International Journal Volume 10, Number 2, December 2021
Information Technology Applications (ITA)

ALGORITMS FOR ENVIRONMENT MAPS OF MOBILE ROBOT

Oto Haffner

Abstract:

Robot needs to have some representation of the environment to free of collision move. There are different types of environment maps in mobile robotics however the most used are metric maps. The aim of this work is the proposal of algorithms, for real robotic system, that can map the environment. In this work we deal with the analysis of properties of the laser scanner and the making of its model. Further a method is proposed for making map based on the properties of the scanner and program implementation of this method. The result of this work is a program that maps the environment and which is usable on real robot system.

Keywords:

Map, laser, rangefinder, environment, mapping.

ACM Computing Classification System:

Information systems, Information systems applications.

Introduction

Robot is complicated mechanical device, controlled by computer, equipped by many sensors to get information from surrounding environment and by many actuators for manipulating with objects or self manipulating. This definition is one of the many.

Robots can be sorted to more groups by their construction. Independent group are autonomous mobile robots. There must be defined word autonomous and mobile. The mobile robot is able to move and orient in unknown environment. To do this actions robot needs to know answers to three main questions: “Where am I?”, “Where I want to go?” and “How to get there?”. To get answers to these questions, robot needs somehow to sense the surrounding space.

Sensors for localization and navigation can help us to answer the previous three questions. One of the sensors for navigation is laser rangefinder with which we work in this paper and also analyze its characteristics.

To do the tasks of mobile robotics, mainly localization and navigation, it is necessary to have a representation of environment - configuration space. The main configuration spaces are geometric, topologic and metric maps. In this work we will deal with methodology of making the metric map using laser rangefinder.

1 Sensor for Navigation

One of the main activities of mobile robot is obtaining and processing of information about environment in which robot is moving. This information robot obtains thanks to different sensors.

The most used are rangefinders worked on different physical principles – ultrasonic, infrared and laser. These types of rangefinders can measure distance to measured object and also direction to this object.

1.1 Parameters of laser rangefinder HOKUYO UTM-30LX

The parameters stated by the manufacturer are defined for a surface of exact size, situated perpendicularly to the beam of measurement. In practice it is often necessary to scan various kinds of surfaces from different angles, therefore it is necessary to experimentally verify the stated parameters for different surfaces and different angles of measurement. To complete the data necessary for the creation of the sensor model three types of experiments were performed. The first was focused on the repeatability of the measurement perpendicularly to the object, the second was aimed at the repeatability of the measurement under the angle of 45° and the third was aimed to the stability of the measurement of the edge of an object. The first two tests verify the stability of the measured distance, while the third test verifies the stability of the measured angle and the measurement on the border of two objects. (Dekan et al, 2011) The surfaces of white wall, white paper, black paper, white T-shirt, aluminium foil, mirror and plexiglas were used for these tests.

1.2 Result of measurement

For each material were taken 200 measurements in predetermined distance (500mm and 1500mm) and two angles (90° and 45°). For mirror and plexiglass surface were taken additive measurements because of unclear results. From all measurement results was found some knowledge. Laser rangefinder would give bad measurements results for the material of plexiglass or glass and mirror or high polished metal surface. These facts can be compensated by the synthesis of data from the more sensors worked on different physical principle. We can also guess that the rangefinder would bad interpret materials or obstacles which are very thin or narrow e.g. wire mesh.

Because the laser rangefinder is planar, the measurement must be interpreted in plane. This is the reason why there is needed a measurement aimed at consistency of data around the edges of obstacles. That measurement was done on Institute of Control and Industrial Informatics FEI STU (Dekan et al, 2011). The measurement was taken by four distances 50 cm, 150 cm, 250 cm and 530 cm. For each distance, measurements were performed 10 times with 500 measured distances per data set. The measurement result is the number of shifts of the object for its left and right edge (Tab.1).

Table 1.

