Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 3980 – 3993

doi:10.3906/elk-1807-335

A novel map-merging technique for occupancy grid-based maps using multiple robots: a semantic approach

Akif DURDU[1*], **Mehmet KORKMAZ**[2]
[1]Department of Electrical and Electronics Engineering, Faculty of Engineering and Natural Sciences, Konya Technical University, Konya, Turkey
[2]Department of Civil, Environmental and Geodetic Engineering, The Ohio State University, Columbus, Ohio, USA

Received: 31.07.2018 • **Accepted/Published Online:** 08.04.2019 • **Final Version:** 18.09.2019

Abstract: Map merging is a noteworthy phenomenon for cases such as search and rescue and disaster areas in which the duration is quite significant when gathering information about an environment. It is obvious that the total mapping time decreases if the number of agents (robots) increases. However, the use of multiple agents leads to problems such as task allocation schemes and the fusing of local maps. Examining the present methods, it is generally observed that the common features of local maps have been found and the global map is formed by obtaining related transformation between local maps. However, such implementations may be risky when local maps have symmetrical areas. Hence, a novel and semantic approach has been developed to solve this problem. The developed method counts on the reliability level of feature points. If relevant feature points are trusted, local maps are merged according to the best point or points. The simulation results from a robot operating system and a real-time experiment support the proposed method's efficiency, and mapping can be performed even for environments that have symmetrical similar parts and the task time can thus be reduced.

Key words: Map merging, occupancy grid maps, semantic algorithm, simultaneous localization and mapping

1. Introduction

From a robotics perspective, maps allow robots to determine how the world around them appears [1–3]. Robots can fulfill assigned tasks and discover new points by virtue of maps. Several approaches have been advanced in the literature to solve the robotic mapping or simultaneous localization and mapping (SLAM) problem for both single and multiple robots [4–9]. As well as obtaining maps where a robot can carry out its assigned tasks, the duration of building maps also plays a vital role in some circumstances such as search and rescue (SaR) operations in disaster situations. This is because finding entrapped victims or applying intervention strategies is mostly based on the current maps of the disaster areas; therefore, the duration of obtaining maps has to be as fast as possible. A common intervention in SaR areas is fulfilled by humans or trained dogs to attain quick results. However, this can be dangerous when the area contains dangerous waste such as nuclear fallout. Therefore, robots can be used as substitutes for humans or trained dogs in such cases. It is clear that performing interventions with a single robot in large-scale zones is time-consuming. Therefore, it is generally done with a team of robots. As a result, mapping and intervention times can be reduced when robot teams are used rather than a single one [10–12]. However, one of the most challenging problems with multirobot teams is the combination of partial maps supplied by each robot [12–14]. The usual aim is to find common features

*Correspondence: akifdurdu@gmail.com

in local maps and transformations between robots or local maps to acquire a global map of an environment. When the transformation or common features are known, merging can be easily fulfilled. However, the correct detection of common features in local maps is a substantial problem [14–16]. The algorithms defined in the literature can easily diverge from the consistent global map if an environment has symmetrical features. This study aims to determine the solutions of complications regarding common features used in existing map-merging problems. Accordingly, a new and semantic approach is presented to problems encountered in existing methods. Through the developed method, common feature points are determined in each local map and whether these points are trusted is evaluated. According to the reliability of features, the transformation points of the maps are determined and the global map is constituted in consideration of these points. Section 2 reviews some methods established in the literature about map merging. The following section explains the evaluation process of whether a feature point can be trusted and related semantic algorithms regarding merging local maps are explained in detail. Later, the results are presented with some experiments. In the last section, findings obtained from the results of the study are evaluated and remarks are made regarding future studies.

2. Related work

The problem in robotic mapping is to obtain a spatial model of a robot's environment via its sensors. The map of an environment is built incrementally from each observation [17]. This is usually known as a robotic mapping problem and has been analyzed for more than three decades. On the other hand, map merging can be considered as a subset of the robotic mapping phenomenon. In other respects, map merging is the creation of a consistent global map from local ones that are produced by different agents [13, 14, 18]. The map-merging problem has dynamic and static approaches. In dynamic approaches, global and local maps are produced simultaneously. Contrary to this, a global map is created after each local agent explores its own region in static approaches [19]. As well as type of approaches, another important parameter of map-merging problems is the type of the map. Feature [20] or occupancy-based [13] map representations are often in the foreground and occupancy-based maps are frequently preferred by researchers due to their being both true and metric representations of the world and presenting information about unexplored regions [21]. These advantages have led to occupancy-based maps being used in this study. In a similar manner to different map types, map merging can also be characterized by various approaches as based on features, iterative closest points (ICP), or overall-direct optimization [21–23]. This paper uses feature-based methods to match the regions of points of interest because the environment has sufficient properties for extraction. Although there have been several different approaches to map fusion, the problem of map merging is actually a result of multirobot analysis. Continuous measurement of the robot's environment in mapping applications could cause residual errors because the used sensors always have some noise. Even though this is not a significant problem for small-scale environments, the cumulative error, which is a result of successive residual errors, must be deliberated in large-scale environment mapping. Hence, one of the most remarkable advantages of multirobot mapping is that it decreases residual errors when mapping large-scale areas. In addition, another advantage is that the multirobot approach reduces the total mapping duration [14, 24]. Thus far, several studies have examined the merging of occupancy grid maps. Lee et al. [15] addressed a sinogram-based method to simulate offline occupancy grid maps. Matching between maps has come through the use of multiple analyses with sinograms. Radon transformation is used to extract sinograms; according to the results, the developed approach gives better results than a single sinogram and other methods. Tsardoulias et al. [25] investigated the map-merging problem in conjunction with RFID tag localization and topological information. Their approach is mainly based on obstacles' and RFID tags' locations

