Scientific Journal of Riga Technical University
Computer Science. Applied Computer Systems
2011
Volume 47

Map Merging in the Context of Image Processing

Ilze Andersone, *Riga Technical University*

***Abstract* - The area of map merging is tightly connected to the area of image processing. Usually metric maps created by robots are represented as occupancy grids. It is easy to apply algorithms used in image processing to this kind of map representation. The image processing subfield that is closest to the map merging is the image registration. It can be assumed that metric map merging methods similarly to the image registration methods consist of three components: feature space, search strategy and similarity metric. Algorithms from image processing can also be used in map merging for map preprocessing. The goal of this paper is to explore similarities between the fields of map merging and image processing and to determine how the results of this research can be used for the development of a map merging framework and consequently new map merging approaches.**

Keywords: Image processing, map merging, robotic mapping

I. Introduction

One of the fundamental problems in mobile robotics is the environment mapping problem. Robots need to be able to construct a map of the environment and to use it for navigation. As the use of robot teams becomes more and more popular, the issue of robot coordination becomes important. If multiple robots are used for the exploration of the environment, their collected information has to be fused into one general global map. The fusion of the map information from multiple robots into one global map is called map merging [1].

The problem of map merging is not a simple one. Different robot teams use different map representations. The most popular map representations used in robotic mapping are topological maps and metric maps. Topological maps are the connectivity graphs where vertices represent the objects of the environment and edges represent the paths between those objects [1]. Metric maps represent the environment as a set of geometric information acquired from the sensor measurements [2]. Each of these representations requires a different map merging approach. It is more complicated to merge metric maps because no additional structural information is available.

Besides the map representation the knowledge about the reference frames of the robots must be taken into account. If each robot knows the location of the other robots the map merging problem can be solved relatively easily. This is called distributed mapping [3]. The map merging is harder to implement when the reference frames of the robots are unknown. In this case the overlap of the maps is not known and it may not even exist [2]. In this paper only the case of merging metric maps in unknown reference frames is considered.

The most popular representation of a metric map is an occupancy grid [4]. Occupancy grid is an array where the occupancy value of each cell represents whether the location it relates to is a free space or an obstacle [5]. The occupancy grids can be thought of as images where the occupancy value is represented by a color [3]. The occupancy grid can also be seen as a grayscale image where each cell carries only intensity information.

The similarity between occupancy grids and images suggests that occupancy grids can be processed similarly to the images i.e. by using image processing algorithms. In the image processing context the map merging problem can be defined as the overlapping of two or more images where the images are grayscale and no information about the relative reference frames of the images is available.

The goal of this paper is to explore similarities between the fields of map merging and image processing and to determine how the results of this research can be used for a map merging framework and consequently the development of new map merging approaches.

The structure of this paper is as follows. At first the related work is discussed in chapter II. Then the image processing and specifically image registration is considered in the chapter III. In chapter IV the map merging is analyzed in the context of image processing. In chapter V the importance of image and map preprocessing is emphasized and three preprocessing methods examined. Chapter VI shows how the inferences acquired in the previous chapters can be used to design a map merging framework and the prototype of this framework is introduced. Finally, the conclusions are drawn and possible future work defined.

II. Related Work

Robot teams originated in late 1980s [6], however only during the last ten years has the multi-robot mapping question been intensively studied. This may be due to the fact that, although robot teams offer multiple advantages over single robot platforms, several new problems, specific to the multi-robot mapping, arise [7].

As Konolige et al notes “map merging is an interesting and difficult problem, which has not enjoyed the same attention that localization and map building have” [1]. Although several years have passed since this statement was made and the map merging problem has received more attention, most of the published papers on the matter are descriptions of specific map merging methods.

There are several authors who have become acquainted with and shortly described the situation in the map merging research area [1], [3], [9], [10], [11]. The authors who have addressed the map merging in the image processing context most are Birk and Carpin in [3]. They point out that map merging in the context of image processing is the problem of moving one of the images around until a part of it is aligned with a similar part in another image. This problem is harder than that of image registration because the region of overlap is unknown and has to be identified in two maps. Although this problem is also present in the image stitching, the occupancy grids lack rich textures that are common in photographs.

However, there are no papers where map merging is comprehensively analyzed in the image processing context. No authors have tried to develop a map merging framework that would help in the development and research of the map merging methods.

