A. Birk, A Quantitative Assessment of Structural Errors in Grid Maps, Autonomous Robots, vol. 28, pp. 187-196, 2010. DOI: 10.1007/s10514-009-9159-2

A Quantitative Assessment of Structural Errors in Grid Maps

Andreas Birk
Jacobs University
Campus Ring 1, 28759 Bremen, Germany
a.birk@jacobs-university.de

Abstract

Various common error sources affect the quality of a map, e.g., salt and pepper noise and other forms of noise that are more or less uniformly distributed over the map. But there also exist errors that only occur very rarely in the mapping process but that have severe effects on the final result. They influence not only the local accuracy but also the whole spatial layout of the map. Examples of related error sources include bump noise in the robot's pose or residual errors in Simultaneous Localization and Mapping (SLAM). The concept of brokenness is introduced in this article to capture the notion of structural errors in grid maps. The map is partitioned into regions that are locally consistent with ground truth but "off" relative to each other. Brokenness measures the number of these regions and their spatial relations. A theoretical basis is introduced to derive the concept of brokenness in a formal way. Furthermore, it is shown how brokenness can be computed in an algorithmic way. Experiments with maps from simulated as well as real world data are presented. They show that the metric can indeed be used to automatically determine the structural quality of a map in a quantitative way.

Keywords:

mobile robot performance metric map quality ground truth comparison topology registration

1 Introduction

Mapping is a core task for the whole field of mobile robotics (Thrun, 2002). It should hence be of strong interest to be able to assess the quality of a map, or of a mapping approach in well defined ways. But the benefits of a map or of a mapping algorithm are usually presented through differences in the underlying theories, in the run times, and at most in a qualitative visual assessment of map quality, e.g., by displaying the generated maps and a comparison basis like the results of other approaches or a ground truth layout. The necessary references for this qualitative comparison, i.e., results from other mapping approaches or ground truth, are surprisingly often omitted. This is a general problem as for example also pointed out by Chandran-Ramesh and Newman (2008): "hardly any effort has been expended to determine the quality of a map once it is built".

One way to address the problem is a purely task oriented viewpoint; Leonard and Durrant-Whyte (1992) formulated it as follows: "We feel the ultimate test of a map is not 'does it look good?' but 'how accurately, adequately or quickly can X be performed with it?'". There is no general quality of a map or of a mapping algorithm from this viewpoint. Instead, each user has to run individual benchmarks based on the task X for which the map is to be used. Though there is a grain of truth in the above statement, it is not a sufficient motivation to completely abandon general ways to assess map quality.

One simple, general criterion for the quality of a map M is for example its *coverage*, i.e., the amount of the ground truth that is represented in M (Lee, 2003). Further general criteria for map quality are its *level of detail* and its *accuracy* (Lee, 2003). 2D occupancy grids (Elfes, 1989; Moravec and Elfes, 1985) are the predominant form of map representation. The level of detail in a grid map is determined by the resolution of M. The number of cells that contain information determines its coverage. What remains is the problem of measuring accuracy.

One option for measuring the accuracy of a map is to use assumptions about the environment, e.g., the presence of planar walls, and to quantify the local consistency of this assumption (Chandran-Ramesh and Newman, 2008). But this approach has its limits as it leads to false results for maps, or parts of a map where the assumptions about the environment properties do not hold.

An other approach is to use the uncertainty information that is linked to the map in case a Simultaneous Localization and Mapping (SLAM) algorithm is used. Concretely, the overall uncertainty in the final result of the SLAM can

be seen as a measure of the confidence in the accuracy of the result (Frese, 2006). But this leads to a chicken and egg problem. This approach assumes that the SLAM algorithm does not contain any faults, i.e., that increasing accuracy in the map leads to decreasing uncertainties in the SLAM. But a measure for map accuracy is supposed to test the correctness of the underlying mapping algorithm; it should not depend on it.

A way out of this dilemma is the usage of a comparison basis in form of a reference map like a ground truth layout or the result of an other mapping algorithm. Repositories with ground truth data and raw sensor streams are of interest in this context. Examples include the Robotics Data Set Repository (RADISH) and the Rawseeds website. Further options for the generation of references are standardized test scenarios in simulated as well as real world environments (Scrapper et al, 2007, 2008).

Given a ground truth or other reference representation R, it is possible to estimate the accuracy of a map M by relating the content of each cell in M to the content of the corresponding cell in R. This can, for example, be done with cross entropy (H. Moravec, 1993) as a correlation estimate. Cross entropy has a major disadvantage: It only takes information from co-located cells in M and R into account. The map has hence to be very similar to the reference to give any meaningful accuracy estimate. Cross entropy fails to provide information about the map quality if there are any larger disturbances.

The obvious way to overcome this disadvantage is to employ a measure of distance between similar points in M and R, for example in form of the Least Mean Squares of Euclidean Distances (LMS-ED) (Yairi, 2004). LMS-ED has a crucial disadvantage, namely it is expensive to compute. It is therefore not convenient to use it for grid maps. One option is to restrict the computations to limited set of landmarks (Wnuk, 2005; Jose-Luis Blanco, 2007). But this makes the accuracy measure dependent on the way the landmarks are detected.

An alternative to LMS-ED is an approach based on Manhattan Distances (Varsadan et al, 2009). It allows a very fast computation of a map quality, which provides quantitative assessments of the level of noise in the M. Concretely, this map quality metric shows decreasing values for increasing amounts of common error sources like salt and pepper noise and the global effects of translation, rotation, and scaling.

All the above methods have in common that they only address errors that are uniformly distributed over the map. In this article, a form of structural error is addressed, namely the "brokenness" of a map or the degree with which a map can be partitioned into submaps that are "off" with ground truth relative to each other. Figure 1 shows a typical example. Suppose map (a) represents ground truth. Map (b) is broken in the sense that a complete partition, namely the room shown in the bottom left in this map, is rotated with respect to ground truth. Please note that this particular partition, i.e., the submap that covers this room, is consistent within itself. It can be considered as a good representation of the corresponding part in ground truth. But this submap will lead to very low accuracy measures if a cell based correlation metric is used to measure the quality of the overall map.

