

PARAMETERIZED SENSOR MODEL AND AN APPROACH FOR MEASURING GOODNESS OF ROBOTIC MAPS

*Yeasin Arafat [†]

Office of the President of the General Assembly
United Nations, New York, NY 10017
email: arafat@un.org

Thomas Hellström

Dept. of Computer Science
Umea Universitet, Sweden
email: thomash@cs.umu.se

Jayedur Rashid [‡]

MR Lab at AASS
Örebro Universitet, Sweden
email: jrd@aass.oru.se

ABSTRACT

Map building is a classical problem in mobile and autonomous robotics, and sensor models is a way to interpret raw sensory information, especially for building maps. In this paper we propose a parameterized sensor model, and optimize map goodness with respect to these parameters. A new approach, measuring the goodness of maps without a handcrafted map of the actual environment is introduced and evaluated. Three different techniques; statistical analysis, derivative of images, and comparison of binary maps have been used as estimates of map goodness. The results show that the proposed sensor model generates better maps than a standard sensor model. However, the proposed approach of measuring goodness of maps does not improve the results as much as expected.

KEY WORDS

Robotics, Intelligent Robotics, Sensor Model, Robotic Map, Map Goodness.

1 Introduction

Autonomous robots usually use an explicit map of the actual operating environment. There exist several techniques for building such maps [14]. Generally, a robot uses various sensors such as laser-scanners, cameras, and sonars to sense the environment. Sonars are the most commonly used sensors because of price, relatively low energy consumption and ease of use [11, 10]. To interpret the sensor readouts, some kind of sensor model is used.

Many sensor models include parameters, and the quality of generated map depends on these parameters. In this paper, we propose a modification of a generic parameterized sensor model described in [10]. A new approach, *measuring goodness of maps without using the ground map* ("real" map of the environment), is proposed and evaluated. Our map goodness approach is inspired by Martin and Moravec [7]. For all experiments, we have used an Amigobot [12] from ActiveMedia. It has six sonars mounted around its rim for 360 degrees coverage.

The paper is outlined as follows: in Section 2 we discuss related works. Section 3 describes the general concepts and theories of sonar based map building. Next we illustrate our proposed sensor model in Section 4. New ways for measuring goodness of robotic maps, which is used for parameter optimization of the proposed model is discussed in Section 5. At the end, we discuss the results and conclusions of the work in Section 6.

2 Related Works

Generally, there are two approaches for sonar-based map building, related to the data representation [10, 11]: *grid based* and *feature based*. Grid based maps, also known as evidence-grids or occupancy grids, were introduced by Moravec and Elfes [8, 2]. In this approach, sensory information is represented in a two dimensional (2D) or three dimensional (3D) grid, also known as an occupancy grid. *Feature-Based* maps [13] represent the physical world by points and lines. In the present work, grid-based techniques are used.

A sensor model interprets the sensor readings. R. Murphy describes a general sensor model [10], which is used and further developed in this paper. Moravec and Elfes have proposed a probabilistic sensor model for generating high resolution maps [9]. Fichtner and GroBmann have designed a probabilistic sensor model for camera pose estimation in hallways and cluttered space and have used this model for probabilistic robot localization [3].

The sensor model is used by a so called *evidential method* to create and update the map. The three most common evidential methods are Bayesian, Dempster-Shafer and HIMM [10]. In this paper, we have used a Bayesian approach for grid updating.

For several reasons estimates of goodness of maps are important. They may for instance, be used as merit values for automatic learning of sensor models [7], for building Self-Organization Maps [15], etc. Since the Self-Organizing Map (SOM) involves both a vector quantization and a nonlinear projection of the input space to some output space, looking at the average quantization error is a measure of map goodness [15, 6]. Ypma and Duin [15] have proposed a novel variant of "failure detection in process monitoring" measure for map-goodness, which is based on the cost function for minimizing SOM training algorithm.

*Please note that the authors name are given in alphabetical order.

[†]During this work the author was in the Dept. of Internetworking, Royal Institute of Technology- KTH, Stockholm, Sweden.

[‡]The work is done when the author was in Umea University, Sweden.

The use of information theory concepts such as entropy is another approach [7]. In [9], Moravec and Elfes present another approach for scoring maps over grid cell similarities. If an empty cell of one map overlaps the empty cell of another map then positive score is increased otherwise the negative score is increased and similar for occupied cells. This provides a measurement for goodness of maps or for matching maps. However, most of these approaches rely on an existing ground map of the environment, and thus our work is distinct from these approaches.