III. Image registration

The goal of image processing is to interpret the images and acquire some previously unknown information [12]. Image processing has many subfields and a lot of vastly different applications [12]. However, not all image processing subfields can be directly related to the task of map merging.

A crucial step of any image processing task that combines the final information from various data sources is the image registration [13]. Image registration is the process where several images of the same place that are taken at different time, from different viewpoints and with different sensors are overlapped [13]. It can be assumed that all image registration methods consist of three components [14] (Fig. 1):

- Feature space,
- Search strategy,
- Similarity metric.

The feature space is the collection of features in the image that are used for image comparison. The search strategy chooses the strategy accordingly to image deformations. Usually the information source in searching is the collection of features. The similarity metric determines the successfulness of the transformation [14].

It can be seen that feature identification is an important part of image registration. However, if the environment the robots have mapped is unstructured, then the identification of the features is considerably more complicated. Unstructured environments are characterised by the lack of features of one type. It does not mean that it is impossible to identify any features in the maps in unstructured environments. Such maps can be interpreted as images, too, and it is possible to identify objects in them. The main problem is that the types of the objects in the environment are not known.

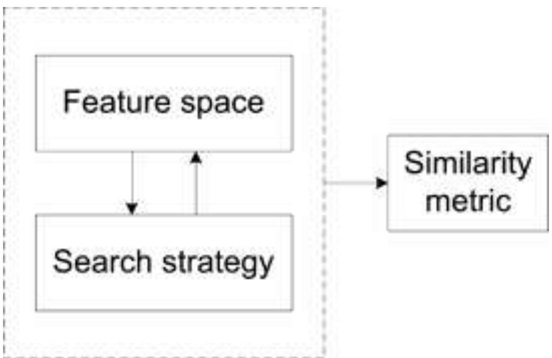

Fig. 1. The components of the image registration and map merging methods

IV. Map merging components

The essence of the map merging task is the combination of two local maps into one global map. It is easy to relate the map merging problem to image registration task if two robots are considered to be independent data sources and their local maps – images. The occupancy grid can be seen as a grayscale image where each cell carries only intensity information.

It is easy to adapt algorithms used in image processing for the occupancy grid map representation. As map merging and image registration methods have so many similarities, it can be assumed that metric map merging methods similarly to image registration methods consist of three components: feature space, search strategy and similarity metric (Fig. 1). Similarity metric is an independent component that can be adjusted to any map merging approach. On the contrary, feature space and search strategy are often tightly interconnected.

Like image registration methods all map merging methods compute potential map transformations and overlap the maps based on these transformations. Not all map merging methods search for specific objects in the maps. Nevertheless all of them use some local and global information about maps to be merged. It is necessary to analyze the maps and acquire some additional information as the transformation space can be very large and the evaluation of all possible transformations can take a very long time.

A. Feature space

Different feature spaces can be utilized for the merging of metric maps. Konolige was one of the first researchers who proposed to use features for the map merging [8]. His proposal was to manually mark objects in the maps and use those objects for the purpose of the map merging. It is obvious that such approach is not applicable if a robot team is attempting to create the map of the environment autonomously.

Amigoni in [10] uses angles between segments as a feature space. However, a specific type of maps is required in this case. Ho [11] supplements the maps with visual information from a video camera during the mapping so that the maps are easier to merge later. Lakaemper modifies the maps so that they contain only simple lines [15]. These lines are then used as a feature space for map merging. Carpin uses the Hough spectrum in map merging [16]. The Hough spectrum is able to tell which directions of lines are the most common in the maps.

All of these feature spaces can be helpful in acquiring the global map faster. Nevertheless all of them also possess some flaws. Some of them [10], [11] require a specific type of maps that are not always available. Others [10], [15], [16] require a lot of straight lines in the maps for the map merging approach to be effective.

There is a map merging approach that is able to merge occupancy grid maps and does not require a specific environment. Birk and Carpin in [17] and [3] use an image similarity heuristic as a feature space. The main disadvantage of this approach is that the feature space has to be computed in each iteration and therefore the time required for the map merging rapidly grows when the maps become larger. Also it is easy for the search algorithm to get stuck in a local maximum while using this feature space.