Please note that the term *structural* error is used in this article for faults in the map that affect its global spatial layout. The alternative term "topological" is omitted for two main reasons: First of all, "topological" may be misleading with respect to the representation that is used. Topological maps in form of graphs explicitly represent a spatial structure. In grid maps, the spatial structure is only in the eye of the human observer. The underlying data structure is nothing but a uniform grid. Second, topological maps are usually based on higher-level spatial information, i.e., the building blocks of the map have a semantic quality like "room", "corridor", etc. This is not the case for grid maps. It is mere coincidence that a whole room is "broken" in figure 1. Structural errors in grid maps usually do not correspond to any semantically meaningful part of the map.

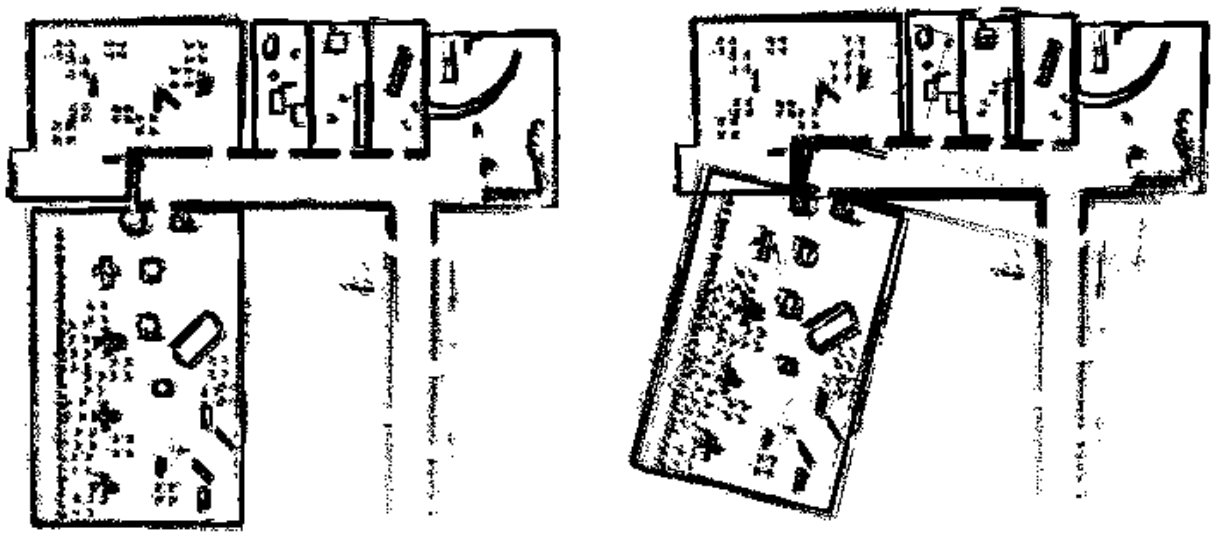
We consider brokenness to be an important, if not even the most important, form of structural error in grid maps. The source of brokenness is a form of error that occurs very regularly in robot mapping but that is usually neglected, namely bump noise. Bump noise occurs only very rarely in contrast to Gaussian noise on sensor readings. There is a - usually very small - probability p_B that at time t the value $v_B(t)$ of this noise is non-Zero. Gaussian noise is in contrast almost always non-Zero, i.e., at every moment t it is superimposed on the sensor readings. Despite its rare occurrence, bump noise has severe effects as it can lead to large "jumps" in sensor readings with severe effects on localization estimates (Gutmann et al, 1998).

In the context of mapping, bump noise can be modeled as

$$v_B(t) = (\theta, d) = \begin{cases} \theta \in \mathcal{U}[0, 2\pi], d \in \mathcal{U}[0, r] & \text{with prob.} p_B \\ (0, 0) & \text{with prob.} 1 - p_B \end{cases}$$

where θ is an angle and d is a translation by a certain distance that are both superimposed on the estimated location of a robot at time t, and $\mathcal{U}[c_1, c_2]$ is a uniform distribution of samples over the interval $[c_1, c_2]$. Roughly speaking, $v_B(t)$ changes the estimated orientation of the robot by an arbitrary angle and shifts its estimated position within a certain radius r on rare occasions that have probability p_B to occur; but most of the time there is no effect of $v_B(t)$.

Typical sources of bump noise in mapping are slipping wheels or tracks when using odometry or an incorrect re-



(a) undisturbed map (b) ”broken” map

Figure 1: A typical example of a “broken” map where a complete partition of map (b) - namely the whole room on the bottom left - is “off” by an angular error compared to (a) as a representation of ground truth.

sult of a scan matcher. One may argue that state of the art Simultaneous Localization and Mapping (SLAM) algorithms are well suited to compensate this kind of error (see (Thrun et al, 2005; Thrun, 2002; Frese, 2006) for a general overview of SLAM). But SLAM requires loop closing for this purpose and one can not assume that all parts of all maps are always a part of a loop. And even if this would be the case, it is not unlikely that a residual error remains after relaxation or filtering in the loop closing process. Please note that a single - potentially residual - bump error in just one localization estimate among the thousands that form the basis of a map can lead to the severe effects of brokenness as illustrated in figure 1.

The rest of this article is structured as follows. In section 2, a formal definition of brokenness is introduced. This includes a formal derivation of the degree of brokenness of a map M in form of the number of consistent partitions into which M can be divided. A general algorithmic way to compute the degree of brokenness is presented in section 3. It is based on the idea of using recursive registration and masking steps to identify the partitions and the spatial relations between them. In addition to the general algorithmic formulation, a concrete way of implementation is introduced. Experiments and results are presented in section 4. It is shown with several maps from simulation as well as from real world data that the correct degree of brokenness can indeed be computed in an automatic way to assess the structural quality of a map. Section 5 concludes the article.