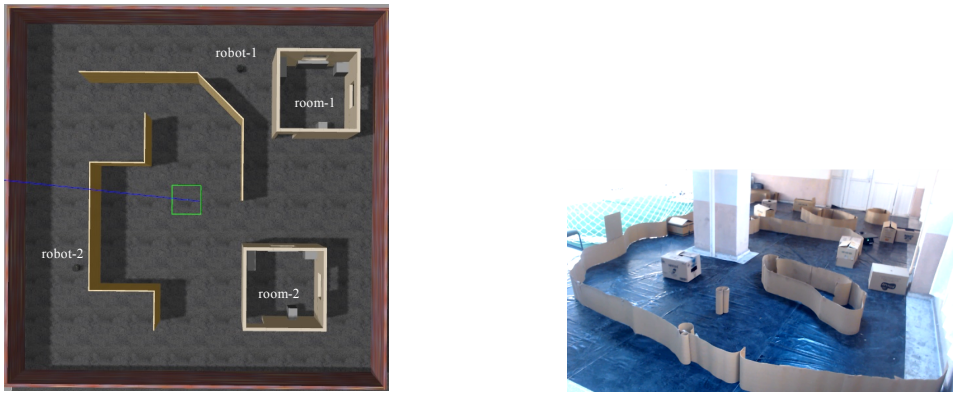
and ICP algorithm transformation. The steps they applied are first of all to find the alignments between related obstacles; later, common RFID tags' pose rotations are determined. They also improved this process to obtain exact alignment using the ICP algorithm and refined the map by a modified blurring process in order to prevent inconsistencies. Ferrão et al. [26] focused on merging multiple agent local maps using affine transformations. They performed their experiment on two robots but mentioned that the process could be enhanced by repeating the same procedure so that multiple robot map merging is achieved. They also used SIFT features to determine common points on the local maps. According to their comments, sometimes the method did not work properly since the algorithm could not find the right corresponding point sets and this led to misalignment and improper matching between local maps. They mentioned these kinds of limitations in their conclusion. As we witness this drawback, we also improve this method by adding reliability criteria in the selection of features in our study so that the algorithm does not only choose the key features but also evaluates them regarding defined reliability criteria. Saeedi et al. [27] handled the map-merging problem along with SLAM application. They extended a single SLAM approach to the multiple one and enhanced a novel schema for map merging, based on a multistep process. The steps they used basically consist of some image processing and transformation matrix tuning. They chose edges and segmentation blocks created based on edges as features for local maps and applied detection, segmentation, and verification processes on the local maps. According to the segments histograms they determined whether the feature was a good candidate or not. If their application scenario is examined, it is clearly seen that local maps are obviously different from each other and this brings to mind the question of what happens if the environment has symmetrical areas. This study and its references also presented a detailed review about occupancy grid map merging. Birk and Carpin [13] emphasized the importance of multirobot mapping and implied that occupancy grid maps are suitable for robotic applications. One main advantage of their method is that the instantaneous poses of the robots allied with one another are unimportant. Adaptive random walking is used in the searching process. The process of matching between local maps is performed using a similarity function based on the comparison of similar regions in local maps. However, the study does not contain any clues for symmetrical regions. Topal et al. [10, 12] benefited from using a scale-invariant feature transform (SIFT)-based method to merge different local occupancy grid maps. The aim is to combine partial maps created in the results of the mapping task assigned to robots. The study's main idea is to merge maps by using SIFT features. However, there has been no previous discussion about cases in which local maps have symmetrical parts, as in our work. Robustness is not handled on such occasions. In this manner, it is obvious that the algorithm they presented will most likely make a mistake when merging local maps as shown in Section 4. At this point, our improved method provides a solution using the reliability criteria for feature selection.

3. Improved method

3.1. Test environment

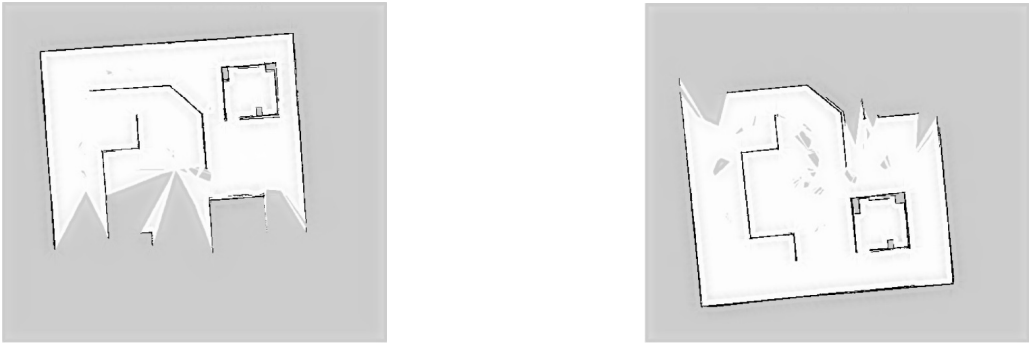
Two different environments are used for the experiments being carried out in this study. Symmetrical regions are introduced in the test environment (room-1 and room-2); hence, the enhanced algorithm can be validated. There are two robots in the environment, and both build their own local maps simultaneously. The particle filter-based Rao–Blackwellized decomposition technique (RBPF) is employed to create a map of the environment demonstrated in Figure 1a and Figure 1b. According to this method, each particle carries its own map and particles are omitted regarding observations and measurements that are taken from the environment. The output of the algorithm is an occupied grid map of the environment. Local maps that robots make for the environment in Figure 1a are pointed out in Figures 2a and 2b. Similarly, partial maps for the real-time case

are indicated in Section 4.2. The pieces observed in local maps are colored to symbolize information; black regions are occupied, white regions are free, and gray regions are spaces that have not yet been visited by any robot.