3 Sonar Based Map building

In this work we have used ultrasonic sound sensors, also called *sonars*. They have a wide angle of view, are inexpensive and easy to control [11, 10]. With this kind of sensors, grid based models are suitable because of their unreliable readings problem [11]. Grid based map methods typically use Bayesian rule of probability for updating the map. It allows a statistical expression of confidence in the correctness of data by projecting the sonar response onto the grids.

We have used Evidence-Grid model [8, 2]. The evidence grid is a two or three dimensional grid (2D is most commonly used and we are using too) corresponding to the actual world. In grid, the world is defined as a discrete spatial lattice in which each cell of the grid is a discrete state variable [4, 1]. This is the the spatial world representation and based on probabilities. Each cell of the grid is given a value as the belief: *occupied*, *empty* or *unknown* by each sensor readout and some evidential methods. The evidential method is the sensor model with some techniques that generate numerical values from the model.

3.1 Sensor Model

A sensor model delivers conditional probabilities $P(s|H)$ on a cell basis. The s stands for the proposition “the sensor delivers the value s ”. The H stands for the proposition “the state of the cell“, where H can be either *Empty* or *Occupied*.

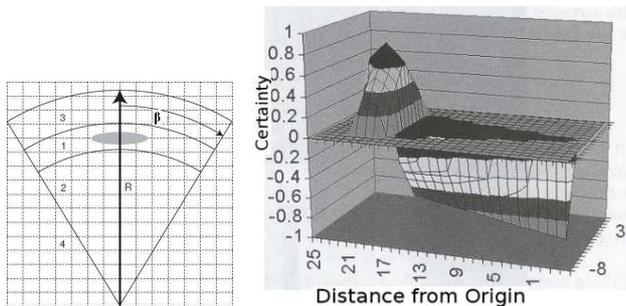


Figure 1. 2D (left) and 3D (right) representation of a sonar beam projected onto an occupancy grid (adapted from R. Murphy [10]).

Most roboticists have converged to models similar to the one illustrated in Figure 1 (adopted from Murphy [10]). In this bird-view figure, the sonar sensor is assumed to be placed at the beginning of the arrow, 2β . R represents the maximum range at which the sensor can detect objects. The model is divided into three regions: **Region 1:** affected cells are probably *occupied*, i.e. $P(s|H)$ is high. **Region 2:** affected cells of this region are probably *empty*, i.e. $P(s|H)$ is low. **Region 3:** the states of the affected cells are *unknown*, i.e. $P(s|H)$ is around 0.5 (or some other chosen constant). These claims are stronger for cells close to the acoustic axis and are also stronger close to the sensor. To assign probabilities corresponding to this intuitive model, we use a set of functions proposed in [10].

Probabilities for each cell in **Region 1** are given by Equations 1 and 2:

$$P(s|Occupied) = \frac{\frac{R-r}{R} + \frac{\beta-\alpha}{\beta}}{2} \times Max_{occupied} \quad (1)$$

$$P(s|Empty) = 1.0 - P(s|Occupied) \quad (2)$$

where r and α are the distance and angle respectively of the grid cell seen from the sensor position, R is the maximum sensing range for the sensor, β is half the width of the sensor sensing cone and $Max_{occupied}$ is a constant < 1 (the value 0.98 has been used in the presented work) introduced to assure that the probability for a cell being occupied is never equal to 1 [10]. The thickness, width or *tolerance* of **Region 1** is denoted by the ϵ .

Probabilities for grid cells in **Region 2** are given by Equations 3 and 4:

$$P(s|Occupied) = 1.0 - P(s|Empty) \quad (3)$$

$$P(s|Empty) = \frac{\frac{R-r}{R} + \frac{\beta-\alpha}{\beta}}{2} \quad (4)$$

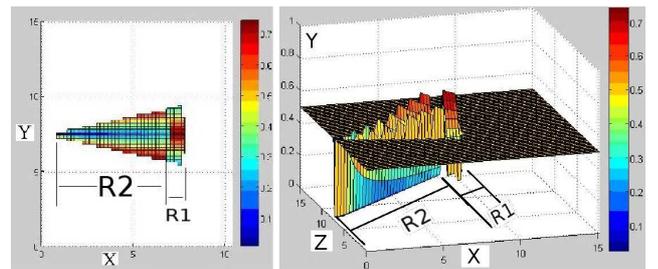


Figure 2. 2D (left) and 3D (right) presentation of a sonar beam using equations 1, 2, 3 and 4. The colored bar represents the probability of belief values.

The sensor model obtained from equations 1, 2, 3 and 4 are plotted in 2D (left) and 3D (right) view in the Figure 2. The red cells are the beliefs of occupied whereas the blue cells are the beliefs of empty spaces, and the flat space represents unknown. We call these set of functions *transfer functions* of the sensor model.