Measured distance	Number of measurement									
	1	2	3	4	5	6	7	8	9	10
50 cm	161	246	197	220	180	190	146	108	136	248
150 cm	191	27	105	211	85	90	60	31	180	33
250 cm	44	43	48	38	160	74	39	5	142	18
530 cm	5	32	34	0	42	74	14	2	1	0

Based on the Gaussian distribution and the measured distances it is possible to determine the shift of the object border on the basis of angular distances between two beams of the laser range finder (Tab.2) according to:

$$y = 3.\sigma.\sin(d).dst \quad (1)$$

where d is the angle between two consecutive laser beams and dst is the measured distance.

Table 2.

Measured distance	σ [mm]	y [mm]
50 cm	0.6078	3.9783
150 cm	0.416	8.1677
250 cm	0.2907	9.5131
530 cm	0.1269	8.802

1.3 Model of laser rangefinder

The complete sensor model consists of two partial models. Hokuyo UTM-30LX is a planar rangefinder so it measures distance in the plane. It must be interpreted in the plane.

Based on the data obtained from the measurement (Fig.2), the Gaussian sensor model was chosen. The probability distribution of measured distances in the Gaussian model equal to:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2)$$

where σ is the standard deviation and μ is the average distance of measurements. For partial model in the direction of measuring distance was Gauss distribution determined as average standard deviation from all type of measurement. The value is σ = 4,9391. The resulting Gaussian distribution is in (Fig.2).

As the model is in the direction perpendicular to the direction of distance measurements, the model derived from the shift of object boundaries was chosen. The resulting standard deviation for the Gaussian distribution in this direction was σ = 3mm. The resulting distribution can be seen in (Fig.3) (Dekan et al, 2011).

Based on these two resultant Gaussian distributions the resulting model of laser range finder has been created (Fig.4):

$$f(x,y) = \frac{1}{2\pi\sqrt{\sigma_x^2\sigma_y^2}} e^{-\left(\frac{(x-\mu_x)^2}{2\sigma_x^2}+\frac{(y-\mu_y)^2}{2\sigma_y^2}\right)} \quad (3)$$

where x is the corresponding measured distance, the σ_x is standard deviation of the Gaussian distribution in the direction of the measured distance, y is the corresponding dispersion distance in a direction perpendicular to the measured distance and σ_y is the standard deviation of the Gaussian distribution in the direction perpendicular to the measured distance.

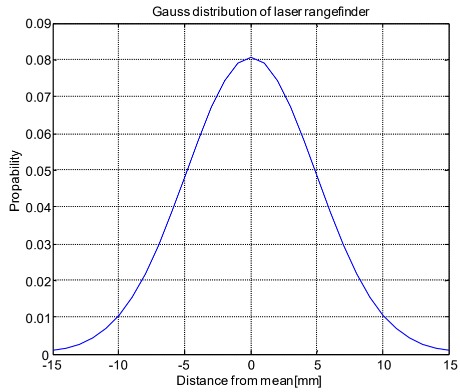

Fig.1. Gauss distribution of laser rangefinder for x direction.

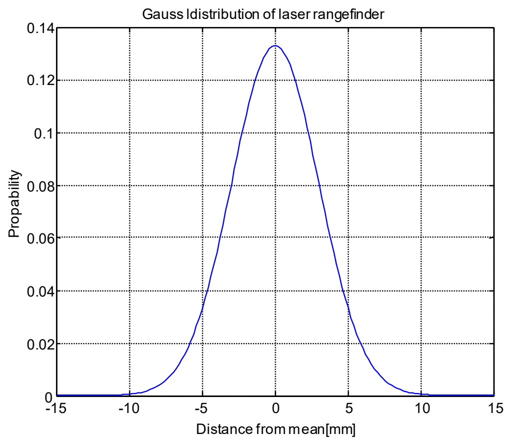

Fig.2. Gauss distribution of laser rangefinder for y direction.

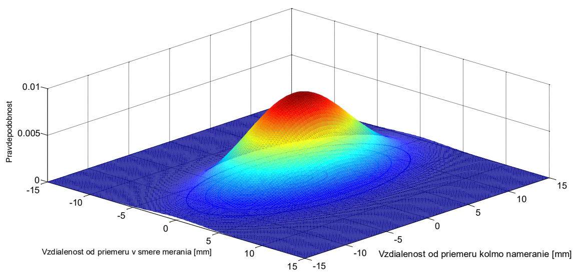

Fig.3. Final model of laser rangefinder. Left coordinate: distance from mean in x direction, Right coordinate: distance from mean in y direction. Elevation: probability.