(a) The simulation environment (b) Real-time environment

Figure 1. The experimental environments.



(a) (b)

Figure 2. Local maps for the simulation experiment: (a) the first robot's partial map of the environment and (b) the second robot's partial map of the environment.

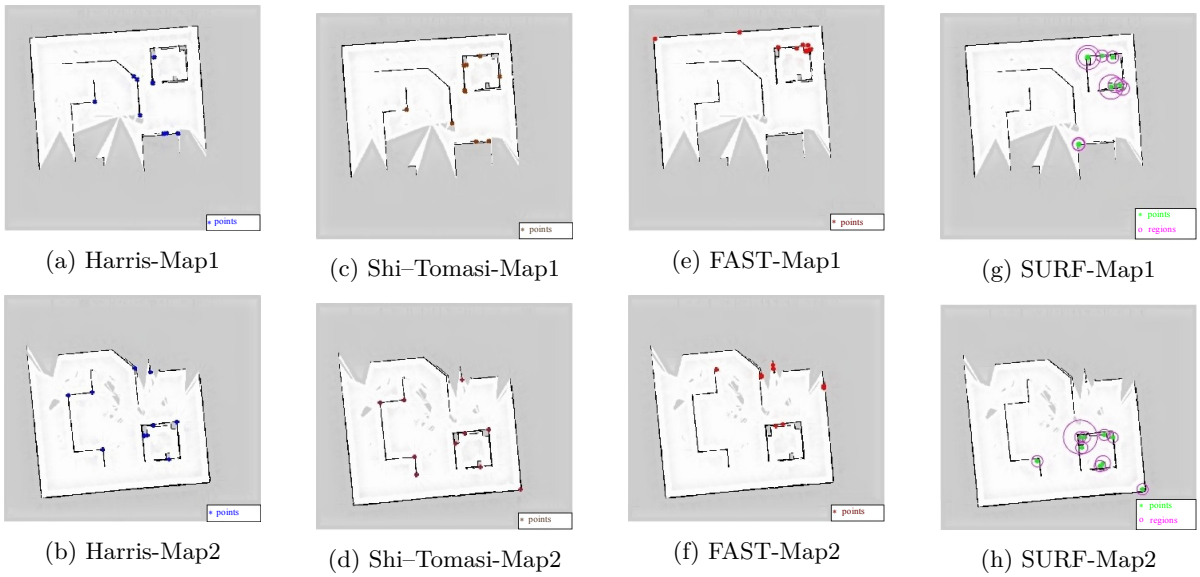
3.2. Feature extraction process

Various features can be extracted from the images through the numerous methods in the literature. These features are important specifications for the image and give it a unique character. Some of the best known and used methods are Harris, Shi–Thomasi, SIFT, speeded up robust features (SURF), and the features from accelerated segment test (FAST). One of the oldest but most used methods is the Harris corner detection method [28–30], given in Eq. (1):

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u, y+v) - I(x,y)]^2. \tag{1}$$

According to this method, the value of E is investigated by a small change in u and v. It is obvious that the change is close to 0 if the frames are similar. On the contrary, because there will be more variation at the corner points, the value of E reaches its maximum. The values that maximize E are searched with the help of

the Taylor series. The answers for the associated points are measured and it is investigated whether the feature is a corner point or not. One of the powerful upsides of this method is that the direction of the frames is not important. On the other hand, the new Shi–Tomasi [31] method makes a small modification to the scoring function suggested in the Harris corner detection method. They improved Harris's method by measuring the response of feature points in a different way.



(a) Harris-Map1 (c) Shi–Tomasi-Map1 (e) FAST-Map1 (g) SURF-Map1

(b) Harris-Map2 (d) Shi–Tomasi-Map2 (f) FAST-Map2 (h) SURF-Map2

Figure 3. Feature points relating to first and second maps, respectively: (a, b) the Harris method, (c, d) the Shi–Tomasi method, (e, f) the FAST method, (g, h) the SURF method.

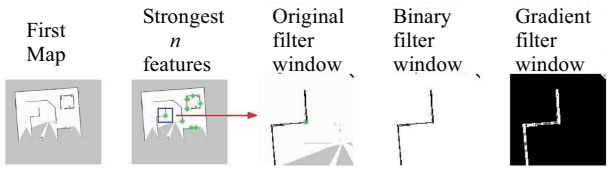


Figure 4. Feature points on the local maps and a region of interest around a selected feature point.

The other well-known and newly discovered feature extraction method is FAST [29, 32]. This method makes a notable improvement in computation time. For this reason, it is generally preferred in video processing and machine learning applications. SURF is another feature recognition method and is based on the sum of two-dimensional Haar wavelets. It can be thought of as a development of the SIFT method. Authors claim that this method is more robust and faster than SIFT [29, 30, 33, 34]. In our study, the best n features belonging to local maps are shown in Figure 3, in which the feature points are extracted according to the Harris (Figure 3a, Figure 3b), Shi–Tomasi (Figure 3c, Figure 3d), FAST (Figure 3e, Figure 3f), and SURF methods (Figure 3g, Figure 3h), respectively. In addition to the feature points, a feature frame is necessary for the detection of the trusted points and similarity comparison between the local maps. As such, a window (L) is created according to the filter size (f) around these feature points, and a corresponding feature window is obtained (Figure 4). Through these windows, it is investigated whether the feature is trusted or not and the similarity between these windows is sought to detect the similar points between images as suggested in Section 3.3.