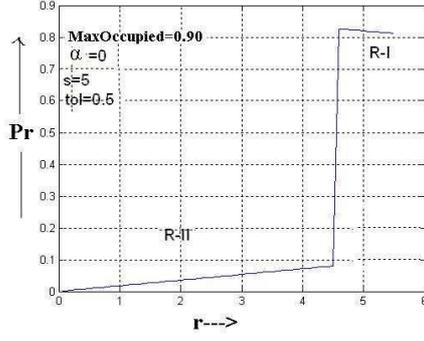


Figure 3. Probability trend of a sensor readout ($s=5$) through acoustic axis, i.e., $\alpha = 0$. $Max_{occupied} = 0.9$ and $Tolerance = \pm 0.5$ in **Region 1** from Equation 4. R-I and R-II denotes **Region-1** and **Region-2** respectively.

3.2 Analysis and Limitations of the Transfer Functions

If we compare the transfer functions in Figure 2 and the sensor model in Figure 1, we see some discrepancies. Theoretically for a good sensor model, **Region 1** should not contain any belief that leads the robot as *empty* space and that of **Region 2** denoting *occupied*. In the 2D representation in Figure 2, we see some red cells which denote *occupied*, at the corner of the sensor cone in **Region 2** as well as some blue cells which denotes *empty* at the corner of **Region 1**. In the 3D view of the sensor model, these cells (*red* in Region-2 and *blue* in Region-1) are seen as spikes in opposite direction i.e., upward peaks in **Region 2** and downward spikes in **Region 1**.

By plotting these transfer functions throughout the acoustic axis ($\alpha = 0$), we obtain a trend of believes or probabilities similar to the Figure 3. The following features are noticed in **Region 1**

- At the beginning of the region (from left to right) the belief is the highest and
- Up to end of the region the belief decreases gradually.

However, we argue that it is more reasonable to give the highest belief to the point, which is obtained as the return of the sensor reading and both sides of that point should contain a gradual lesser believes. For instance, if the sensor reading is 500 units, the point that stands exactly 500 units away from the origin of the sensor should be more reliable than others in **Region 1**.

On the other hand in the empty region, **Region 2** in the Figure 3, the lowest belief is zero, i.e., $Pr = 0$ (at the beginning). This raises an issue when a grid-cell is continuously updated at runtime by multiple sensors. For instance, consider the situation: if a grid element eg. $grid[i][j]$, contains zero or very low value close to zero belief, and if the same grid element is judged by another sensor reading as

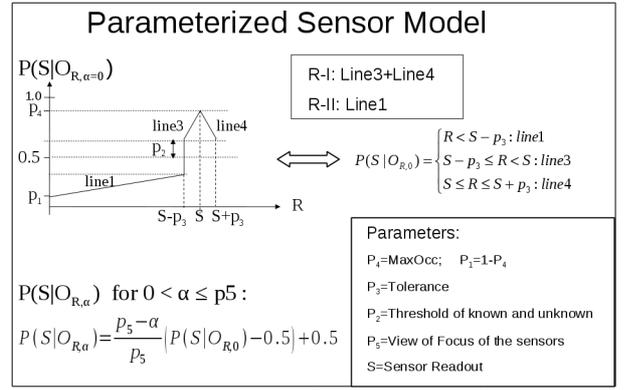


Figure 4. Proposed Parameterized Sensor Model.

occupied then the updating of that element either becomes zero or very slow.

By analyzing the transfer Equations 3 and 4 for this region, we see the highest belief can raise up to 0.5 when $r = R$ and $\alpha = 0$. We know that the belief value 0.5 states the status of a grid element eg. $grid[i][j]$ as *unknown* neither *Occupied* nor *Empty* which is not the correct interpretation for a good sensor model.

3.3 Bayesian Updating

The most common technique for fusing sensor data is to translate sensor readouts into probabilities and to combine those probabilities using the Bayes rule.

$P(s|H)$, as given by the sensor model, is converted into expected $P(H|s)$, by using Bayes' rule as follows:

$$P(H|s) = \frac{P(s|H)P(H)}{P(H)P(s|H) + P(\neg H)P(s|\neg H)} \quad (5)$$

Equation 5 generates the probability only one sensor readout. However, fusing more than one probabilities either from two different sensors sensing at the same time or at two different times, is also possible by using *Recursive Bayes' Rule*. The recursive Bayes' rule is as follows:

$$P(H|s_n) = \frac{P(s_n|H)P(H|s_{n-1})}{P(s_{n-1}|H)P(H|s_{n-1}) + P(s_{n-1}|\neg H)P(\neg H|s_{n-1})} \quad (6)$$

where $P(s_n|H)$ and $P(s_{n-1}|H)$ denote the probability at time t_n and t_{n-1} respectively. We refer the readers to Robin R. Murphy [10] for detail description and formula derivation.