2 A formal definition of Map Brokenness

As mentioned before, this article deals with the predominant form of maps as 2D arrays. It is assumed here that each cell of a map only contains binary information. This binary information can be seen as a predicate p that is fulfilled or not for this cell. A typical example is *occupancy*, i.e., the fact whether a location represented by a cell corresponds to free space or not. Please note that all concepts presented here can be easily extended to more general forms of maps where the cells contain more detailed information like probabilities or pixel values. But as a matter of convenience, it is assumed that a map $\mathbf{M}$ of size $k_x \times k_y$ consists of cells $\mathbf{m}(x, y)$ with

$$\mathbf{M} = \{\mathbf{m}(x, y) \mid \forall_{1 \leq x \leq k_x, 1 \leq y \leq k_y} \; \mathbf{m}(x, y) = p \;\vee\; \mathbf{m}(x, y) = \neg p\}$$

Based on this definition of a map $\mathbf{M}$, it is possible to define the set $\mathbf{L}^{\mathbf{M}}$ of locations (x, y) where the predicate p, for example occupancy, is fulfilled:

$$\mathbf{L}^{\mathbf{M}} = \{(x, y) \mid \mathbf{m}(x, y) = p\}$$

It is obvious that there is a canonical way to compute $\mathbf{M}$ from a given $\mathbf{L^M}$ and vice versa. In the following, we will only deal with sets of locations $\mathbf{L^M}$ with property p and use this as a synonym for grid maps. A spatial transform $\mathcal{T}_{\theta,d}$ with a rotation θ followed by a translation d on a map $\mathbf{L^M}$ is defined as:

$$\mathcal{T}_{\theta,d}(\mathbf{L^M}) = \{(x',y') \mid \forall (x,y) \in \mathbf{L^M} : (x',y') = \mathbf{t}_{\theta,d}(x,y)\ \}$$

where $\mathbf{t}_{\theta,d}(x,y)$ is the spatial transform of the point $(x,y) \in \mathbb{R}^2$ with a rotation by the angle $\theta \in [0, 2\pi]$ followed by a translation by the distance $d \in \mathbb{R}$.

Finally, the operator $\overset{\mathcal{T}_{\theta,d}}{\rightarrow}$ applied to two maps L^1 and L^2 denotes the set union of the points in L^1 with the points in L^2 after a non-trivial spatial transformation of the later ones, i.e.,

$$\mathbf{L^1} \overset{\mathcal{T}_{\theta,d}}{\rightarrow} \mathbf{L^2} \hat{=} \mathbf{L^1} \cup \mathcal{T}_{\theta,d}(\mathbf{L^2})$$

with $\theta \neq 0 \vee d \neq 0$.

Based on the above notations, the brokenness of a map $\mathbf{M}$ can be defined as follows:

Definition: A map $\mathbf{M}$ is broken with degree $n_{BN} \geq 1$ with respect to a reference $\mathbf{R}$ if

$$\exists_{0 \leq i \leq n_{BN}}\ \mathbf{L^i},\ \exists_{1 \leq j \leq n_{BN}}\ \mathcal{T}^j$$

with

$$\mathbf{L^M} = \mathbf{L^0} \cup \mathbf{L^1} \cup ... \cup \mathbf{L^{n_{BN}}}$$

and

$$\mathbf{L^R} = \mathbf{L^0} \overset{\mathcal{T}^1}{\rightarrow} \mathbf{L^1} ... \overset{\mathcal{T}^{n_{BN}}}{\rightarrow} \mathbf{L^{n_{BN}}}$$

Intuitively, the degree n_{BN} of brokenness counts the number of transformations on portions into which a map $\mathbf{M}$ is partitioned due to structural errors. Each $\mathbf{L^i}$ is a consistent submap, i.e., it is locally correct in the sense that it can be perfectly aligned with the corresponding part in the reference map $\mathbf{R}$. But the portions $\mathbf{L^i}$ have to be spatially transformed to compensate the structural errors to get a globally correct map. The reference map $\mathbf{R}$ is a ground truth representation in the best case. As a matter of convenience, ground truth is assumed to be the reference if nothing else is explicitly mentioned. This case is also referred to as the general brokenness of a map.

Without loss of generality, it can be assumed that the "sub-maps" are proper partitions that can be ordered by their cardinality #, i.e., it can be assumed that

$$\forall i \neq j : \mathbf{L^i} \cap \mathbf{L^j} = \emptyset$$

and

$$\#\mathbf{L^0} \geq \#\mathbf{L^1} \geq ... \geq \#\mathbf{L^{n_{BN}}}$$

This property can be used to define more fine-grained metrics for brokenness that take the sizes of the different partitions into account. Consider for example the case when two maps $\mathbf{M_1}$ and $\mathbf{M_2}$ are both broken with the same degree k. It can then be of interest to compare the size of the largest consistent partition $\mathbf{L_1^0}$ of $\mathbf{M_1}$ with the size of the largest consistent partition $\mathbf{L_2^0}$ of $\mathbf{M_2}$. Furthermore, it can be of interest to apply a norm to the parameters of the spatial transforms $\mathcal{T}$ needed to get the partitions into a globally consistent map.

In the following section, a general algorithmic approach to compute the degree of brokenness n_{BN} is introduced. It also leads to a determination of the proper partitions - and hence their sizes - as well as the underlying spatial transforms between them.

3 Computing Brokenness

3.1 A General Approach

A general algorithmic approach to compute the brokenness of a map is presented in this section. It is based on the idea of employing map merging to determine the different partitions in a map with respect to a reference map. Map merging deals with the fusion of two maps. It is based on the detection of "identical" regions in the maps where they can be "glued" together. Map merging is, for example, of interest in the context of multi robot mapping (Thrun et al, 2000; Thrun, 2001; Madhavan et al, 2004; Dedeoglu and Sukhatme, 2000; Howard, 2004; Williams et al, 2002; Fenwick et al, 2002; Roy and Dudek, 2001; Ko et al, 2003).