For purposes of mobile robotic can be model of laser rangefinder be given in form of table. For the selected raster 10x10mm is table (Tab.3), where the number in the cell represents probability of occupancy.

Table 3.

0.0074	0.1397	0.0074
0.0329	0.6228	0.0329
0.0074	0.1397	0.0074

2 Map Making Method

For the best representation of environment using laser rangefinder was chosen a metric map. This map uses grid with cells. In our case, the cells are represented by pixels in map. It is easy to determine the occupancy of each pixel.

At first, it was necessary to design a procedure thanks to data from laser which can be potted to map. The principle we describe on (Fig.5).

The method of making local map is needed. Let us based on the assumption that the initial position of robot is **[x0, y0]**. There is known distance **r** and angle **α** for each measured point. We must work with fact that the angel of first measured point is set to 45°. Each next measurement has the angel smaller for 0.25°(angle step) respectively for angel 0.25 * index_measurement. With this fact is worked in the calculation of the resulting angle. As well we must work with orientation of the coordinate system. In this case we do not work with rotation of robot yet. On the base of mentioned facts, we can define the coordinates of measured points as:

$$x = x_0 + r * \cos(45° - 0{,}25 * index\_measurement)$$
$$y = y_0 + r * \sin(45° - 0{,}25 * index\_measurement) \qquad (4)$$

where *x, y* is coordinates of measured point, x_0,y_0 is initial robot position, *r* is measured distance and *index_measurement* is analogical index of measured point (1-1080).

Robot changes its position and rotation during mapping. We must work with this fact in correction of equations. Also the initial coordinates and robot rotation must be set. Initial rotation angle can be set as 0°. Counter clockwise robot rotating adds to overall rotation angle, anticlockwise robot rotation subtract from overall rotation angle. The problem is how to set the initial robot coordinates in map. Wrong set can cause problems during plotting. With this problem we will deal later. Let us suppose that the robot coordinates can have also negative value but this is not possible in plotting to map.

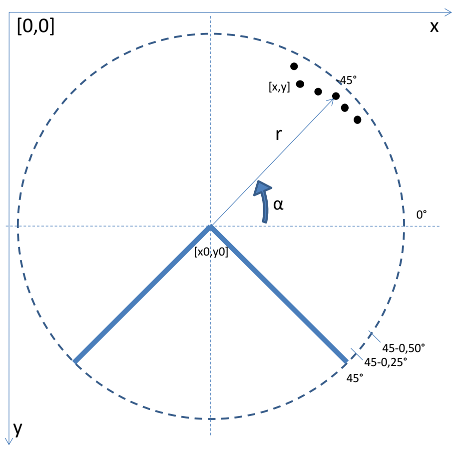

Fig.4. Method sketch.

The coordinates of point for making global map are defined as:

$$x=(x_0+x_r)+r*cos(45°-0,25*index\_measur-\varphi_r)$$
$$y=(y_0-y_r)+r*sin(45°-0,25*index\_measur-\varphi_r) \quad (5)$$

where x,y are coordinates of measured point in the image matrix, x_0,y_0 is initial robot position in coordinate system of image matrix, r is measured distance, *index_measur* is index of measured point, x_r,y_r are robot coordinates in robot coordinate system, φ_r is overall rotation of robot considering the axe y_r.

Philosophy of the relationship of rotation angle and coordinate system of image matrix is for better understanding shown in (Fig.6).

The main idea of making global map is plotting local maps in each coordinates where each map was measured. The main task is to set correct position (coordinates) and correct rotation angle in coordinate system of image matrix.

The main idea of making global map is plotting local maps in each coordinates where each map was measured. The main task is to set correct position (coordinates) and correct rotation angle in coordinate system of image matrix.