4 Proposed Parameterized Sensor Model

To overcome the limitations with the sensor model described in Section 3, we introduce some more parameters. The sensor model is described in two steps. First we consider only grid cells $(0, R)$, i.e. cells at a distance R away

from the sensor, along the acoustic axis. To resolve the issues of slow updating and of controlling the difference between empty and unknown beliefs in **Region-2**, we introduce the parameters P_1 and P_2 . The diagram in Figure 4 describes the resulting sensor model as a function of R . $P(S|O_{R,\alpha})$, i.e. the probability for a grid cell α degrees off the acoustic axis is given by Equation 7.

If the view of focus of the sensor be P_5 , the distribution equation for $0 < \alpha \leq P_5$ range is as follows:

$$P(S|O_{R,\alpha}) = \frac{P_5 - \alpha}{P_5} (P(S|O_{R,\alpha=0}) - 0.5) + 0.5 \quad (7)$$

where P_5 a parameter replacing the previously constant β representing half the width of the sensing cone.

Equation 7 assures 1) that the probability for any grid cell off the acoustic axis in **Region 2** will never be larger than 0.5 and 2) that there will be no valley in **Region 1**, since the lowest probability is at least $(0.5 + P_2) \geq 0.5$ where $P_2 \geq 0.3$. With increasing values of α , $P(S|O_{R,\alpha})$ values are decreased. This is considered an important feature of a good sensor model [10, 4]. The 2D and 3D representations of our proposed sensor model are shown in Figure 5.

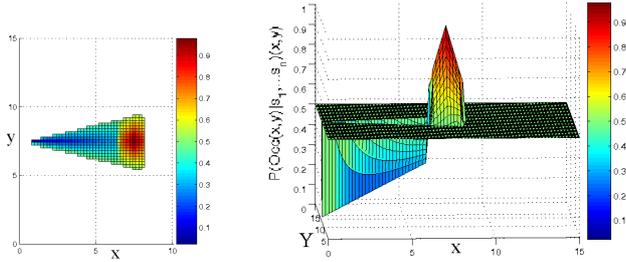


Figure 5. 2D (left) and 3D (right) view of proposed parameterized sensor model with sensor readout, $s=7.5$ units. X-axis is the sensor readout.

In this proposed **Parameterized Sensor Model**, we have used some parameters to control the probabilities/beliefs. Description of those parameters is as follows:

- P_1 : It is the threshold for minimum probability in **Region 2** on acoustic axis. To use $P_1 = 1 - P_4$ is a good idea but to assign individual/different value for this parameter is also interesting.
- P_2 : This parameter controls the highest probability in **Region 2** and the lowest probability in **Region 1** through out acoustic axis i.e., the threshold between known and unknown (Probability=0.5) grid cells through acoustic axis.
- P_3 : P_3 denotes the *tolerance* of the **Region 1**.
- P_4 : Maximum probability of occupancy for a grid-cell is controlled by the P_4 parameter. Since P_1 is the

maximum probability of empty of a grid element, one can consider $P_1 = 1 - P_4$. So $(Occupied_{Maximum} + Empty_{Maximum})=1$ ¹.

- P_5 : P_5 denotes the view of focus for a sensor readout.

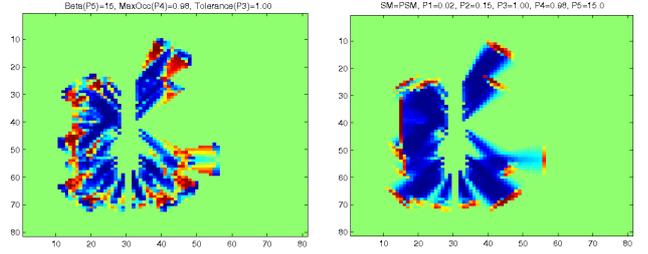


Figure 6. Map built by using Standard Sensor Model (described in **Section 3**) with $Beta=15$, $Tolerance=1.0$, $Max_{Occupied} = 0.98$ (left) and Proposed model, with some arbitrary parameters values; $P_1 = 0.02$ ($P_1 = 1 - P_4$), $P_2 = 0.15$, $P_3 = 1.0$, $P_4 = 0.98$, $P_5 = 15$ (right).

To compare the generated maps with our proposed sensor model and the standard sensor model discussed in Section 3, maps of the same environment were generated using both models. The common parameters have the same values in both cases and the generated maps are shown in Figure 6. The leftmost map is generated using the standard sensor model while the rightmost map is generated using the new proposed model. The red areas represent occupied places (Region-1) whereas the blue areas represent empty places (Region-2). The algorithm described in Section 3 is used to generate the maps. The robot was moving back and forth (in U turn fashion) in a room containing straight walls and two open doors.