Map merging is also related to image registration, i.e., the search process of finding a template in an image (Stricker, 2001; Dorai et al, Jan., 1998; Brown, 1992; Alliney and Morandi, 1986). But the task of map merging is harder than just registration. Please note that instead of locating a known template in an image, an unknown region of overlap has to be identified in two maps for map merging. This is more comparable to image stitching (Lowe, 2004). Image stitching is for example used to generate panoramic views from several overlapping photographs. Solutions from image processing need common reference points that are identified using local image descriptors (Lowe, 2004; Mikolajczyk and Schmid, June, 2003; Gool et al, 1996).

But occupancy grids usually lack rich textures like photographs. Image stitching techniques can hence not be applied in a straightforward manner. But there are several recent advances in map merging (Carpin, 2008a,b; Birk and

Carpin, 2006; Carpin et al, 2005) that provide the means for registering the partitions as needed here. Please note that the general formulation of the computation of brokenness is not dependent on a concrete map merging algorithm and that there are several possible choices for implementing it.

Algorithm 1 The general algorithmic approach to determine the degree of brokenness n of a map $\mathbf{M}$ with respect to a reference $\mathbf{R}$.

$T = L^{\mathbf{M}}$
$n_{BN} = 0$
while $T \neq \emptyset$ **do**

< *register* T *with reference* $\mathbf{R}$ >
find ${}^{n_{BN}}\mathcal{T}_{\theta,d}() : \max \Psi_1({}^{n_{BN}}\mathcal{T}_{\theta,d}(T), R')$ for $R' \subseteq L^{\mathbf{R}}$
$T = {}^{n_{BN}}\mathcal{T}_{\theta,d}(T)$

< *mask out the part of* T *that is well aligned with* $\mathbf{R}$ >
find $\max \mathbf{L}^{\mathbf{n_{BN}}} \subseteq T : \Psi_2(\mathbf{L}^{\mathbf{n_{BN}}}, R'') \leq c$ for $R'' \subseteq L^{\mathbf{R}}$
$T = T \setminus \mathbf{L}^{\mathbf{n_{BN}}}$

< *increment the degree of brokenness* >
$n_{BN} = n_{BN} + 1$

end while

Given a map $\mathbf{M}$ and a reference map $\mathbf{R}$, e.g., ground truth, a way to compute the degree of brokenness n as formally defined in section 2 is now presented.

In the most general way as formulated in algorithm 1, a metric $\Psi_1(X_1, X_2)$ is needed to determine the "similarity" between two raster data sets X_1 and X_2. The registration step in algorithm 1 is used to determine a spatial transform $\mathcal{T}_{\theta,d}()$, which finds the best, i.e., most similar, match of a partition in the reference $\mathbf{R}$ with a partition of the transformed map $\mathcal{T}_{\theta,d}(T)$. In the beginning of the algorithm, T is a copy of $\mathbf{M}$. Roughly speaking, T is "moved around" until the largest possible region matches with a corresponding region in the reference $\mathbf{R}$.

In the following step, the matching regions are masked out from the maps T and $\mathbf{R}$. Again, a similarity metric $\Psi_2()$ is used. This second similarity metric $\Psi_2()$ may be identical with $\Psi_1()$. In the previous registration step, the largest partition that is consistent with the reference is determined, i.e., it is registered by a transform $\mathcal{T}()$ with the corresponding region in the reference $\mathbf{R}$. Cells $m(x, y)$ in this matching partition of $\mathcal{T}(T)$ have hence a high similarity to co-located cells $r(x, y)$ in $\mathbf{R}$. This property is now used to remove this partition $\mathbf{L}^{\mathbf{n_{BN}}}$ from the current T. Then, the registration and masking steps are recursively applied to the remainder $T \setminus \mathbf{L}^{\mathbf{n_{BN}}}$.

3.2 Concrete Algorithms for the Registration and Masking

As mentioned before, the general principle of computing brokenness can be achieved with fundamentally different implementations of the registration and masking steps. Here, a map merging algorithm introduced in (Birk and Carpin, 2006) is used. It is based on a metric D_ψ to measure the (dis)similarity of 2D raster data sets $\mathbf{M_1}$ and $\mathbf{M_2}$ (Birk, 1996):

$$D_\psi(\mathbf{M_1}, \mathbf{M_2}) = d(\mathbf{M_1}, \mathbf{M_2}) + d(\mathbf{M_2}, \mathbf{M_1})$$

with

$$d(\mathbf{M_1}, \mathbf{M_2}) = \sum_{m_1(x_1,y_1)=p} \min\{MD((x_1, y_1), (x_2, y_2)) \mid m_2(x_2, y_2) = p\}$$

where $MD((x_1, y_1), (x_2, y_2)) = |x_1 - x_2| + |y_1 - y_2|$ is the Manhattan-Distance between (x_1, y_1) and (x_2, y_2).

The metric D_ψ is supplemented by a correlation metric that computes the number of cells where the contents in the two raster datasets *agree*, i.e., where they are identical, respectively *disagree*:

$$agr(\mathbf{M_1}, \mathbf{M_2}) = \#\{(x, y) \mid m_1(x, y) = m_2(x, y)\}$$

$$dis(\mathbf{M_1}, \mathbf{M_2}) = \#\{(x, y) \mid m_1(x, y) \neq m_2(x, y)\}$$

Based on these notations, the $\Psi_1()$ used here is defined as:

$$\Psi_1(\mathbf{M_1}, \mathbf{M_2}) = -D_\psi(\mathbf{M_1}, \mathbf{M_2}) + c_{lock} \cdot (agr(\mathbf{M_1}, \mathbf{M_2}) - dis(\mathbf{M_1}, \mathbf{M_2}))$$

The constant $c_{lock} \geq 0$ allows to trade convergence speed with the amount of necessary overlap between partitions and the corresponding regions in the registration process. If c_{lock} is zero then the algorithm only registers partitions that have a large amount of overlap with ground truth. If c_{lock} is increased, then smaller and smaller amounts of overlap are necessary but the computation time increases. As discussed in detail in (Birk and Carpin, 2006), the choice of c_{lock} is not very critical, it can even be made adaptive within the algorithm. Here, a constant value of $c_{lock} = 1$ is used, which in general was found to work well.