3.3. Description of trusted points and semantic algorithm

The semantic approach used in the article is based on whether the relevant feature points are trusted or not. It introduces a threshold value for the determination of feature point reliability. The basic idea underlying the identification of trusted points is to highlight the features at the inner points since the robot is more likely to see those areas. With this approach, the uncertainty in the filtered windows is investigated by examining free, occupied, and unknown cells. It has the ability to find the common features in the inner parts by eliminating the outer points or untrusted features. Depending on the developed threshold description, the feature point is either eliminated or labeled as trusted (Algorithm 1).

Definition: Let N and M be real positive numbers and I or M be a map function of size N $\times$ M. *I(x, y)* has free, occupied, or unknown points that are accepted if the belief is positive, negative, or unknown, respectively. If this value equals zero, information about the grid is not available, and it is also called an unknown cell. P is a set of feature points according to the feature extraction method for the image I and p is a point that denotes a feature relevant map, I (2):

$$p = \{p(n,m)|p(n,m) \in P, n \leq N, m \leq M \wedge n,m \in \mathbb{Z}^{+}\}\forall P(u,v) \subseteq I_{N\times M}(x,y). \tag{2}$$

L is a subwindow for the detection of the trusted points (3):

$$L = \{L_{f\times f}|L(x_c,y_c) = p(n,m) \wedge p(n,m) \in P\}\forall L_{f\times f} \subseteq I_{N\times M}, \tag{3}$$

where x_c and y_c are the centers of the L filter-window and f is a filter size.

By expressions, t is a trusted point only if condition (4) is provided:

$$t \leftarrow \left\{ \begin{array}{ll} \frac{\#L(x,y)|_{unknowncells}}{f} \leq f, & \text{p is trusted;} \\ \frac{\#L(x,y)|_{unknowncells}}{f} > f, & \text{p is not trusted.} \end{array} \right\} t \in P \wedge \forall L \subseteq I_{N\times M}(x,y). \tag{4}$$

If Eq. (4) is examined, it is clearly seen that trusted points are dependent on the number of unknown cells and related features. The choice of the filter size has a twofold significance for acquiring trusted points and matching the local maps associated with trusted points. First, there may not be any unknown cells around the feature points as a result of the tiny selected filter size.

On the other hand, when the filter size is large, the computation load may increase. When the number of unknown cells in the feature window increases, it may become difficult to define the feature point. Second, the correlations between the feature windows are examined to match the feature points found in local maps. This correlation value is likely to be the same for irrelevant features when the filter size is too small. Conversely, for the conditions of large selected values of *f*, the computation time may increase and the matching of feature windows may be improper due to the fact that the local occupancy grid maps have small differences. Therefore, a selection of an appropriate filter size is required. According to our approach, the filter size can be determined based on some general environmental features, namely map resolution, distance between corridors, and confidence interval. The corridors and distance values of some parts of the environment are indicated in Figure 5 in meters and pixels. The filter size, *f*, can be formulated as a pixel as in Eq. (5):

$$f[px] = c_w[m] \times r[m/px]^{-1} + \alpha[px], \tag{5}$$

where f is the filter size, c_w is the most seen corridor width, r is the resolution of the map, and α is the confidence interval. The value of f is set by increasing or decreasing the confidence interval.

ALGORITHM 1: The feature selection and reliability evaluation

```
Input  : M^i                                   /* local maps generated by i^th robot */
Output: T_i                                    /* a set of trusted points for each map M^i */
for all M^i do
    P_i ← function FEATUREEXTRACTION(M^i)      /* determine the features */
    for all p_i do
        T_i ← If (function TRUSTEDPOINTDETECTOR(p_i))   /* test for trusted points */
        end
    end
end
function TRUSTEDPOINTDETECTOR(p_i)
  L_i ← split M around p_i at the size of filter
  w_i ← calculate L_i reliability scale
if w_i > threshold then
    p_i is trusted point
else
    discard p_i
end
  end function
```

The critical part for the determination of the filter size is that neither large nor small values are set. If the environment is established as in Figure 1a and the corridor width is considered to be about 3 m, the filter size is obtained as 60 according to Eq. (5). The experimental results reveal that the value of the confidence interval, α, can be selected in the range of +/- 10 [px]. Under these circumstances, the filter size, f, can be selected in the range of 50–70. In order to merge the partial maps while considering the reliability features, the semantic algorithm scheme is used (Algorithm 2).

ALGORITHM 2: The semantic map merging.

```
Input  : M^i, T_j        /* Local maps generated by n^th robot and a set of trusted points belong to local maps */
Output: M^F                                    /* Global single map generated by algorithm */
for all M^i do
    for all t_j do
        ∀ t_j ∈ T_j
        L_{i,j} ← split M around tj at the size of filter
        ∇f_{i,j} ← ( ∂L_{i,j}/∂x , ∂L_{i,j}/∂y )   /* calculate the gradients */
    end
end
for all i=1:#{T_j} ∈ M^1 do
                                               /* for two local maps */
    X = f_{1,i}
    for all j=1:#{T_j} ∈ M^2 do
        Y = f_{2,j}
        C(i,j) ← ( cov(X,Y) / (s_x s_y) )
    end
end
arg max_{i,j} C= {i,j | C(i,j) ≥ C(k,l), ∀ C(k,l) ∈ C }
m_p ← C(i,j)
if m_p is positively high then
    match the local maps using the m_p trusted point
else
    'No matching'
end
```

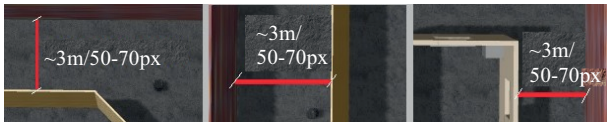


Figure 5. Some corridor views of environment.

4. Implementation and results

Experiments are carried out in both created and real-time environments that are built to validate the accuracy of the enhanced method. In order to form a global map, an exchange between the robot's local maps is achieved based on the specified method [10]. Within this scope, the methods stated in the literature attempt to fuse the local maps obtained from different robots. The features acquired from the local maps are used to create a global map via Harris, Shi–Tomasi, FAST, and SURF methods.