The current situation in the field of metric map merging shows that the choice of the feature space is still a problem. No single approach is applicable to every situation. All feature spaces currently used in map merging can be divided into two groups:

- Local feature spaces – objects (edges, lines, corners etc.) are identified in the maps and used for map merging [15], [10].
- Global feature spaces – global information (specters, image similarity metrics) about the maps is acquired and used for map merging [16], [17].

Both of these feature space groups have been applied to map merging with some degree of success. No group has proved to be definitely better than the other.

B. Search strategy

Search strategies used in map merging have one common characteristic: they are based on the acquired feature space. For example, search strategy described in [16] is based on Hough specters acquired from the maps. Between these specters the correlation is found and consequently the most promising rotations acquired. Strategy used in [15] is based on the comparison of specific lines.

The fact is that different search strategies are used for different search spaces and that means that it is often impossible to align the feature space and search strategy components from two different map merging methods. However, there are cases when such alignment is possible i.e. when search strategies or feature spaces of both map merging approaches are similar or easily modified.

C. Similarity metric

The similarity metric determines how successful the map merging result is. Although the introduction of similarity metric does not guarantee correctness of the map merging result, it allows the discarding of the obviously incorrect transformations.

A very simple similarity metric is proposed by Birk and Carpin in [3]. At first the occupancy grids are simplified by changing the occupancy values of their cells to -1 (free), +1 (occupied), or 0 (unknown). The cell value becomes +1 if it is positive, -1 if it is negative, and remains 0 otherwise. Then the count of the cells with similar values *agr* and with different values *dis* for the current transformation are acquired. The *agr* and *dis* similarity values for a particular cell *agr(x,y)* and *dis(x,y)* are acquired by (1) and (2) [3].

$$agr(x,y) = \begin{cases} +1, \text{if } \mathrm{m1(x,y) = m2(x,y)} \\ 0, \text{if } \mathrm{m1(x,y) \neq m2(x,y)} \end{cases} \quad (1)$$

$$dis(x,y) = \begin{cases} +1, \text{if } \mathrm{m1(x,y) \neq m2(x,y)} \\ 0, \text{if } \mathrm{m1(x,y) = m2(x,y)} \end{cases} \quad (2)$$

The similarity of the maps is acquired by (3) [3].Only cells with values -1 and +1 are taken into account because only these cells contain information about the environment.

$$res = \frac{agr}{agr + dis} \quad (3)$$

The value of the result *res* is a range [0; +1.0] and shows percentage-wise how many cells of the both maps in the overlap are equal. If *res* is low then there are a lot of differences in the overlapping regions of the maps and the map merging procedure is a failure. If *res* is high then the result of the map merging is probably successful [3].

This similarity metric is simple and it works well when maps are sufficiently precise and similar. In reality the sensors of the robots are not perfect and the differences in the maps are unavoidable [7]. In the occupancy grid map each cell contains probability of the corresponding area of the environment being navigable or blocked by obstacle. The rounding of these probabilities causes the loss of some information. For example, the difference between -0.2 and +0.2 is not as significant as the difference between -1 and +1. In this case the cells -0.2 and +0.2 could actually represent the same area of the environment and the difference could be caused by the imperfect sensors of the robots.

This problem can be addressed by introducing a similarity metric that uses cell probabilities. In this case the values of *agr* and *dis* are computed by using (4) and (5).

$$agr(x,y) = \begin{cases} +1, \text{if } |\mathrm{m1(x,y) - m2(x,y)}| < 1 \\ 0, \text{if } |\mathrm{m1(x,y) - m2(x,y)}| \geq 1 \end{cases} \quad (4)$$

$$dis(x,y) = \begin{cases} +1, \text{if } |\mathrm{m1(x,y) - m2(x,y)}| \geq 1 \\ 0, \text{if } |\mathrm{m1(x,y) - m2(x,y)}| < 1 \end{cases} \quad (5)$$

Another way to compute the similarity of the maps is to compare the sets consisting of multiple cells. Such similarity metric resembles previously described metrics but it is more resistant to the local inaccuracies of the maps. The average value is computed for a cell set of each map. Then the comparison is implemented by using these average values.