Many other possible choices for Ψ_1 exist, for example computing some form of correlation like least mean squared Euclidean distances, or more specialized functions like the Hausdorff distance (Rucklidge, 1997; Huttenlocher and Rucklidge, 1993). But D_ψ has several interesting properties. First of all, it can be very efficiently computed, namely in linear time. Second, it provides meaningful gradients with respect to rotation and translation. These gradients are useful in the search process with which the registration is carried out. An Adaptive Random Walk (Carpin and

Pillonetto, 2003b,a, 2005) is used here for minimizing $\Psi_1()$ by searching over the space of possible transformations.

Finally, there is the need for a concrete implementation of the masking step. The same similarity metric as in the registration is used here, i.e., $\Psi_2 = \Psi_1$. The metric is simply embedded in a region-growing process that determines the largest partition with highest similarity to the reference. This is followed by a removal of this region from the current map, which is then used for a new registration followed by an other masking operation, and so on.

4 Experiments and Results

Two sets of maps are used for the following experiments. Both sets consist of a reference map and several versions of it with varying amounts of brokenness due to bump noise. The first set of maps (figure 2) is generated with USARsim (USARsim, 2006), a high fidelity robot simulator (Carpin et al, 2006). USARsim is based on the Unreal Game engine including a physics and 3D visualization engine. It also features realistic robot components models with realistic noise on the sensor data. The robot-model used to generate the maps is based on the Jacobs rescue robots (Birk et al, 2006). The environment is a detailed model of the R1 research building at Jacobs University.

The second set of maps (figure 2) is based on real world data from the Robotics Data Set Repository (Radish) (Howard and Roy, 2003). Concretely, the maps are based on the "datasetap_hill_07b", which contains the raw sensor data from four robots. For the experiments, the maps are generated from the raw data of the 3rd robot in the team; this robot simply explored most of the environment. The data is processed with a state-of-the-art SLAM algorithm (Grisetti et al, 2005) and varying degrees of bump noise.

An undisturbed map $\mathbf{R}$ is used in both sets as reference. The levels of bump noise vary within the sets, i.e., different degrees of brokenness ranging from $n_{BN} = 1$ to $n_{BN} = 5$ for set 1 and from $n_{BN} = 1$ to $n_{BN} = 3$ for set 2 are investigated. Each map in a set is compared to the reference using the algorithm from section 3.2. For completeness, the reference $\mathbf{R}$ is also compared to itself; obviously the result should be a brokenness of degree 0. This is also confirmed in the experiments.

As mentioned in section 2, it can be of interest to also consider the size of the partitions as well as the amount of transformations needed to register a partition properly with ground truth. The general algorithm for computing brokenness introduced in section 3.1 - and hence also the specific instance used here - provides the partitions and the transformations as a fringe benefit. So, it comes for free to use this information.

A straightforward metric $\|L\|$ for assessing the size of a partition L is the percentage of its cells with respect to the size of the whole map, i.e., $\|L\| = \#L/\#L^M$. For measuring the amount of each transformation error, it makes sense to concentrate on the angle in the transformations between partitions as the brokenness is almost exclusively caused by rotational errors. One metric $\|\mathcal{T}\|$ for one such transformation error is hence $\|\mathcal{T}_{\theta,d}\| = |(\theta - \pi)|/\pi$. The sum of all transformation errors $\|\mathcal{T}_G\|$ can serve as a more fine-grained global measure for brokenness in addition to the degree of brokenness.

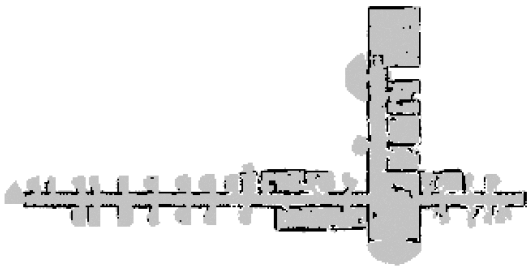
The results of the experiments are shown in tables 1 and 2. The most important fact is that in all cases the correct degree of brokenness n_{BN} is properly determined. Furthermore, the global measure $\|\mathcal{T}_G\|$ provides more detailed information about the brokenness. It can, for example, be used for rankings of the maps. The values of the relative partition sizes $\|L\|$ and the transformation metrics $\|\mathcal{T}\|$ for the individual partitions are listed for the sake of completeness in tables 1 and 2. The run times t are based on computations on a Intel Core-2 Duo 1.8 GHz processor under Linux. As can be expected, the run times increase roughly linearly with the degree of brokenness as each partition requires a full registration step, which is the computationally most expensive part in the overall algorithm.

The two datasets in the above experiments are chosen such that the area of each new partition L_k is monotonically decreasing with increasing brokenness $n_{BN} = k$ for each new map M_k. This is only done for the sake of presentation, namely to ensure the same order of the partitions in each map M_k in the tables 1 and 2. Figure 4 shows that the algorithm also works as expected if this is not the case, i.e., if the order of the partitions changes for increasing brokenness. In this final experiment, the brokenness of the two maps $\mathbf{M_3^{1'}}$ and $\mathbf{M_4^{1'}}$ is determined with map R_1 as reference (figure 2(a)). Both maps $\mathbf{M_3^{1'}}$ and $\mathbf{M_4^{1'}}$ contain an identical region, which is indicated by a dashed box in in the subfigures 4(a) and 4(b). This region has different relative sizes in the two maps. It is hence once correctly identified as partition L_2 in $\mathbf{M_3^{1'}}$, i.e., as the 3rd largest partition in this case, and once correctly as L_1 in $\mathbf{M_4^{1'}}$, i.e., as the 2nd largest partition.