The walls (red), the empty spaces (blue), and the openings (doors) are more visible in the rightmost map than in the leftmost one. Moreover, in the left image many areas contain a mixture of red and blue colors, which can be considered as errors in this spatial representation. Thus, the proposed sensor model generates a better representation of the environment than standard sensor model. For these tests, the following parameters were used: $P_1 = 0.02$, $P_2 = 0.15$, $P_3 = 1.0$, $P_4 = 0.98$, $P_5 = 15$.

It is reasonable to assume that the proposed model works better with some specific set of parameters. We denote this parameter set the "optimal parameter set". The aim of the parameter optimization is to find this optimal parameter set. For this, estimates of map goodness are used, as described in the next section.

5 Map goodness without Ground-map

To measure the goodness of maps without a ground map we have opted various statistical methods. The hypothesis

¹ $Empty_{Maximum}$ means the lowest probability.

is that a good sensor model generates the same or a closer belief of a cell in the occupancy grid either the cell denotes empty or occupied space in the physical environment. Therefore, the standard deviation of a cell after the generation of the map should be as small as possible. For example, assume that $grid[x][y]$ represents an occupied cell by several sensor readings. So $grid[x][y]$ is updated by several different or equal belief values obtained from sensor model and using the Bayesian rules. If the values obtained from sensor model are more similar, the sensor model is more optimal. This is our *first technique*.

The *second approach* is to find the noise in a map by using derivative of image (described "Derivative of maps and Comparison" below in detail) and comparing the noise among maps. The map containing the lowest noise is the best one. Our *third approach* is to run the robot more than one time along the same path, and to compare the resulting maps. Below, we describe the three approaches in detail.

First Approach: Mean of Standard Deviations For each cell of the grid for a single map, the generated numerical values from our proposed model have been saved and standard deviation of those values has been calculated. If the grid's dimension be $L \times M \times N$, for each cell $X_{i,j}$ in $L \times M$ plan the standard deviation over third dimension (towards N) is $\sigma_{i,j} = \sqrt{\frac{1}{N-1} \sum_{k=1}^N (X_{i,j,k} - \bar{X}_{i,j})^2}$; where $i \in L$ and $j \in M$, and $\bar{X}_{i,j} = \frac{1}{N} \sum_{k=1}^N X_{i,j,k}$. Accordingly for the whole map, we get $L \times M$ dimension matrix MSD containing the standard deviation, σ for each cell as

$$MSD = \begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} & \dots & \sigma_{1,M} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \sigma_{L,1} & \sigma_{L,2} & \dots & \sigma_{L,M} \end{pmatrix}$$

But we are interested only to non zero elements of MSD. A vector consisting of non zero $\sigma_{i,j}$ is formed from the MSD and this vector we call *vector of standard deviation of non zero sigma* (VSDNS). Thus, $VSDNS = [\sigma_1 \sigma_2 \dots \sigma_P]$; where $\sigma_p = \sigma_{i,j} \neq 0$; $p = 1, 2, 3 \dots P$ and P is the total number of nonzero element of matrix MSD. The final VSDNS value of a map is equivalent to its arithmetic mean i.e., $VSDNS = \bar{\sigma} = \frac{1}{P} \sum_{p=1}^P \sigma_p$.

We have generated 1440 maps with different combination of parameters (see Table 1) and we have calculated $VSDNS_l$ where $l = 1, 2 \dots 1440$. After generating a histogram² for 1440 VSDNS with 14 bins, we have found that the histogram represents a normal distribution showed in Figure 7. The table of the figure shows the value containing each bin.

Since the lower standard deviation vectors (VSDNS) represent better parameter set, the lowest bin of the histogram denotes the optimal parameter set. The width of the bin containing lowest values, for instance values between 0.054079 and 0.059045, we call Expected Interval or Flat

²Histogram: histogram is defined as a bar graph that shows frequency data.

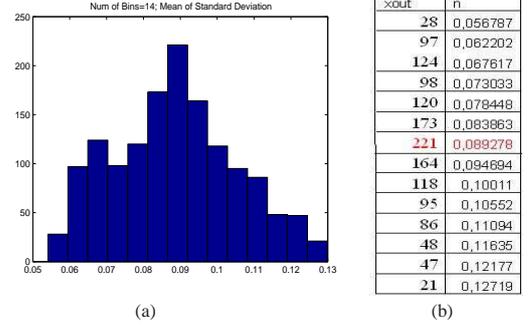


Figure 7. (a)Histogram of the *mean of standard deviation*. (b)Table containing values of histogram.