V. PREPROCESSING OF THE MAPS

An important requirement for the map merging during the exploration is to acquire the result as soon as possible [16]. The faster the maps are merged, the greater is the benefit of using a robot team in the environment exploration. Therefore the time devoted to map merging should be minimized. The preprocessing of the maps can potentially reduce the time required for the map merging. This is especially true when the

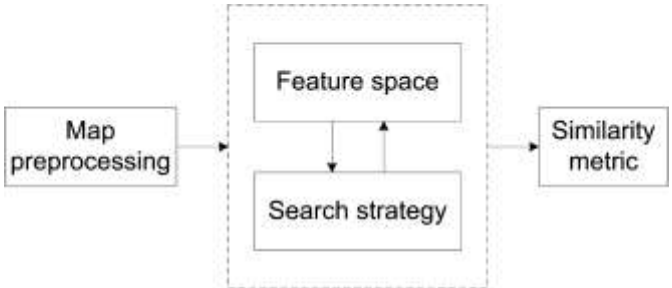

Fig. 2. The process of the map merging. It starts with the map preprocessing step. Then the transformation is found with the feature space and search strategy steps. Then the result is verified with similarity metric.

map merging algorithm is iterative, i.e., it processes the same occupancy grid arrays repeatedly in each step.

It is possible to preprocess robot maps by using techniques from the field of image processing. Usually image preprocessing is applied to decrease the time required for further procession or to improve the quality of the result. In this chapter three approaches – the reduction of the map size, edge detection and map alignment – are considered. Fig. 2 depicts the process of map merging together with the map preprocessing step.

It must be noted that techniques described further are not the only ones that can be used for map preprocessing. There are also other ways, e.g., line simplification introduced in [15]. However, it is impossible to look closely at all the possible techniques in one article.

A. The reduction of the map size

The reduction of the map size is one possible way to reduce total time that is necessary for the map merging procedure. There is a significant difference whether 1000*1000=1000000 cells or 500*500=250000 cells have to be processed. In the latter case there are four times less cells than in the first case. Some map merging methods (e.g., [3]) process the maps multiple times. Therefore the time of the map merging can be greatly reduced.

The reduction of the map size is similar to the problem of image down-scaling in image processing. The map size can be reduced by using any down-scaling algorithm. A very simple example of such algorithm that reduces the map size four times can be seen in the pseudo code further. In this case *m* is the map to be down-scaled, *mX* and *mY* represent the size of the map and *res* is the map acquired after the reduction.

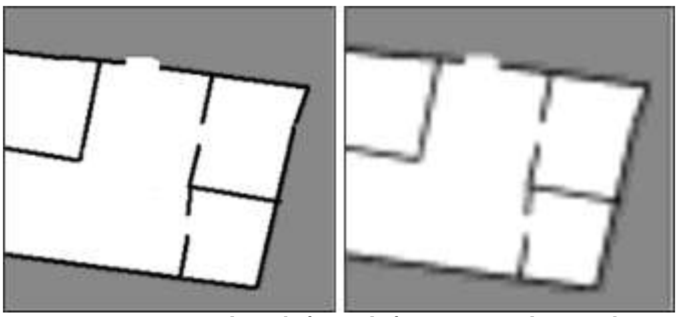

Fig. 3. An occupancy grid map before and after map size reduction. The map size is reduced 16 times. It can be seen that the resolution of the map becomes much lower but the features of the map can still be identified.

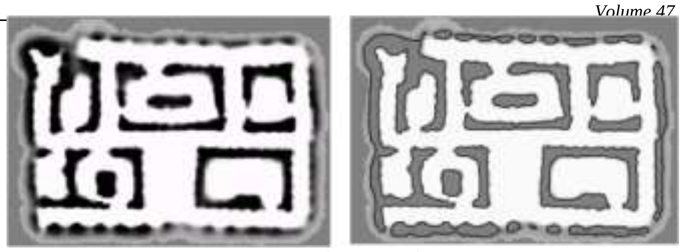

Fig. 4. An occupancy grid map before and after edge detection. During the edge detection the number of occupied cells is reduced. The border cells remain and serve for the purposes of the map merging.

```
For (i = 0, i < mX / 2, i++)   // every two rows
  For (j = 0, j < mY / 2, j++) // every two columns
  sum = m[2*i,2*j] + m[2*i+1,2*j] + m[2*i,2*j+1] +
m[2*i+1,2*j+1]     // the sum of the 4 cell values
  res[i,j] = sum / 4 //result is the average value
  End
End
```

If the size of any map side does not divide by two, then some cells are not taken into account in the processing of the result. However, in most cases this loss is insignificant because most often the values of the cells that are close to the map border are equal to 0, i.e., unknown.