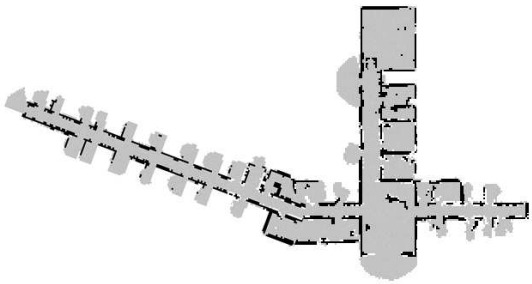
5 Conclusions

In this article, contributions to assessing structural errors in grid maps are made, i.e., errors that disturb the large scale spatial layout of a map with respect to a reference, typically ground truth.

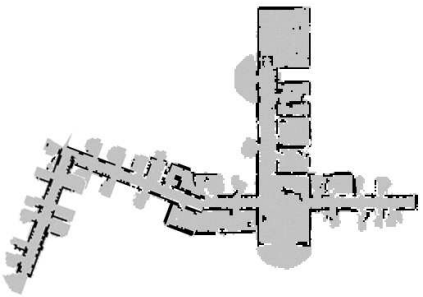
First and foremost, the concept of brokenness is introduced. It is an important - if not even the most important



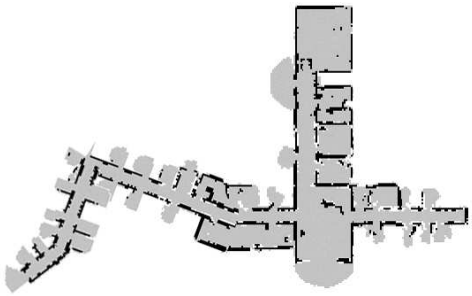
(a) undisturbed map $\mathbf{R^1}$



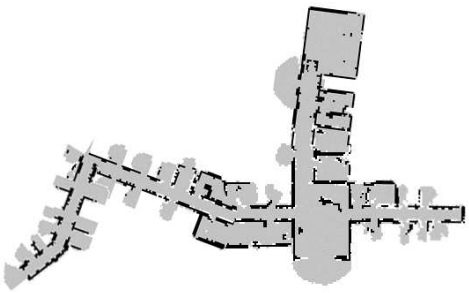
(b) $\mathbf{M_1^1}$, broken, degree $n = 1$



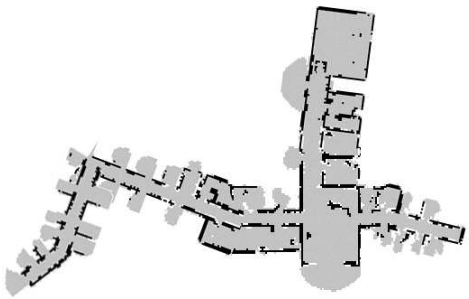
(c) $\mathbf{M_2^1}$, broken, degree $n = 2$



(d) $\mathbf{M_3^1}$, broken, degree $n = 3$

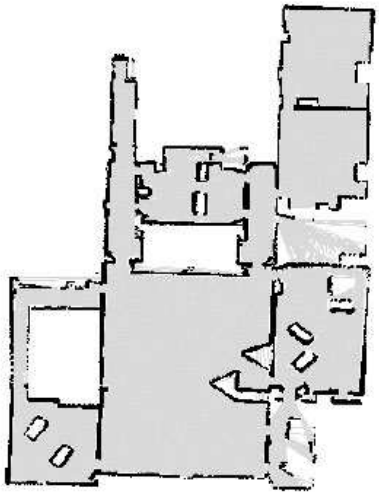


(e) $\mathbf{M_4^1}$, broken, degree $n = 4$

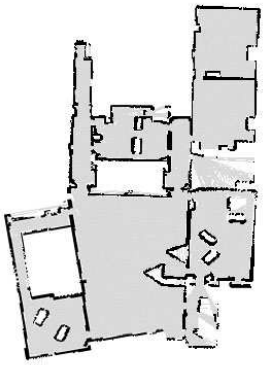


(f) $\mathbf{M_5^1}$, broken, degree $n = 5$

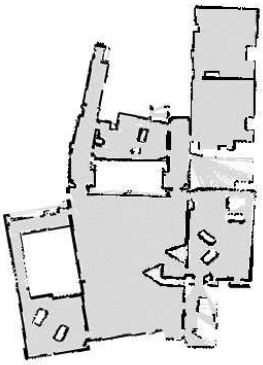
Figure 2: A first set of experiments is based on map data from simulations. The undisturbed map $\mathbf{R^1}$ is used as a reference. The five maps M_1^1 to M_5^1 show varying degrees of brokenness due to bump noise.



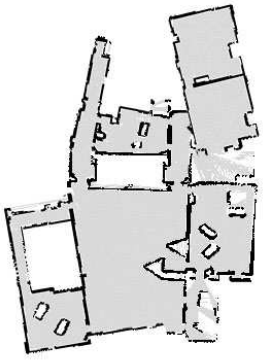
(a) undisturbed map $\mathbf{R^2}$



(b) $\mathbf{M_1^2}$, broken, degree $n = 1$



(c) $\mathbf{M_2^2}$, broken, degree $n = 2$



(d) $\mathbf{M_3^2}$, broken, degree $n = 3$

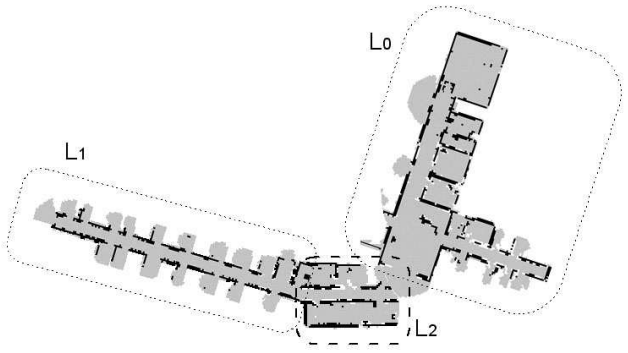
Figure 3: A second set of experiments is based on maps generated from real world data. The undisturbed map $\mathbf{R^2}$ is used as a reference. The three maps M_1^2 to M_3^2 show varying degrees of brokenness due to bump noise.