4.1. Simulation results

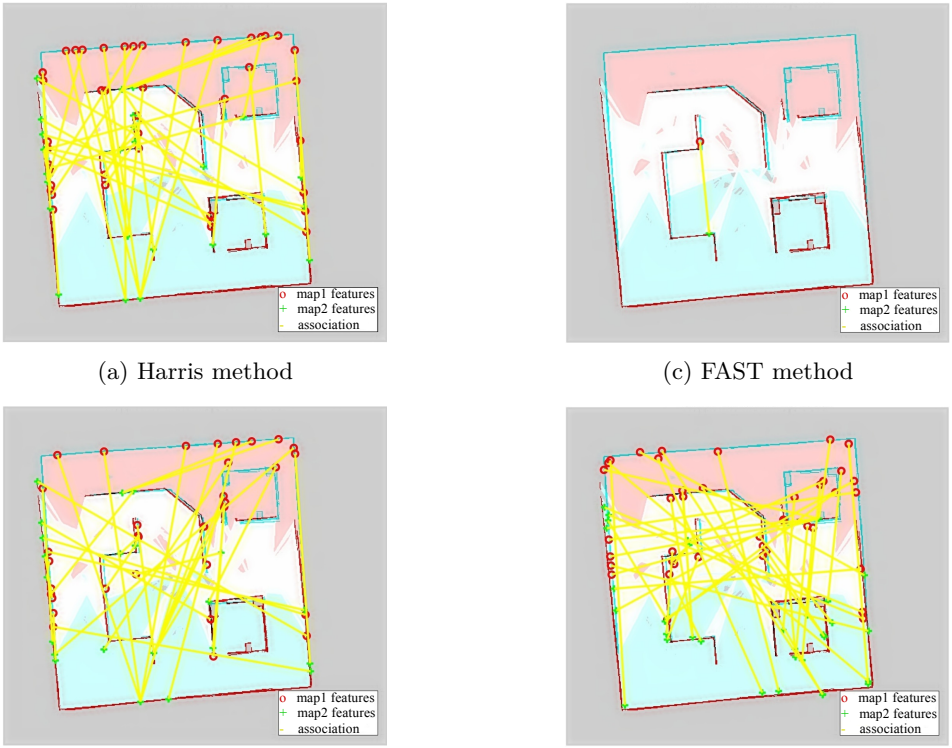
It is clear that within the created environment, there are two identical rooms in different regions. These rooms are a good landmark from the perspective of finding a common feature but they may also lead to confusion. Figures 6a–6d illustrate the existing methods' map fusion results. By carefully examining Figure 6, the handicap referred to in Section 2 is encountered in the existent methods described in the literature. One consequence of having symmetrical parts is matching different features as though they were the same. This problem stems from the fact that local maps have symmetrical parts. Although many features can be found in the local maps via the existing methods, it is perceived that room-1 and room-2 are the same in the matching process and these points are thought to be similar. Thus, this process leads to an inaccurate merging and the developed semantic method offers a new solution. This enhanced method is based on merging maps from a semantic point of view by assuming that the robots are likely to navigate or can see the inner regions of the environment more clearly than the outer ones.

A map of the environment using the Harris and Shi–Tomasi corner detection method, which is improved with the presented semantic approach, is obtained as in Figures 7a and 7b. The experiments are run 10 times for the comparison of the time efficiency and the status of the global map as seen in Table 1. When the Harris and Shi–Tomasi methods are combined with the enhanced semantic technique, a global map is acquired using the trusted feature points. The time comparison showed no huge differences between the methods. On the contrary, a global map could not be constituted when using the FAST and SURF features. As expected, the global map was obtained in 65 s through a multirobot approach that was improved using the mentioned algorithm, while 105 s was needed in the single-robot case. In addition to this, one of the robots was assigned to navigate from one point to another. For a fair comparison, this robot is initiated from the same place in both situations. According to the elapsed times, the robot needs 125 s to fulfill the task. However, this task time decreases to 85 s in the multimap case.

4.2. Real-time results

In addition to the simulation study, the enhanced method is also tested in a real-time environment to validate the efficiency. In order to do that, first of all, local maps are obtained (Figures 8a and 8b) and existing methods in the literature are applied to obtain the merged global map. The pitfalls of the existing methods have been clearly recognized once again and these methods fuse the local maps improperly (Figures 9a–9d).

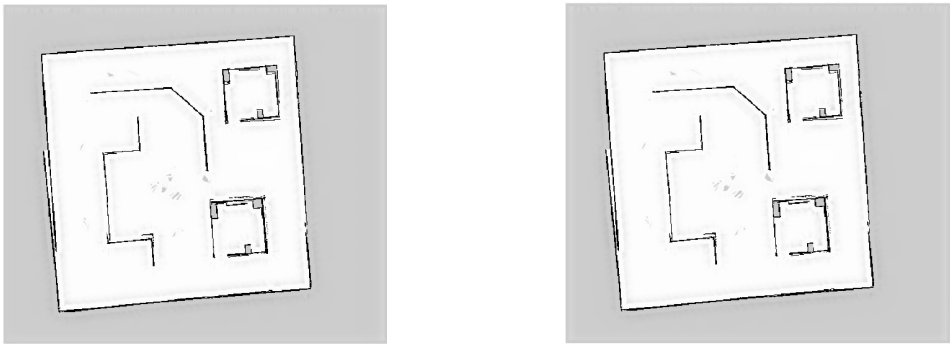
As a result of this, the improved semantic approach is applied to the local maps of the robots. The filter



(a) Harris method (c) FAST method

(b) Shi–Tomasi method (d) SURF method

Figure 6. Keypoint matching.