The reduction of the map size itself is not considered as a map merging approach but it is compatible with any map merging approach that uses occupancy grid maps as an input.

It is easy to see how the reduction of the map size can reduce the time necessary for the map merging. Yet very often the purpose of image preprocessing is the improvement of the resulting image quality. It is possible that by reducing the map size too many times, the quality of the result will be reduced, too. It is especially true if the algorithm is as simple as the one just described. Although the algorithm is fast, after only a few iterations (reductions) a significant smudging of the maps will be observed.

B. The edge detection

Another way to reduce the map merging time is to use edge detection before map merging. The edge detection detects the boundaries between objects [18]. The edge detection process serves to simplify the analysis of images by reducing the amount of data to be processed, while preserving useful structural information about object boundaries [19]. If the method uses only cells with ‘occupied’ value for the merging purposes, then the edge detection can greatly reduce the number of the cells to be processed. Fig. 4. depicts a map before and after edge detection.

In image processing the edge detection is not a trivial task. In images the edges usually correspond to the variations of illumination, orientation and depth of the scene [20]. These variations manifest as changes in the intensity [20]. In the map merging case the edges represent the border between occupied and navigable part of the environment. The occupancy grid is purely 2-dimensional. Therefore no illumination, orientation or depth is present there.

The border between the occupied and free cells in the occupancy grid can be determined relatively easily. If every

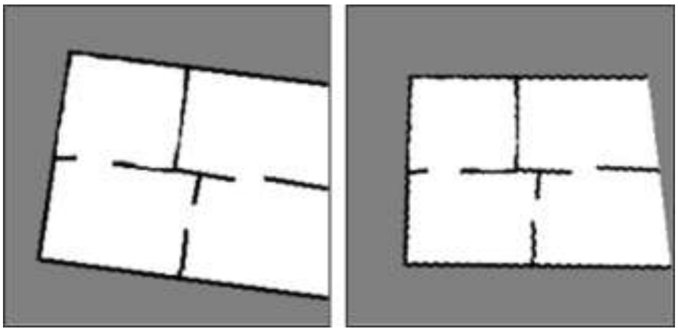

Fig. 5. An occupancy grid map before and after map alignment. The map is rotated by using information gained by computing Hough spectrum.

cell represents the occupation probability, then the borders of the occupied part of the map can be determined with an algorithm depicted below.

```
For (each_cell_c in_the_map)
  If (c.cellvalue > 0) // cell is probably occupied
  Then
    // If all adjacent cells are occupied
    If (cellvalue of all adjacent cells > 0)
    Then
      c.cellvalue = 0; // cellvalue becomes unknown
    End
  End
End
```

In this case only those occupied cells that border free or unknown cells remain marked as occupied. The rest are marked as unknown and are not regarded in the map merging process. All important objects are still represented in the map, because all the border cells have remained in the map. The edge detection result with this algorithm is depicted in Fig. 4.

C. The map alignment

Some map merging approaches work better if specific map preprocessing is executed before the map merging. One such approach is map merging by using Hough specters that is described in [16].

The X-spectra and Y-spectra of the maps are used for the computation of translation part of the transformation. This approach may be useless if the map does not provide distinctive projections along the *x* and *y* axis, i.e., the spectra are mostly flat [16]. For Carpin's approach to work adequately the alignment of one map against the *x* axis is required.

Alignment against the *x* axis is essentially the rotation of the map so that the most of the straight lines in the map would be parallel to the *x* axis (Fig. 5). The rotation can be computed by using the Hough spectrum as described in [16].

VI. The Design of a Map Merging Framework

The analysis of map merging in the context of the image processing gives an insight into the structure of map merging methods. Practically all metric map merging approaches are based on the same principles – they extract a feature space, use it for searching and determine the success of the merging [3], [10], [11], [12], [15], [16]. Some of the methods also