Table 1: The results of the experiments based on the maps from figure 2. The degree of brokenness n_{BN} is always correctly determined; the global transformation norm $\|\mathcal{T}_G\|$ gives in addition a more detailed numerical assessment of brokenness. The norms $\|\mathcal{T}_i\|$, respectively $\|Ł_i\|$ measure the transformations between the partitions and their sizes. The runtime t is in minutes (m) and seconds (ss).

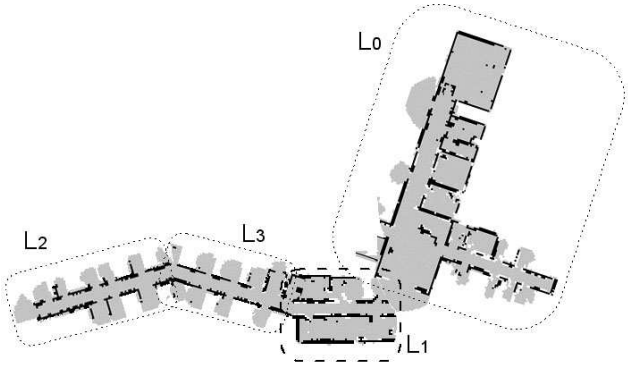
map	n_{BN}	$\|\mathcal{T}_G\|$	$\|\mathcal{T}_1\|$ (%)	$\|\mathcal{T}_2\|$ (%)	$\|\mathcal{T}_3\|$ (%)	$\|\mathcal{T}_4\|$ (%)	$\|\mathcal{T}_5\|$ (%)	$\|L_0\|$ (%)	$\|L_1\|$ (%)	$\|L_2\|$ (%)	$\|L_3\|$ (%)	$\|L_4\|$ (%)	$\|L_5\|$	t (m:ss)
$\mathbf{R^1}$	0	0	-	-	-	-	-	100	-	-	-	-	-	0:15
$\mathbf{M_1^1}$	1	22	22	-	-	-	-	63	37	-	-	-	-	1:16
$\mathbf{M_2^1}$	2	74	22	52	-	-	-	64	17	19	-	-	-	2:02
$\mathbf{M_3^1}$	3	82	22	52	8	-	-	64	17	13	6	-	-	4:30
$\mathbf{M_4^1}$	4	86	22	52	8	4	-	37	17	13	6	27	-	5:18
$\mathbf{M_5^1}$	5	98	22	52	8	4	12	34	17	13	6	27	3	6:14

Table 2: The results of the experiments based on the maps from figure 3. Again, the degree of brokenness n_{BN} is always correctly determined.

map	n_{BN}	$\|\mathcal{T}_G\|$	$\|\mathcal{T}_1\|$ (%)	$\|\mathcal{T}_2\|$ (%)	$\|\mathcal{T}_3\|$ (%)	$\|L_0\|$ (%)	$\|L_1\|$ (%)	$\|L_2\|$ (%)	$\|L_3\|$	t (m:ss)
$\mathbf{R^2}$	0	0	-	-	-	100	-	-	-	0:27
$\mathbf{M_1^2}$	1	9	9	-	-	82	18	-	-	1:07
$\mathbf{M_2^2}$	2	15	9	6	-	66	18	16	-	2:23
$\mathbf{M_3^2}$	3	27	9	6	12	40	18	16	26	3:50



(a) map $\mathbf{M}_3^{1'}$; $n_{BN} = 3$



(b) map $\mathbf{M}_4^{1'}$; $n_{BN} = 4$

Figure 4: The two broken maps shown above illustrate that the presented algorithm correctly determines the partitions L_i in the order of their sizes, i.e., in the order of the relative number of cells where predicate p - here occupancy - is fulfilled. The brokenness of the two maps $\mathbf{M}_3^{1'}$ and $\mathbf{M}_4^{1'}$ is computed with map R_1 as reference (figure 2(a)). The same area indicated by a dashed box is once correctly identified as partition L_2 and once correctly as partition L_1.

- form of structural error. Typical sources of brokenness in maps are bump noise in robot localization, and residual errors in SLAM. A formal way is used to derive the degree of brokenness of a map. The introduced concepts can in general be useful for discussions of map quality. Despite its formal basis, the degree of brokenness is based on intuitive notions that are also helpful to describe common properties of robot maps in informal ways.

Second, a general way to compute the degree of brokenness is introduced. It is based on recursively using registration and masking operations to determine the partitions of a map that are locally consistent but spatially transformed with respect to each other. Third, it is shown through experiments with a concrete implementation of a registration and a masking algorithm that the degree of brokenness can indeed be computed in an automatic way, hence allowing a quantitative assessment of the structural quality of a map.

References

Alliney S, Morandi C (1986) Digital image registration using projections. IEEE Trans Pattern Anal Mach Intell 8(2):222–233

Birk A (1996) Learning geometric concepts with an evolutionary algorithm. In: Proc. of The Fifth Annual Conference on Evolutionary Programming, The MIT Press, Cambridge

Birk A, Carpin S (2006) Merging occupancy grid maps from multiple robots. IEEE Proceedings, special issue on Multi-Robot Systems 94(7):1384–1397

Birk A, Pathak K, Schwertfeger S, Chonnaparamutt W (2006) The iub rugbot: an intelligent, rugged mobile robot for search and rescue operations. In: IEEE International Workshop on Safety, Security, and Rescue Robotics (SSRR), IEEE Press

Brown LG (1992) A survey of image registration techniques. ACM Comput Surv 24(4):325–376

Carpin S (2008a) Fast and accurate map merging for multi-robot systems. Autonomous Robots 25(3):305–316