Interval for the parameters set and we find all maps and parameter sets responsible for this interval. Thus, the average of individual parameter over these parameter sets denote the optimal parameter set. Table 1 shows the range of different parameters used for testing this method (2nd, 3rd and 4th column) and the 5th column contains the obtained optimum parameter set. In Figure 8, sub figure (d) contains the optimal parameter set found by this method.

On the other hand, if we consider the minimum VSDNS instead of the average, the obtained parameter set is 6th column of the table and the generated map is sub figure (e). Again if we consider \overline{VSDNS} instead of using histogram, we get the sub figure (f) which is remarkably better than sub figure (d) and (e) in Figure 8 and the values of parameters are in 7th column of the table 1 although this is a crude method.

From Table 2, which shows the detail of the first bar of the histogram in Figure 7, we see some tendencies in individual parameters. Lower values of **P3**, **P5** and higher values of **P2** give less VSDNS whereas **P1** and **P4** don't show any specific patterns. This can be seen as a *stability test* for the parameters. A major concern with this method of using histogram is to decide a suitable number of bins.

Second Approach: Derivative of maps and Comparison

Here the hypothesis is that *each cell of the grid contains similar value with respect to its nearest neighbors*. This can be obtained by using the concept of derivatives in an image [5]. By calculating this derivative for each cell and by applying statistical measurement technique such as mean or sum, we can achieve a metric for the map. Since the derivative function denotes the discrepancy between cells, we call this measurement *noise* of a map. The map containing less noise is better and thus we get the optimal parameter set for the sensor model.

To calculate the noise of a map we have used summation statistical measurement and we denote the 2D map grid $G=M \times N$ and accounted neighbor window $W = m \times n$. For any cell $G(x,y)$ of the grid, the derivative $D_{x,y}$ is calculated as:

Parameter	Min	Step Size	Max (using histogram)	Optimum ($VSDNS_{minimum}$)	Optimum (\overline{VSDNS})	Optimum
*P1	0.02	0.01	0.05	0.03	0.02	0.02
P2	0.05	0.05	0.2	0.171	0.2	0.05
2xP3	0.5	0.5	3.0	0.786	0.5	2.0
P4	0.95	0.01	0.98	0.97	0.98	0.98
P5	6.0	**arbitrary	40.0	6.14	6	14

Table 1. Used parameters set with minimum(2nd column), incremental step size (3rd col) and maximum(4th col) range. 5th col denotes the optimal parameter set by using lowest column of histogram mentioned in Figure 7. 6th col contains the lowest VSDNS generated set and 7th col is obtained by taking the arithmetic mean over 1440 VSDNS. *P1:P1=1-P4 relation has been used. ** Values are arbitrarily selected {6, 7, 8, 10, 13, 14, 15, 16, 17, 18, 20, 25, 27, 35, 40} and attention is payed at the area of commonly used value eg. 15.

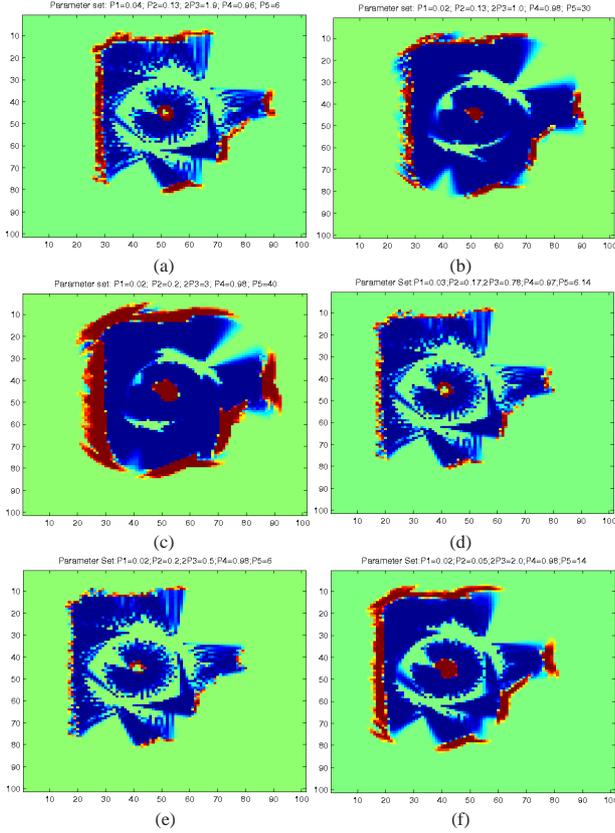


Figure 8. Finding optimal parameter set by *mean of standard deviation* method.

$$S_{x,y} = \sum_{i=x-\lfloor m/2 \rfloor}^{x+\lfloor m/2 \rfloor} \sum_{j=y-\lfloor n/2 \rfloor}^{y+\lfloor n/2 \rfloor} abs(W_{i,j} - W_{x,y});$$

$$D_{x,y} = S_{x,y} / (m \times n - 1);$$

where, $x = 2, 3, \dots, M - 1$ and $y = 2, 3, \dots, N - 1$. The noise or derivative of the map D_{map} is calculated from the derivatives of cells by $D_{map} = \sum_{x=2}^M \sum_{y=2}^N D_{x,y}$. Note that if the maps are different in sizes then normalization is essential to avoid scaling effect.