The main problem is alignment local robot coordinate system in global coordinate system of environment map (image matrix). However this work does not deal with problems of robot localization.

3 Program Realisation

After making of sensor model (section 1) and method of map making (section 2) we can do a program realisation. Because program codes for laser rangefinder UTM-30LX are for operating system Linux and also the mobile robotic system in FEI STU work under this operating system, we decides to use operating system Linux-Ubuntu 12.04 . For working with images was used library OpenCV 2.4 which is a library of programming functions mainly aimed at real-time computer vision (Bradski and Aguado, 2008). The resulting program is a console application.

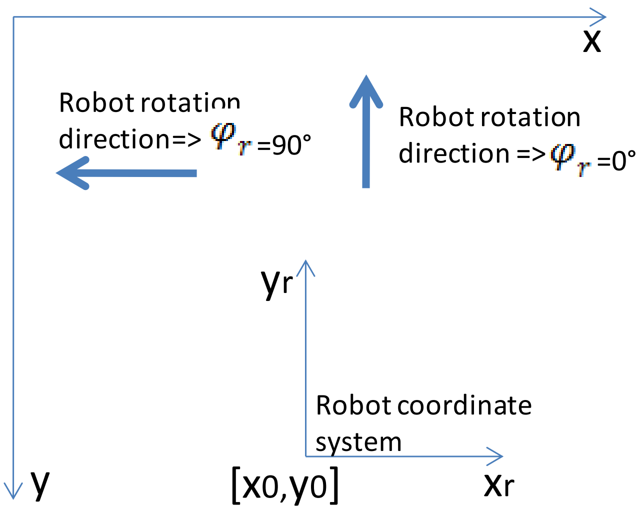

Fig.5. Coordinate system of picture matrix.

3.1 Local map

For making local map we need this information: measured data, robot position, map scale. Measured data from laser scanner are saved in text file, each measure on its own line. From structure like this is easy to connect one measure with measure angle. Distances are measured in millimeters. It must be set the map scale and robot initial position before plotting the map. Orientation and robot coordinate system is shown on (Fig.5). By the equation (5) (section 2) can be calculated position of measured data in the image (map). For simplicity, in the early stages of making maps we will not use sensor model.

For making of first local map was created artificial environment for paper boxes (Fig.6 - left). The distance of paper box from laser was less than 600 mm so the size of image matrix was set to 1000 x 1000 pixels with scale 1:1 (1 pixel = 1 mm). The position of laser in map was experimentally set as [500, 800]. The resulting map we can see on (Fig.6 - right). In this case the neighbouring measured points were connected with segment.

3.2 Global map

To make a global map we need this information: measured data, actual position and rotation of robot, initial position and rotation in map, scale.

For making a global map was created artificial environment from paper boxes which simulate real robot environment (Fig.8).

In this environment 21 measurements were done. For each measurement was remembered the position of laser and a total rotation angle.

Fig.6. Artificially made environment (left) and its local map (right).

Fig.7. Artificially made environment for global map.

Each measurement is saved in text file. Position and robot angle are saved in one text file. The structure of data is: X position, Y position, Angle. Metric unit of position is centimeter and angle decimal degree. The robot position is not position in map but position in robot coordinate system (section 2). There can be a situation that the map size is too small and some data would not be plotted. That is the reason why there is a scale. The scale has an effect on the robot coordinates. The resulting equations for measured point with scale are:

$$x=(x_0+x_r*10/SCALE)+r/SCALE* *\cos(45°-0{,}25*index\_measur -\varphi_r)$$

$$y=(y_0-y_r*10/SCALE)+r/SCALE* *\sin(45°-0{,}25*index\_measur -\varphi_r) \quad (6)$$

where x,y is position of point in coordinate system of image matrix, x_0,y_0 is initial position of robot in coordinate system of map, r is measured distance, *index_measur* is index of measured point, x_r,y_r is position in coordinate system of robot, φ_r is total robot rotation considering the axe y_r, *SCALE* is our set scale.

3.3 Map plotting 1

First way of making map was connecting the center of laser and plotted point with segment. Resulting map we can see in (Fig.8).