Carpin S (2008b) Merging maps via hough transform. In: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pp 1878–1883, tY - CONF

Carpin S, Pillonetto G (2003a) Learning sample distribution for randomized robot motion planning: role of history size. In: Proceedings of the 3rd International Conference on Artificial Intelligence and Applications, ACTA press, pp 58–63

Carpin S, Pillonetto G (2003b) Robot motion planning using adaptive random walks. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp 3809–3814

Carpin S, Pillonetto G (2005) Motion planning using adaptive random walks. IEEE Transactions on Robotics 21(1):129–136

Carpin S, Birk A, Jucikas V (2005) On map merging. International Journal of Robotics and Autonomous Systems 53:1–14

Carpin S, Birk A, Lewis M, Jacoff A (2006) High fidelity tools for rescue robotics: results and perspectives. In: Noda I, Jacoff A, Bredenfeld A, Takahashi Y (eds) RoboCup 2005: Robot Soccer World Cup IX, Lecture Notes in Artificial Intelligence (LNAI), Springer

Chandran-Ramesh M, Newman P (2008) Assessing map quality using conditional random fields. In: Laugier C, Siegwart R (eds) Field and Service Robotics, Springer Tracts in Advanced Robotics, Springer

Dedeoglu G, Sukhatme G (2000) Landmark-based matching algorithm for cooperative mapping by autonomous robots. In: Distributed Autonomous Robotic Systems 4, Springer, pp 251–260

Dorai C, Wang G, Jain AK, Mercer C (Jan., 1998) Registration and integration of multiple object views for 3d model construction. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(1):83–89

Elfes A (1989) Using occupancy grids for mobile robot perception and navigation. Computer 22(6):46–57, tY - JOUR

Fenwick J, Newman P, Leonard J (2002) Cooperative concurrent mapping and localization. In: Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA, IEEE Computer Society Press

Frese U (2006) A discussion of simultaneous localization and mapping. Autonomous Robots 20:25–42

Gool LV, Moons T, Ungureanu D (1996) Affine/photometric invariants for planar intensity patterns. In: Proceedings of European Conference on Computer Vision

Grisetti G, Stachniss C, Burgard W (2005) Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA

Gutmann JS, Burgard W, Fox D, Konolige K (1998) An experimental comparison of localization methods. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), vol 2, pp 736–743 vol.2, tY - CONF

H Moravec MB (1993) Learning sensor models for evidence grids. CMU Robotics Institute 1991 Annual Research Review pp 8–15

Howard A (2004) Multi-robot mapping using manifold representations. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp 4198–4203

Howard A, Roy N (2003) The Robotics Data Set Repository (Radish). http://radish.sourceforge.net/

Huttenlocher DP, Rucklidge WJ (1993) A multi-resolution technique for comparing images using the hausdorff distance. In: Proceedings Computer Vision and Pattern Recognition, New York, NY, pp 705–706

Jose-Luis Blanco JAFM Javier Gonzalez (2007) A new method for robust and efficient occupancy grid-map matching. Pattern Recognition and Image Analysis 4478:194–201

Ko J, Stewart B, Fox D, Konolige K, Limketkai B (2003) A practical, decision-theoretic approach to multi-robot mapping and exploration. In: Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Lee DC (2003) The Map-Building and Exploration Strategies of a Simple Sonar-Equipped Mobile Robot: An Experimental, Quantitative Evaluation. Distinguished Dissertations in Computer Science, Cambridge University Press

Leonard JJ, Durrant-Whyte HF (1992) Directed Sonar Sensing for Mobile Robot Navigation. Springer

Lowe DG (2004) Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2):91–110

Madhavan R, Fregene K, Parker L (2004) Terrain aided distributed heterogenous multirobot localization and mapping. Autonomous Robots 17:23–39

Mikolajczyk K, Schmid C (June, 2003) A performance evaluation of local descriptors. In: Proceedings of Computer Vision and Pattern Recognition

Moravec H, Elfes A (1985) High resolution maps from wide angle sonar. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp 116–121

Roy N, Dudek G (2001) Collaborative exploration and rendezvous: Algorithms, performance bounds and observations. Autonomous Robots 11

Rucklidge WJ (1997) Efficiently locating objects using the hausdorff distance. International Journal of Computer Vision 24(3):251–270

Scrapper C, Madhavan R, Balakirsky S (2007) Stable navigation solutions for robots in complex environments. In: IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR), pp 1–6, tY - CONF

Scrapper C, Madhavan R, Balakirsky S (2008) Performance analysis for stable mobile robot navigation solutions. In: Proceedings of SPIE, International Society for Optical Engineering

Stricker D (2001) Tracking with reference images: a real-time and markerless tracking solution for out-door augmented reality applications. In: Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage, ACM Press, pp 77–82

Thrun S (2001) A probabilistic online mapping algorithm for teams of mobile robots. International Journal of Robotics Research 20(5):335–363

Thrun S (2002) Robotic mapping: A survey. In: Lakemeyer G, Nebel B (eds) Exploring Artificial Intelligence in the New Millenium, Morgan Kaufmann

Thrun S, Burgard W, Fox D (2000) A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp 321–328

Thrun S, Burgard W, Fox D (2005) Probabilistic Robotics. MIT Press

USARsim (2006) Unified System for Automation and Robotics Simulator (USARsim). http://usarsim.sourceforge.net/

Varsadan I, Birk A, Pfingsthorn M (2009) Determining map quality through an image similarity metric. In: Iocchi L, Matsubara H, Weitzenfeld A, Zhou C (eds) RoboCup 2008: Robot WorldCup XII, Lecture Notes in Artificial Intelligence (LNAI), Springer

Williams S, Dissanayake G, Durrant-Whyte H (2002) Towards multi-vehicle simultaneous localisation and mapping. In: Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA, IEEE Computer Society Press

Wnuk K (2005) Dense 3d mapping with monocular vision

Yairi T (2004) Covisibility based map learning method for mobile robots. 8th Pacific Rim International Conference on Artificial Intelligence