https://www.ros.org

[3] S. Gatesichapakorn, J. Takamatsu, and M. Ruchanurucks, "ROS based autonomous mobile robot navigation using 2d LiDAR and RGB-d camera," in *2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*. IEEE, Jan. 2019. [Online]. Available: https://doi.org/10.1109/ica-symp.2019.8645984

[4] D. Kuhner, L. Fiederer, J. Aldinger, F. Burget, M. Völker, R. Schirrmeister, C. Do, J. Boedecker, B. Nebel, T. Ball, and W. Burgard, "A service assistant combining autonomous robotics, flexible goal formulation, and deep-learning-based brain–computer interfacing," *Robotics and Autonomous Systems*, vol. 116, pp. 98–113, Jun. 2019. [Online]. Available: https://doi.org/10.1016/j.robot.2019.02.015

[5] C. Moro, G. Nejat, and A. Mihailidis, "Learning and personalizing socially assistive robot behaviors to aid with activities of daily living," *ACM Transactions on Human-Robot Interaction*, vol. 7, no. 2, pp. 1–25, Oct. 2018. [Online]. Available: https://doi.org/10.1145/3277903

[6] R. Quilez, A. Zeeman, N. Mitton, and J. Vandaele, "Docking autonomous robots in passive docks with infrared sensors and qr codes," *EAI Endorsed Trans. Self Adapt. Syst.*, vol. 1, p. e2, 2015.

[7] K. Berns and E. von Puttkamer, "Simultaneous localization and mapping (SLAM)," in *Autonomous Land Vehicles*. Teubner, 2009, pp. 146–172. [Online]. Available: https://doi.org/10.1007/978-3-8348-9334-5-6

[8] R. Lemus, S. Díaz, C. Gutiérrez, D. Rodríguez, and F. Escobar, "SLAM-r algorithm of simultaneous localization and mapping using RFID for obstacle location and recognition," *Journal of Applied Research and Technology*, vol. 12, no. 3, pp. 551–559, Jun. 2014. [Online]. Available: https://doi.org/10.1016/s1665-6423(14)71634-7

[9] L. Buonocore, S. R. B. dos Santos, A. A. Neto, and C. L. Nascimento, "FastSLAM filter implementation for indoor autonomous robot," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, Jun. 2016. [Online]. Available: https://doi.org/10.1109/ivs.2016.7535430

[10] S. Asteriadis, A. Chatzitofis, D. Zarpalas, D. S. Alexiadis, and P. Daras, "Estimating human motion from multiple kinect sensors," in *Proceedings of the 6th International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications*, ser. MIRAGE '13. New York, NY, USA: Association for Computing Machinery, 2013. [Online]. Available: https://doi.org/10.1145/2466715.2466727

[11] M. Filipenko and I. Afanasyev, "Comparison of various slam systems for mobile robot in an indoor environment," in *2018 International Conference on Intelligent Systems (IS)*, 2018, pp. 400–407.

[12] J. M. O'Kane, "Global localization using odometry," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006, pp. 37–42.

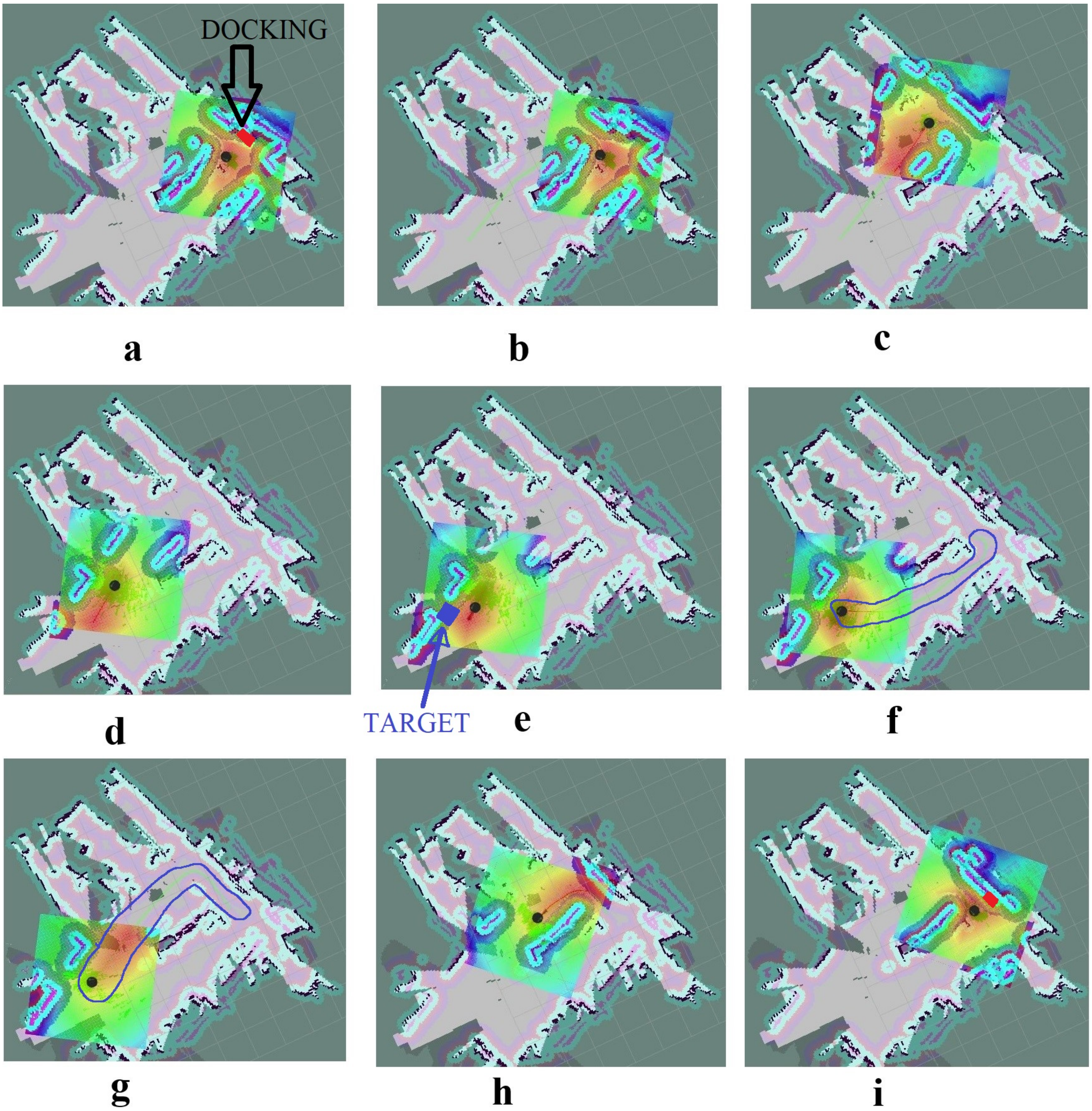

Fig. 3. Autonomous Movement Using Consolidated SLAM and Signal Reference Point.