Disadvantage of this way is that the data and borders plotted in the image are redrawn. In (Fig.8) we can see how measured data overlap the image size. This is caused only by missing obstacle. The white color changed to violet is only for illustration. In (Fig.9) is demonstrated partial drawing of the map.

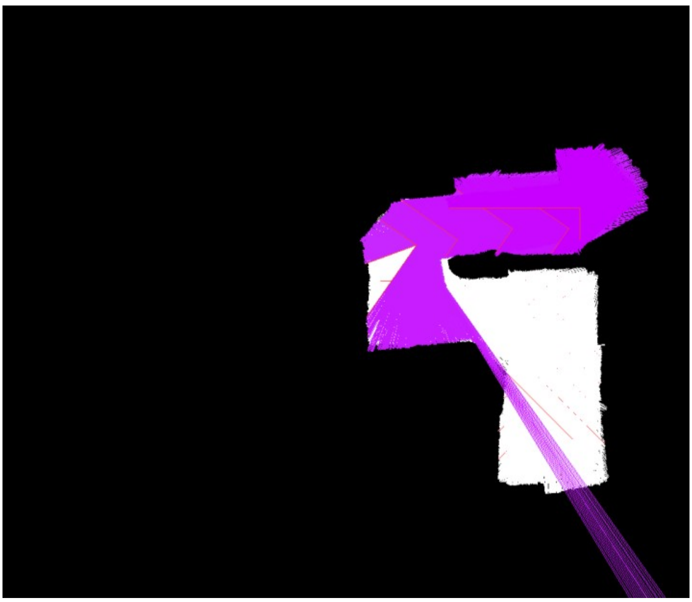

Fig.8. Global map drawn by first way.

Fig.9. Partial drawing of map.

3.4 Map plotting 2

We tried to improve previous way of making map by setting the brightness level of segment between center of laser and measured point to one level- white color and the measured point to another- black color. Resulting map we can see in (Fig.10).

The border of map was not highlighted by another brightness level and that is the reason why they were overlapped. The border is better to see in this map however the data are still redrawn and it cause map distortion. In (Fig.11) we can see partial drawing of map.

3.5 Map plotting 3

In this way we decided to not draw the segment connecting the center of laser and measured point. Only the measured point is drawn. Resulting map we can see in (Fig.12).

In previous way of making map we highlighted the borders of map however some part of border was redrawn due to position error. In this map we can see each point measured by the laser. In the upper part of map we can see the imperfection. This was caused by the wrong determining of laser position. In (Fig.13) is demonstrated partial drawing of map.

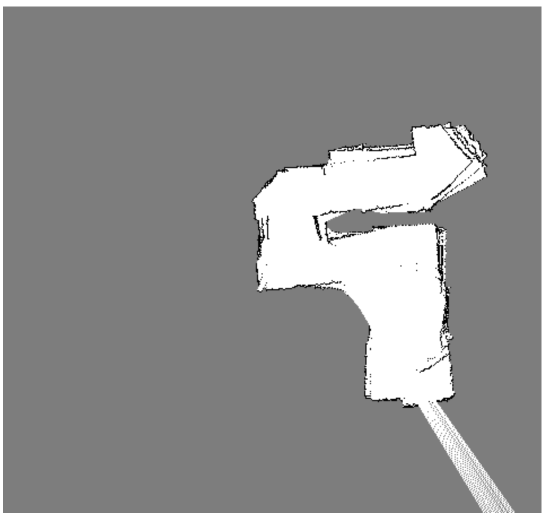

Fig.10. Global map drawn by second way.

Fig.11. Partial drawing of map.

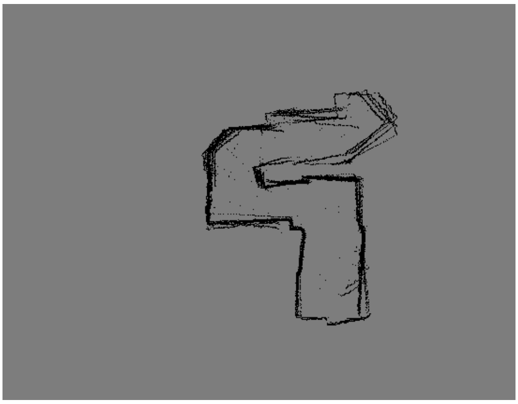

Fig.12. Global map drawn by third way.