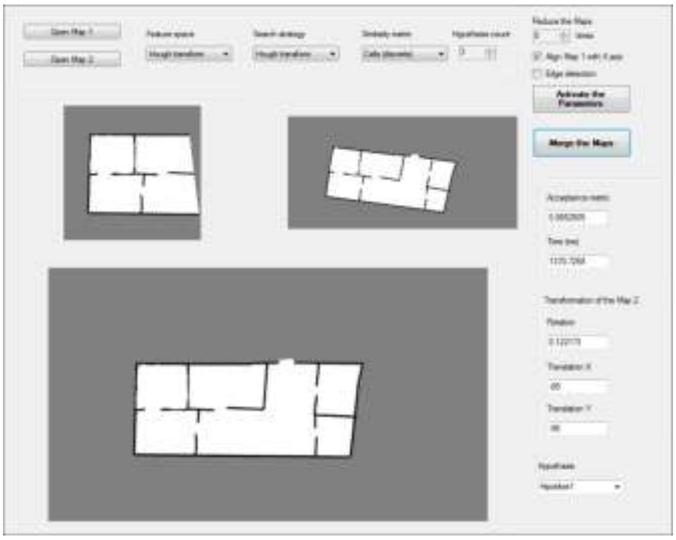

Fig. 6. The screenshot of the map merging framework prototype

preprocess the maps by using some image processing algorithms [15].

It is possible to develop new map merging approaches based on this knowledge. A map merging framework can be designed to help in fulfilling this aim. The framework should be able to enable the implementation and combination of separate map merging components (see Fig. 1). Such framework offers many possibilities in researching and testing various map merging approaches and their components. The components of currently existing map merging approaches as well as completely new components can be combined and their performance tested.

To implement the ideas described above, a map merging framework prototype has been designed and developed (see Fig. 6). This framework aids in developing new map merging methods by giving an opportunity to implement separate map merging components and to combine them to check the performance of the particular combination. These components can also be combined with image preprocessing algorithms to test the impact of these algorithms on different map merging approaches.

Currently the map merging framework contains the following components and algorithms:

- Two feature space detection algorithms – Hough specter detection [16] and Image similarity [3].
- Two search strategies – Hough transformation [16] and Carpin random walk [3].
- Three similarity metrics described in Chapter IV.C.
- A map size reduction algorithm that is described in Chapter V.A. It is currently possible to reduce the size of the map 4 and 16 times.
- An edge detection algorithm described in Chapter V.B.
- A map alignment algorithm described in Chapter V.C.
- The map merging procedure that performs the merging of the maps by using a transformation acquired by the chosen combination of feature space and search strategy.

In Figure 6 an example of map merging can be seen. First, it is necessary to choose two maps to be merged. After that it is possible to choose the feature space, search strategy and similarity metric from the list of approaches implemented in the framework. Only compatible approaches can be combined – the input of the search strategy and the output of the feature space must be similar. Any similarity metric can be used as it is basically a result verification applied after the map merging has taken place.

Additionally to the basic map merging components the map preprocessing techniques can be selected – the size of maps can be reduced selected number of times, map 1 can be aligned against the *x* axis (alignment can be expanded to both maps if necessary) and the edge detection can be performed on the maps.

In the example in Figure 6 Hough specter detection is selected as a feature space, Hough transformation is selected as a search strategy and discrete cell count is selected as a similarity metric. Additionally the map 1 is aligned against the *x* axis.

It is planned to add other map merging components to the framework and start the testing of different combinations in the near future.

VII. Conclusions

In this paper the problem of map merging in the context of image processing was analyzed. Map merging and image processing have many common characteristics, therefore it is reasonable to use the image processing advances in the map merging area.

The image processing subfield closest to map merging is image registration. Image registration is an image processing task that combines the final information from various data sources. Although not exactly the same, this problem is somehow similar to map merging.

As the map merging and image registration methods have many similarities, it can be assumed that metric map merging methods similarly to image registration methods consist of three components: feature space, search strategy and similarity metric. All these components were analyzed in the context of map merging and relevant examples of existing map merging approaches were presented.

Apart from image registration other image processing approaches can be applied to map merging. Maps can be preprocessed before the merging by using image processing methods to reduce the time required for the merging or to improve the quality of the result. Three map preprocessing possibilities were considered in this paper – map size reduction, edge detection and map alignment.

The research of map merging in the context of image processing has helped to determine the requirements of a map merging framework. The framework should allow to easily implement separate map merging components and map preprocessing algorithms and to test them by combining in any permissible combination. Such framework can be of a great assistance in the development of new map merging approaches.

Based on these requirements a map merging framework prototype was designed and developed. At the moment two feature space detection algorithms and search strategies, three similarity metrics, map size reduction, edge detection, map alignment and map merging algorithms have been implemented.