(a) Harris method (b) Shi–Tomasi method

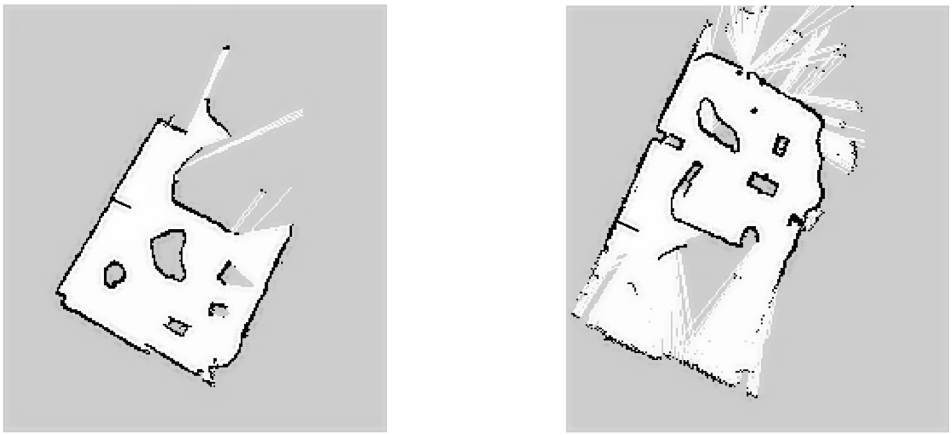
Figure 7. Result map with regard to improved algorithm.

size is determined according to Eq. (5). The narrowest corridor size is measured at about 20 in pixel standard. After that, the same procedure and algorithm are applied to the robots' local maps. According to the results, it is observed that the real-time experiment (Figure 10) is consistent with the simulation and the developed algorithm shows a clear advantage over previous approaches.

Different from the simulation experiment, the global map is obtained only with the Harris detection based method. There are several explanations for this finding. First of all, the extracted features may not be compatible with the developed algorithm since the enhanced method is based on features extracted from the maps. Another reason might be the nuance between partial maps which is natural outcome of the multi-robot

Table 1: Average elapsed time and matching state for the simulation experiment.

Algorithm	Time for merging	Matching state
Harris	1.713244	Yes
Shi–Tomasi	1.712669	Yes
FAST	1.418407	No
SURF	1.544664	No



(a) First robot's partial map of environment

(b) Second robot's partial map of environment

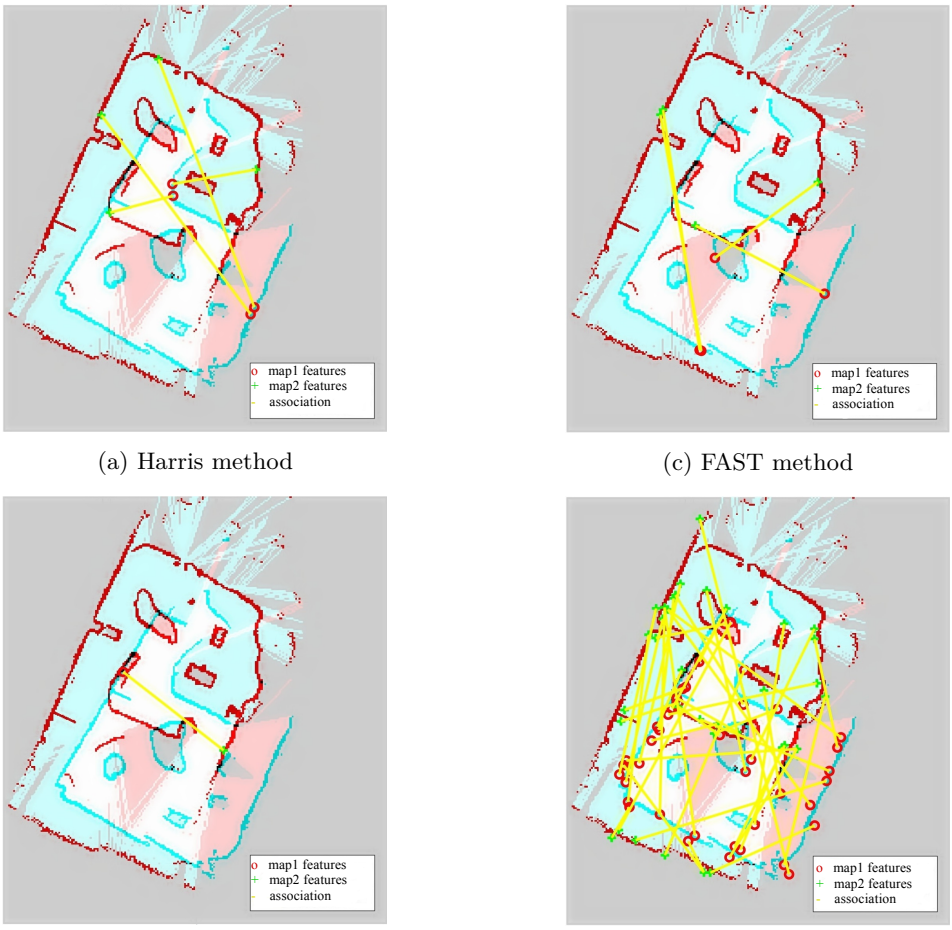
Figure 8. Local maps for the real-time experiment.

mapping. Despite this, we can still state that our semantic outline could be expanded by new kind of feature extraction methods. Table 2 highlights the time for seeking the global map and final matching states of the methods. It is should be noted that time for searching the global map is consistent with each study and there are no significiant difference between the methods.

Table 2: Average elapsed time and matching state for the real-time experiment.