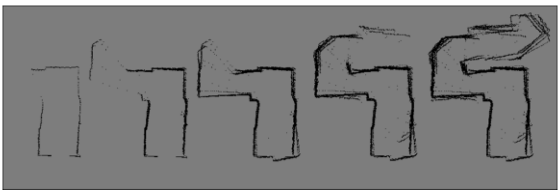

Fig.13. Partial drawing of map.

3.6 Map plotting 4

We tried to improve the previous way of making map. The main idea is that the drawn points can obtain more brightness level. If the measurement falls in the pixel for the first time, the pixel obtains the upper set brightness level. If the measurement falls in the pixel again, the pixel brightness is lover for set level. This is repeated until the pixel has the lover brightness level- black color. Resulting map, where 3 brightness levels are set, we can see in (Fig.14)

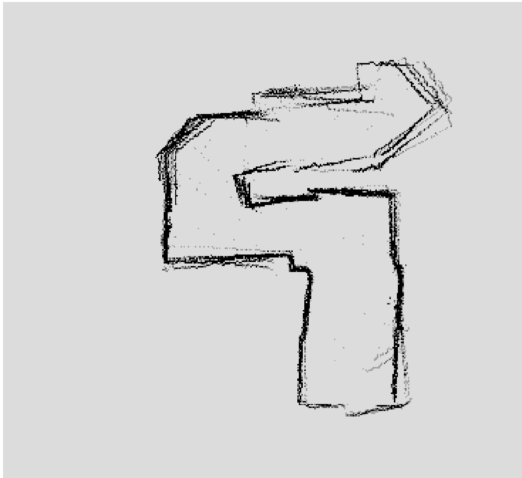

Fig.14. Global map drawn by fourth way.

After thresholding this map we can delete the points with lover brightness level. Less number of measurements fell on these pixels. The less number could cause error of measurement or odometry error but also moving obstacle. The partial drawing of map is in (Fig.15).

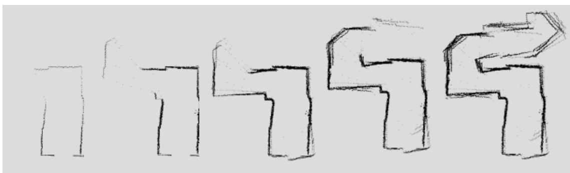

Fig.15. Partial drawing of map.

3.7 Editing resulting map

As the final way of making map was chosen the last (fourth) way. This map needs thresholding (Fig.16) because it contains pixels which have different brightness level.

On the map is applied binary threshold. The pixels with brightness level higher than 160 (set by us), obtain after thresholding maximal brightness (gray color) and all other the minimal (black color). This threshold level was set because if in the point fell 3 measurements (5 possible), we consider this point as occupied. The level of threshold depends on number of brightness level, which can obtain one pixel, but also on the obstacle speed and size. To set the brightness level we did not have enough data, that is the why we only suggest the using of thresholding.

As we can see in (Fig.16), the border of map is not clean and straight. We apply on the map advanced algorithms of image processing. We apply the dilation algorithm which cause “thickening” of border (Fig.17). This algorithm will cause that the distorted map border is filled. In this case the core size is 5 pixels and has circular shape.

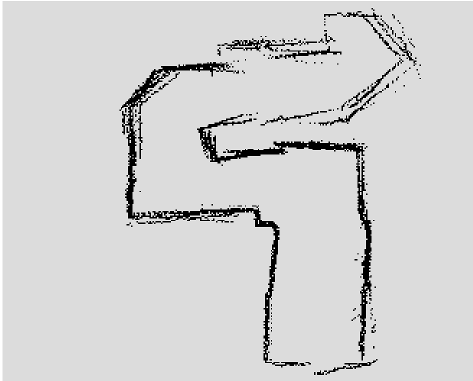

Fig.16. Thresholded global map.