In our implementation, we use 8 connecting neighbors and we apply this technique on 1440 maps generated

P5	P4	2P3	P2	VSDNS
6	0.98	0.5	0.2	0.054079
6	0.97	0.5	0.2	0.054256
6	0.96	0.5	0.2	0.054435
6	0.95	0.5	0.2	0.054615
6	0.98	0.5	0.15	0.055861
6	0.97	0.5	0.15	0.056043
6	0.96	0.5	0.15	0.056228
6	0.95	0.5	0.15	0.056414
6	0.98	1	0.2	0.056517
6	0.97	1	0.2	0.056663
6	0.96	1	0.2	0.056811
6	0.95	1	0.2	0.056961
7	0.98	0.5	0.2	0.057684
6	0.98	0.5	0.1	0.057705
7	0.97	0.5	0.2	0.05787
6	0.97	0.5	0.1	0.057893
7	0.96	0.5	0.2	0.058058
6	0.96	0.5	0.1	0.058082
6	0.98	1	0.15	0.058194
7	0.95	0.5	0.2	0.058248
6	0.95	0.5	0.1	0.058273
6	0.97	1	0.15	0.058347
6	0.96	1	0.15	0.0585
6	0.95	1	0.15	0.058655
6	0.98	1.5	0.2	0.058682
6	0.97	1.5	0.2	0.058801
6	0.96	1.5	0.2	0.058922
6	0.95	1.5	0.2	0.059045

Table 2. This table contains all parameter sets obtained from the first bar of histogram mentioned in figure 7. *P1:P1=1-P4 relation has been used.

by using different values of parameters same as used in the first approach, which is mentioned in Table 1. We have found P1=0.05 (1-P4); P2=0.2; P3=0.5; P4=0.95; P5=40 parameter set as the optimal and the generated map is shown in Figure 9).

We see that the map found as the best one in this approach is not reflecting the actual best one. Rather this map

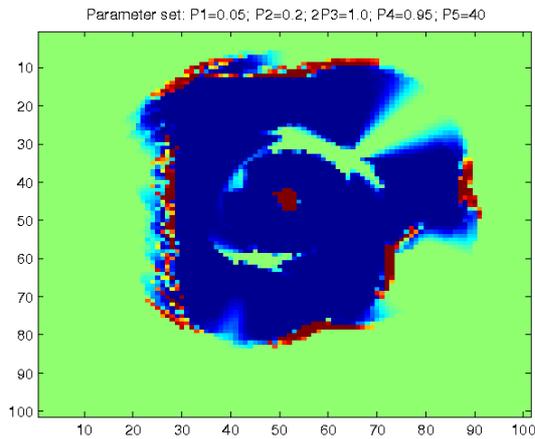


Figure 9. Optimal map produced by *derivative of maps* method and the parameters are $P1=0.05$ (1- $P4$); $P2=0.2$; $P3=0.5$; $P4=0.95$; $P5=40$.

is worse than most of the maps showed in the Figure 8. Thus, this approach is not a good measurement for goodness of maps.

Third Approach: Comparing two different maps of the same Environment The ground assumption of this approach is that in an environment the generated maps should be similar as many times we build a map of it. Provided that the robot is driven every time from the same position with the same direction, same speed and same exploring behavior. Although it is not obvious, we consider the parameter set, which generates similar maps in every run, is the optimum.

We run the robot twice using the same set of parameters and other constraints such as direction, speed and behavior. The resulting two maps A and B should be similar for a good sensor model. By using a threshold value, we transform these maps into binary form such that 1 represent occupied cells and 0 represent empty cells. We then compare these two binary maps and score them according to Moravec and Elfes [9] as follows: for each pair of cells that is equal in both maps A and B , we increase a positive score. If the cells are not equal, we increase a negative. The optimum parameter set is the one that gets the maximum positive score and minimum negative score.

We have applied this technique on 1440 different combinations of parameters of our proposed sensor model. The optimal parameter set is $P1=0.02$ (1- $P4$); $P2=0.05$; $2*P3=3$; $P4=0.98$; $P5=35$ and the map generated by using them is shown in Figure 10.