Algorithm	Time for merging	Matching state
Harris	2.610178	Yes
Shi–Tomasi	2.393639	No
FAST	2.438914	No
SURF	2.306784	No

The real-time experiment durations are parallel with the simulation one. As expected, while one robot can create the map of the environment in 260 s, a robot team could perform the same task in 140 s. At the same time, for a single-robot case, a robot needs 284 s to complete its task of going from one place to another, while this time is reduced to 164 s when using multiple robots.



(a) Harris method

(c) FAST method

(b) Shi–Tomasi method

(d) SURF method

Figure 9. Keypoint matching for real-time experiment.

5. Conclusions

This study investigates the cooperation of multiple robots in map-merging tasks. To do this, a mapping task for created and real-time environments is assigned to the robots. Although the map formed by a single robot is sufficient in many cases, it is undesirable for the mapping process to require a long time period in several conditions. As such, multirobot mapping was used in such conditions to decrease the processing time. Within this framework, the integration of different local maps obtained from different agents into a global map and the methods to follow them are emphasized. To that end, studies have been carried out to merge the maps with these existing methods in the literature. Although these methods are useful for combining images that are closely related, it has been observed that they fail to properly create maps that have symmetrical parts as in the created environment. The existing methods try to match the symmetrical sections as if they were the same. For this reason, it is necessary to develop a new semantic approach to merge maps with symmetrical sections. The solution of this problem and the further development of existing methods are the main contributions of this study. The related procedures have been given and a successful application has been realized in this paper. It is assumed that the robots see the inner regions more clearly than the outer ones. Therefore, in order to determine those features, the reliability criterion is defined and all features are scored using Eq. (4), which is given in

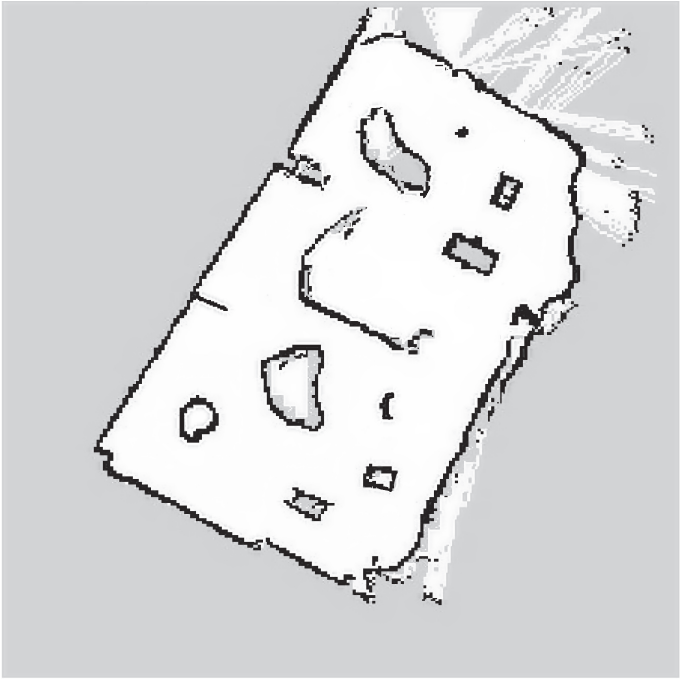


Figure 10. Result map with regard to improved algorithm for the Harris method.

Section 3.3. Once these points have been determined, the same features are produced in both local maps with the help of a correlation between windows formed around these feature points. Although the generated local maps are grid-based and there are nuances even in the same sections, the evidence from the results indicates that the aforementioned method has been able to successfully perform the merging task. Besides this, it is emphasized that there is no trusted point to merge the maps if merging fails. In addition to this, four different feature extraction methods are also compared. Despite the advantages and drawbacks of these methods in terms of the number of features, trusted features are only found in the Harris and Shi–Tomasi methods for the simulation experiment and in the Harris method for the real-time one. The study has elicited some questions in need of further investigation. For instance, the machine learning algorithm incorporates a trusted point detection process and an effective solution to this problem may enhance our solution.

Acknowledgment

The authors are thankful to RAC-LAB (www.rac-lab.com) for providing the robotic equipment for this study.

References

[1] Riazuelo L, Tenorth M, Marco DD, Salas M, Gálvez-López D et al. Roboearth semantic mapping: a cloud enabled knowledge-based approach. IEEE Transactions on Automation Science and Engineering 2015; 12 (2): 432-443. doi: 10.1109/TASE.2014.2377791

[2] Wallgrün IJ. Hierarchical Voronoi Graphs: Spatial Representation and Reasoning for Mobile Robots. Berlin, Germany: Springer Heidelberg, 2010.

[3] Khan S, Ahmmed MK. Where am I? Autonomous navigation system of a mobile robot in an unknown environment. In: 2016 5th International Conference on Informatics, Electronics and Vision; Dhaka, Bangladesh; 2016. pp. 56-61.

[4] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I. IEEE Robotics & Automation Magazine 2006; 13 (2): 99-108. doi: 10.1109/MRA.2006.1638022

[5] Quang HP, Quoc NL. Some improvements in the RGB-D SLAM system. In: 2015 IEEE RIVF International Conference on Computing & Communication Technologies-Research, Innovation, and Vision for the Future; Can Tho, Vietnam; 2015. pp. 112-116.

[6] Yoo JK, Kim JH. Gaze control-based navigation architecture with a situation-specific preference approach for humanoid robots. IEEE/ASME Transactions on Mechatronics 2015; 20 (5): 2425-2436. doi: 10.1109/TMECH.2014.2382633

[7] Thrun S, Montemerlo M, Koller D, Wegbreit B, Nieto J et al. Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association. Journal of Machine Learning Research 2004; 4 (3): 380-407.