Fig.17. Map after dilation.

After application of dilation the following algorithm was erode. This algorithm cause that the borders will be thinner (Fig.18). In our case the core size was 4 pixels and the rectangular shape. Using bigger core caused deleting the thinnest lines.

For purposes of global navigation or localization would be sufficient map with borders or lines thick 1 pixel. The thickness of borders in previous maps does not determine the thickness of obstacle. This thickness is caused only by measurement error. This is the reason why we applied the skeletonization algorithm (Fig.19). The core of this algorithm is distance transformation.

The result of the algorithm is the map skeleton but it contains some unneeded disturbing segments and points. This is the reason why we tried to improve process of editing map. The background brightness level was set to 255 (white color). In process of erode and dilate was the core set to circular shape and size to 7 pixels. The erode filter was used 2 times a row and after this the filter of dilate was used (Fig.18).

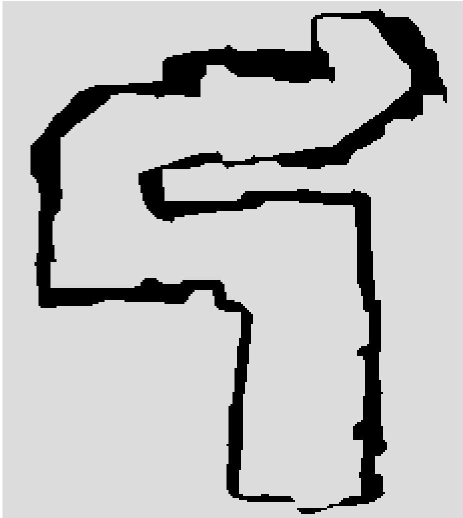

Fig.18. Map after erode.

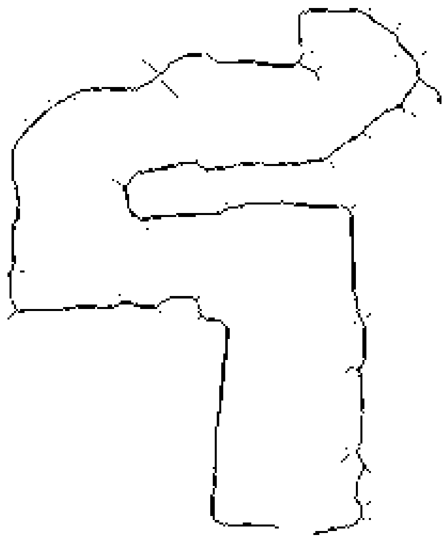

Fig.19. Map after skeleton algorithm.

The result of the algorithm is the map skeleton but it contains some unneeded disturbing segments and points. This is the reason why we tried to improve process of editing map. The background brightness level was set to 255 (white color). In process of erode and dilate was the core set to circular shape and size to 7 pixels. The erode filter was used 2 times a row and after this the filter of dilate was used (Fig.20).

On the filtered map was used Gauss smoothing algorithm and binary thresholding. Gauss filer cause blurring of map borders and their smoothness. The sharp roughness are softened after the thresholding.

On edited map was used Zhang-Sueng thinning algorithm. The resulting figure we can see on (Fig.21). By applying filters and thinning algorithm we get required result. The small disadvantage is that the sharp corners ale rounded now however this does not make problems in tasks of global navigation.

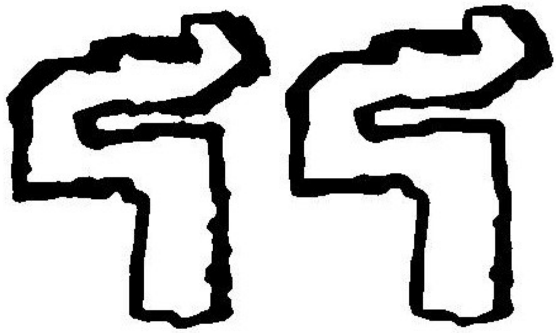

Fig.20. Map after double erode a one dilation (left)
and after gauss smooth fiter and thresholding (right).