We see that the best map found in this approach is better than most maps showed in Figure 8 and at the same time is worse than some others eg. maps showed in sub-figures (a) and (f). This entails that this method is not performing as good as expected in hypothesis.

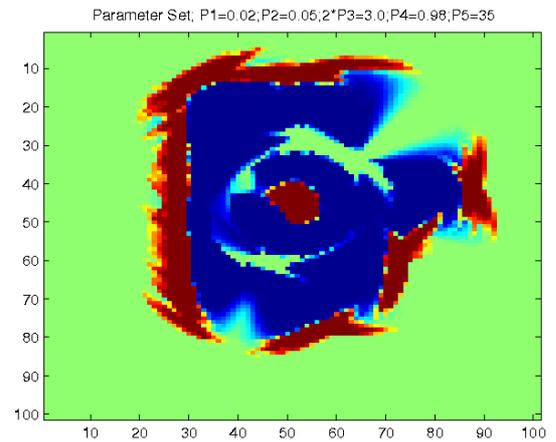


Figure 10. Optimal map by *comparing two different maps of same environment* method and the parameters are $P1=0.02$ (1- $P4$); $P2=0.05$; $2*P3=3$; $P4=0.98$; $P5=35$.

6 Discussion and Conclusions

In this paper we proposed an improvement of the sensor model described in R. Murphy [10]. To find the optimal parameter set for our model, we have tried to use several statistical approaches for measuring the goodness of maps without using the ground map or the original map of the environment. Although we have got some better parameter set by some methods, there exist better set of parameters eg. rightmost sub-figure in Figure 6 or sub-figure (a) in Figure 8, which are not found by our implemented techniques. Thus, none of our evaluated measurements of map goodness produced satisfactory results. In our first approach, the mean of standard deviations (\overline{VSDNS} process) produces better results than others. Despite these inconclusive results, we have found that our proposed sensor model produces better results than the standard sensor model (described in Section 3 and Figure 2) with same parameter set (see Figure 6).

The presented work relies on manual evaluation of generated maps. For future work in the area, an objective way of estimating map quality should be developed.

7 Acknowledgement

We would like to thank the Computer Science department of Umea University for assisting all kind of resources for this work. Thanks to Shafkat Kibria, Erik Billing, and Fabian who spent their valuable time to discuss with us during the work.

References

- [1] A. Elfes. Sonar-based real-world mapping and navigation. *Journal of Robotics and Automation*, 3:249–

265, 1987.

- [2] A. Elfes. Occupancy grids: A stochastic spatial representation for active robot perception. *Proc. of the Sixth Conf. on Uncertainty in AI*, pages 60–70, 1990.
- [3] M. Fichtner and A. GroBmann. A Visual-Sensor Model for Mobile Robot Localisation. Technical report, AI Institute, Dept. of Computer Science, Technische Universitet Dresden, Germany, 2003.
- [4] T. Helstrom. Class lecture of intelligent robotics course(2005, spring semester). Dept. of Computer Science, Umea University, Sweden.
- [5] B. K. P. Horn. *Robot Vision*. The MIT Press, 1986.
- [6] T. Kohonen. *Self-Organizing Maps*. Springer Series in Information Sciences., Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001.
- [7] M. C. Martin and H. P. Moravec. Robot Evidence Grids. Technical report, The Robotics Institute, CMU, Pittsburgh, PA 15213, USA, 1996.
- [8] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9:61–74, 1988.
- [9] H. P. Moravec and A. Elfes. High Resolution Maps from Wide Angle Sonar. Technical report, The Robotics Institute, CMU, Pittsburgh, PA 15213, USA, 1984.
- [10] R. R. Murphy. *Introduction to AI Robotics*. MIT Press, 55 Hayward Street Cambridge, MA 02142-1493, USA, 2000.
- [11] U. Raschke and J. Borenstein. A comparison of grid-type map-building techniques by index of performance. *In Proc. of IEEE Int. Conf. on Robotics and Automation*, 3:1828–1832, 1990.
- [12] ActiveMedia Robotics. Amigobot Website. <http://robots.amigobot.com/SHAmigos.html>.
- [13] W. Rencken. Autonomous sonar navigation in indoor, unknown and unstructured environments. *In Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, pages 431–438, 1994.
- [14] S. Thrun. Robotic Mapping: A Survey. Technical report, School of Computer Science, CMU, Pittsburgh, PA 15213, USA, 2002.
- [15] A. Ypma and R. P. W. Duin. Novelty detection using self-organizing maps. *Progress in Connectionist-Based Information Systems - Proceedings of ICONIP97*, pages 1322–1325, 1997.