The work can be continued by adding new map merging components to the framework. New ways to preprocess the maps can be added or existing preprocessing algorithms can be improved. The map merging components can be combined and their performance tested together with different map preprocessing algorithms. By gradually updating the map merging framework new and effective map merging methods can be developed. In future the map merging framework may be complemented with data base of maps. By running pre-created tests of map merging, an overall performance of the map merging approach could be acquired automatically.

VIII. Acknowledgement

This work has been supported by the European Social Fund within the project „Support for the implementation of doctoral studies at Riga Technical University”.

References

[1] J. Ko, B. Stewart, D. Fox, K. Konolige, B. Limketkai “A practical decision-theoretic approach to multi-robot mapping and exploration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 3232-3238.
[2] S. Thrun “Robotic Mapping: A Survey,” in *Exploring Artificial Intelligence in the New Millennium*, Morgan Kaufmann, 2002, pp. 1-35.
[3] A. Birk and S. Carpin “Merging Occupancy Grid Maps From Multiple Robots,” *Proceedings of the IEEE*, IEEE, 94. vol., 2006, pp. 1384-1397.
[4] R. Siegwart and I. R. Nourbakhsh *Introduction to Autonomous Mobile Robots*. Cambridge: A Bradford Book, The MIT Press, 2004
[5] S. Thrun, W. Burgard, D. Fox *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, MIT Press, 2006
[6] L. E. Parker “Current State of the Art in Distributed Autonomous Mobile Robotics,” in *5th International Symposium on Distributed Autonomous Robotic Systems*, Knoxville: Springer, 2000, pp. 3-12.
[7] I. Andersone “Space exploration with multiple agents,” M.S. thesis, Riga Technical university, Riga, Latvia, 2009
[8] K. Konolige, D. Fox, B. Limketkai, J. Ko, B. Stewart “Map merging for distributed robot navigation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nevada, 2003, pp. 212-217.
[9] W. Huang and K. Beevers “Topological Map Merging,” in *The International Journal on Robotics Research*, 24. vol., No. 8, Sage science press, 2005, pp. 601-613.
[10] F. Amigoni, S. Gasparini, M. Gini “Merging Partial Maps Without Using Odometry,” in *Multi-Robot Systems. From Swarms to Intelligent Automata*, III vol., Netherlands: Springer, 2005, pp. 133-144.
[11] K. Ho and P. Newman “Multiple Map Intersection Detection using Visual Appearance,” in *3rd International Conference on Computational Intelligence*, Singapore, 2005
[12] J. Bernd *Digital Image Processing. 5th Edition*, Springer-Verlag, 2002
[13] B. Zitova and J. Flusser “Image registration methods: a survey,”in *Image and vision computing*, vol. 21, 2003, pp. 977-1000.
[14] L.G. Brown “A survey of image registration techniques,” in *ACM Computing Surveys*, 24, 1992, pp. 326-376.
[15] R. Lakaemper, L. J. Latecki, D. Wolter “Incremental multi-robot mapping,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, 2005, pp. 3846-3851.
[16] S. Carpin “Fast and accurate map merging for multi-robot systems,” in *Autonomous Robots*, 25. vol., USA: Kluwer Academic Publishers, 2008, pp. 305-316.

[17] S. Carpin and A. Birk "Stochastic Map Merging in Rescue Environments," in *Lecture Notes in Computer Science*, 3276. vol., Berlin/Heidelberg: Springer, 2005, pp. 483-490.
[18] D.H. Ballard and C.M. Brown *Computer Vision*, New Jersey: Prentice Hall, 1982
[19] J. Canny "A Computational approach to edge detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 1986, pp. 679-698.
[20] D. Ziou and S. Tabbone "Edge detection techniques: an overview," in *International Journal of Pattern Recognition and Image Analysis*, 8. Vol., 1998, pp. 537-559.

Ilze Andersone received a B.Sc. degree in computer control and computer science in 2007. In 2009 she received a M.Sc. degree in computer systems. Both degrees were acquired at Riga Technical University, Riga, Latvia. She is now a doctoral student at Riga Technical University.
In 2007-2009 she worked as a Laboratory Assistant at the Department of Systems Theory and Design, Riga Technical University. Since 2009 she has been working as a Researcher at the Institute of Applied Computer Systems, Riga Technical University, Riga. Her research interests include robotic mapping, robot teams, and image processing.