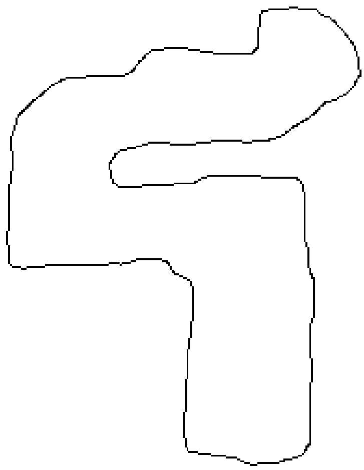

Fig.21. Skeleton of map after using of Zhang-Suen algorithm.

Conclusion

We could met some problems in application of the algorithms of making map.

Map scale

The problem with right setting of scale depends on the environment size that robot is to map. The scale of map could be set manually if the robot operator knows what environment is to be mapped. If the robot is to map building with size of 20 x 20 meters and the map will be 1000 x 1000 pixels the scale could by theoretically set to 1:20. In this case, the initial coordinates of drawing the map have to be set ideally to not exceed the border of image.

Here comes the problem how to set the right initial coordinates. One of the solutions can be oversizing the scale. However in this case the map could be drawn very little in view of size of image matrix. Too high scale can cause higher error in global navigation. Next solution can by adaptive adjustment of scale (Fig.22).

We can determine the position of all point that are to be drawn. If the coordinate fall in to the pixel out of the map matrix, the map will be redrawn in other scale again. Now the pixels will fall in to the new image matrix. This way can be ensured that the map will have the same size and all the flat will be used.

Map using

Primary using of our map will be global navigation thus finding of way between start and finish. Metric global map can be used for making of topological or geometric map. This map will use robot.

However, global map can use also the human. There can be situations where human needs information about size and shape of environment e.g. building where human cannot go by himself. In this case the map can be use for visualization of unknown environment. Global map contains also the metrics so we can compute the content of area.

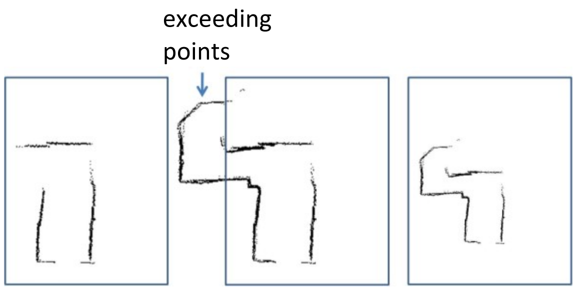

Fig.22. Initial map (left), exceeding points (middle), adapted scale (right).

Acknowledgement

This work has been supported by the Scientific Grant Agency of the Ministry of Education of the Slovak Republic under the grant 1/1241/12.

References

[1] Siegwart, R., Nourbakhsh, I.R., (2004) Introduction to Autonomous Mobile Robots. Massachusetts Institute of Technology,. 336p.

[2] Dekan, M., Duchon, F., Jurišica, L., Vitko, A., (2011) Probabilistic model of laser range-finder, Ad Alta : Journal of Interdisciplinary Research vol.1.

[3] Duchoň, F., (2012). Lokalizácia a navigácia mobilných robotov do vnútorného prostre-dia. Bratislava: STU

[4] Zhang-Suen thinning. [online]. [cit. 2013-05-09]. Available on: http://www-student.cs.uni-bonn.de/~hanisch/sk/zhangsuen.html

[5] Bradski, G., Aguado, A.,(2008). Learning OpenCV. 2nd ed. Sebastopol: O'Reilly, 555 p.

[6] Thrun, S.,(1998) Learning metric-topological maps for indoor mobile robot navigation. Artificial Intelligence. 1998, vol. 99, p. 21-71.

Authors

doc. Ing. Oto Haffner, PhD.
Institute of Automotive Mechatronics,
Slovak University of Technology in Bratislava
Bratislava, Slovak Republic
oto.haffner@stuba.sk
Research activities in image processing, computer vision, pattern recognition, signal/image/video processing, robotics.