[8] De Silva O, Mann GK, Gosine RG. An ultrasonic and vision-based relative positioning sensor for multirobot localization. IEEE Sensors Journal 2015; 15 (3): 1716-1726. doi: 10.1109/JSEN.2014.2364684

[9] Schmidt A. Multi-robot, ekf-based visual slam system. In: Computer Vision and Graphics, ICCVG 2014; Cham, Switzerland; 2014.

[10] Topal S, Erkmen I, Erkmen AM. A novel multirobot map fusion strategy for occupancy grid maps. Turkish Journal of Electrical Engineering and Computer Sciences 2013; 21 (1): 107-119. doi: 10.3906/elk-1106-18

[11] Huang WH, Beevers KR. Topological map merging. International Journal of Robotics Research 2005; 24 (8): 601-613.

[12] Topal S, Erkmen I, Erkmen AM. A novel map merging methodology for multi-robot systems. In: Proceedings of the World Congress on Engineering and Computer Science; San Francisco, CA, USA; 2010. pp. 383-387.

[13] Birk A, Carpin S. Merging occupancy grid maps from multiple robots. Proceedings of the IEEE 2006; 94 (7): 1384-1397. doi: 10.1109/JPROC.2006.876965

[14] Bonanni TM, Della Corte B, Grisetti G. 3-d map merging on pose graphs. IEEE Robotics and Automation Letters 2017; 2 (2): 1031-1038. doi: 10.1109/LRA.2017.2655139

[15] Lee HC, Roh BS, Lee BH. Multi-hypothesis map merging with sinogram-based PSO for multi-robot systems. Electronics Letters 2016; 52 (14): 1213-1214. doi: 10.1049/el.2016.1041

[16] Wang K, Jia S, Li Y, Li X, Guo B. Research on map merging for multi-robotic system based on rtm. In: 2012 International Conference on Information and Automation; Shenyang, China; 2012. pp. 156-161.

[17] Thrun S. Robotic mapping: a survey. In: Lakemeyer G, Nebel B (editors). Exploring Artificial Intelligence in the New Millennium. San Francisco, CA, USA: Morgan Kaufmann, 2003. pp. 1-36.

[18] Samir S, Elouardi A, Samir B, Belhocine M. Vehicle localization systems: towards low-cost architectures. Turkish Journal of Electrical Engineering and Computer Sciences 2016; 24 (4): 2010-2027. doi: 10.3906/elk-1402-260

[19] Aragues R, Sagues C, Mezouar Y. Map Merging. Parallel and Distributed Map Merging and Localization: Algorithms, Tools and Strategies for Robotic Networks. New York, NY, USA: Springer International Publishing, 2015.

[20] Aragues R, Cortes J, Sagues C. Distributed consensus on robot networks for dynamically merging feature-based maps. IEEE Transactions on Robotics 2012; 28 (4): 840-854. doi: 10.1109/TRO.2012.2192012

[21] Li H, Tsukada M, Nashashibi F, Parent M. Multivehicle cooperative local mapping: A methodology based on occupancy grid map merging. IEEE Transactions on Intelligent Transportation Systems 2014; 15 (5): 2089-2100. doi: 10.1109/TITS.2014.2309639

[22] Aragues R, Sagues C, Mezouar Y. Feature-based map merging with dynamic consensus on information increments. Autonomous Robots 2015; 38 (3): 243-259. doi: 10.1007/s10514-014-9406-z

[23] Tamas L, Goron LC. 3d map building with mobile robots. In: 20th Mediterranean Conference on Control & Automation; Barcelona, Spain; 2012. pp. 134-139.

[24] Lee HC, Lee BH. Enhanced-spectrum-based map merging for multi-robot systems. Advanced Robotics 2013; 27 (16): 1285-1300. doi: 10.1080/01691864.2013.819609

[25] Tsardoulias E, Thallas A, Petrou L. Metric Map Merging using RFID Tags & Topological Information. arXiv: 1711.06591, 2017.

[26] Ferrão VT, Vinhal CDN, Cruz G. An occupancy grid map merging algorithm invariant to scale, rotation and translation. In: 2017 Brazilian Conference on Intelligent Systems; Uberlandia, Brazil; 2017. pp. 246-251.

[27] Saeedi S, Paull L, Trentini M, Li H. Occupancy grid map merging for multiple robot simultaneous localization and mapping. International Journal of Robotics and Automation 2015; 30 (2): 149-157. doi: 10.2316/Journal.206.2015.2.206-4028

[28] Harris C, Stephens M. A combined corner and edge detector. In: Proceedings of Fourth Alvey Vision Conference; Manchester, UK; 1988. pp. 147-151.

[29] Chong NS, Yau HK, Mou LDW. Visual detection in omnidirectional view sensors. Signal, Image and Video Processing 2015; 9 (4): 923-940. doi: 10.1007/s11760-013-0528-0

[30] Proença H. Performance evaluation of keypoint detection and matching techniques on grayscale data. Signal, Image and Video Processing 2015; 9 (5): 1009-1019. doi: 10.1007/s11760-013-0535-1

[31] Shi J. Good features to track. In: 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; Seattle, WA, USA; 1994. pp. 593-600.

[32] Rosten E, Drummond T. Machine learning for high-speed corner detection. In: European Conference on Computer Vision; Graz, Austria; 2006. pp. 430-443.

[33] Bay H, Tuytelaars T, Van Gool L. SURF: Speeded up robust features. In: European Conference on Computer Vision; Graz, Austria; 2006. pp. 404-417.

[34] Durdu A, Korkmaz M. Autonomously simultaneous localization and mapping based on line tracking in a factory-like environment. Advances in Electrical and Electronic Engineering Journal 2019; 17 (1): 45-53. doi: 10.15598/aeee.v17i1.3048