Ilze Andersone. Karšu apvienošana attēlu apstrādes kontekstā
Karšu apvienošanas problēmsfēra ir cieši saistīta ar attēlu apstrādes jomu. Visbiežāk robotu sastādītas metriskas kartes tiek attēlotas kā aizņemtības režģi. Šādam karšu atspoguļojumam ir viegli pielāgot attēlu apstrādē izmantotos algoritmus. Attēlu apstrādes joma, kas ir vistuvākā karšu apvienošanai, ir attēlu reģistrēšana. Var pieņemt, ka metrisku karšu apvienošanas metodes, tāpat kā attēlu reģistrēšanas metodes, sastāv no trīs komponentēm: iezīmju telpas, pārmeklēšanas stratēģijas un līdzības metrikas. Attēlu apstrādes algoritmi karšu apvienošanā var tikt izmantoti arī karšu iepriekšējās apstrādes fāzē. Raksta mērķis ir aplūkot karšu apvienošanas un attēlu apstrādes kopīgās iezīmes un noteikt, kā pētījumu rezultāti var tikt izmantoti karšu apvienošanas ietvara un attiecīgi arī jaunu karšu apvienošanas pieeju izstrādē. Karšu apvienošanas pētījumi attēlu apstrādes kontekstā ir ļāvuši noteikt prasības karšu apvienošanas ietvaram. Ietvaram ir jāļauj viegli realizēt atsevišķas karšu apvienošanas komponentes un karšu iepriekšējās apstrādes algoritmus, kā arī tos savstarpēji kombinēt un pārbaudīt. Šāds ietvars var būtiski atvieglot jaunu karšu apvienošanas pieeju izstrādi. Balstoties uz šīm prasībām, tika projektēts un izstrādāts karšu apvienošanas ietvara prototips. Šobrīd tajā ir realizēti divi iezīmju telpas noteikšanas algoritmi un pārmeklēšanas stratēģijas, trīs līdzības metrikas, karšu izmēra samazināšana, robežu noteikšana, karšu savietošana ar x asi un karšu savietošanas algoritms. Darbu var turpināt, papildinot ietvaru ar jaunām karšu apvienošanas komponentēm. Var tikt ieviesti jauni karšu iepriekšējās apstrādes veidi vai arī uzlaboti esošie algoritmi. Karšu apvienošanas komponentes var kombinēt un to veiktspēja pārbaudīta kopā ar dažādiem karšu iepriekšējās apstrādes algoritmiem. Pakāpeniski papildinot karšu apvienošanas ietvaru, var tikt izstrādātas jaunas un efektīvas karšu apvienošanas metodes.

Илзе Андерсоне. Объединение карт в контексте обработки изображений
Область объединения карт тесно связана с областью обработки изображений. Обычно метрические карты, созданные для роботов, представлены как сетка занятости. Алгоритмы, используемые в обработке изображений, просто применить к такого рода представлению карт. Область обработки изображений, которая находится ближе всего к объединению карт, является регистрацией изображений. Можно предположить, что методы объединения метрических карт аналогичны методам регистрации изображений и состоят из трех компонентов: пространства признаков, стратегии поиска и метрики сходства. Алгоритмы обработки изображений также могут быть использованы в объединении карт для предварительной их обработки. Целью данной работы является изучение сходства между областями объединения карт и обработки изображений и определение, как результаты этого исследования могут быть использованы для развития системы объединения карт и, следовательно, новых подходов в области объединения карт. Исследования объединения карт в контексте обработки изображений помогли определить требования к системе объединения карт. Система должна позволять легко использовать отдельные компоненты объединения карт и алгоритмы предварительной обработки карт и проверить их в любой допустимой комбинации. Такая система может быть большим подспорьем в развитии новых подходов в области объединения карт. С учетом этих требований был разработан и создан прототип системы объединения карт. На данный момент реализованы два алгоритма обнаружения пространства признаков и стратегий поиска, три метрики сходства, уменьшение размера карт, выделение границ, выравнивание карт и алгоритм слияния карт. Работа может быть продолжена путем добавления новых компонентов к системе объединения карт. Могут быть введены новые способы предварительной обработки карт или улучшены существующие алгоритмы. Компоненты объединения карт могут комбинироваться и их производительность испытана вместе с алгоритмами предварительной обработки карт. Постепенно могут быть разработаны новые и эффективные методы объединения карт.