2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Normal Distributions Transform Occupancy Maps: Application to Large-Scale Online 3D Mapping

Jari Saarinen[1], Henrik Andreasson[1], Todor Stoyanov[1], Juha Ala-Luhtala[2] and Achim J. Lilienthal[1]

Abstract— **Autonomous vehicles operating in real-world industrial environments have to overcome numerous challenges, chief among which is the creation and maintenance of consistent 3D world models. This paper proposes to address the challenges of online real-world mapping by building upon previous work on compact spatial representation and formulating a novel 3D mapping approach — the Normal Distributions Transform Occupancy Map (NDT-OM). The presented algorithm enables accurate real-time 3D mapping in large-scale dynamic environments employing a recursive update strategy. In addition, the proposed approach can seamlessly provide maps at multiple resolutions allowing for fast utilization in high-level functions such as localization or path planning. Compared to previous approaches that use the NDT representation, the proposed NDT-OM formulates an exact and efficient recursive update formulation and models the full occupancy of the map.**

I. Introduction

Industrial applications of robotics are rapidly shifting from pre-programmed manipulators to highly mobile, intelligent robotic service vehicles. In order to operate efficiently in complex, large scale and realistic industrial environments, autonomous systems have to overcome numerous challenges in perception, localization, planning and control. Creating and maintaining consistent 3D models of the environment is one of the particularly important tasks for autonomous systems, as it constitutes an enabling technology for high-level intelligent behaviors. As high-frame rate, dense 3D depth sensors are becoming more common, both in research and industrial robotics, faster and more efficient 3D mapping algorithms are necessary. In order to provide reliable real-time environment modeling, several requirements have to be fulfilled by the underlying 3D spatial representation.

The occupancy grid map [7] is a popular 2D spatial representation. The key principle is to use a probabilistic framework for representing and accumulating information about occupancy of the space. The 2D occupancy map has many of the desirable properties of spatial representations — it is compact, expressive, easy to update and maintain, as well as able to represent uncertainty, while accounting for sensor noise. The extension of occupancy maps into 3D, however, is not straight forward — high resolution cell decomposition causes the number of cells to grow cubically with the size of the environment, while low resolution results in a deterioration of accuracy. In order to meet the requirements for operating in realistic industrial scenarios, a 3D mapping algorithm needs to use a representation, which is able to accurately represent the environment. In addition, the representation should also be compact to enable mapping over extended periods of time without running out of memory.

In this paper we build upon a previously known spatial representation — the Normal Distributions Transform (NDT). NDT was originally introduced in the context of 2D laser scan registration [2]. The central idea is to represent the observed range points as a set of Gaussian probability distributions. Magnusson et al. [6] first introduced 3D NDT in the context of scan registration. This work was later extended by Stoyanov et al. [9] by introducing 3D NDT-to-NDT registration method, which was shown to outperform state of the art registration approaches in accuracy, speed and robustness. Additionally, Stoyanov et al. in [10] compared the accuracy of 3D-NDT representation to grid-based [15] and triangle mesh based [14] representations using extensive tests conducted in different environments. An important result of [10] was that 3D-NDT typically provides the same or an improved accuracy at lower resolutions. The corresponding compact representation is essential for enabling real-time mapping in large-scale environments with high-density sensors.

Up until now, the primary use of NDT has been in modeling a single or few scans. This work extends the NDT representation by formulating an exact and efficient recursive update of NDT maps from a sequence of measurements. We also add an occupancy update for NDT cells, introducing a novel hybrid 3D representation — the NDT Occupancy Map (NDT-OM). The major contribution of this paper is to formulate a set of tools that enable the use of NDT-OM for long-term mapping in large scale, realistic industrial environments. We show that the proposed NDT-OM fulfills the requirements for a good spatial representation and that it is capable of online, real-time 3D mapping of large scale dynamic environments.

The rest of this paper is organized as follows. Sec. 2 reviews and formalizes prior work on the 3D representations. Sec. 3 presents our approach, Sec. 4 describes the test setup and illustrates the results and Sec. 5 concludes the paper with discussion.

II. Related Work

A. 3D Representations

Several different approaches for 3D spatial modeling have been proposed and successfully used in robotic mapping systems. Elevation grid maps — a 2.5D parametrization of space, obtained by associating a height value to cells

[1] Center of Applied Autonomous Sensor Systems (AASS), Örebro University, Sweden

[2] Department of Mathematics, Tampere University of Technology, Finland

978-1-4673-5643-5/13/$31.00 ©2013 IEEE

organized in a 2D grid, have been used for outdoor robot navigation since the early years of robotics research (e.g. Bares et al. [1]). Several authors have proposed to extend the elevation grid approach by modeling space using *Gaussian Processes* (GP). The available range sensor data can be used to learn the hyper-parameters of a GP, which can then be used to perform regression for any point in 2D space and obtain an interpolated height value, resulting in a continuous spatial model (Lang et al. [5]). Triebel et al. [13] propose the Multi-Level Surface (MLS) map as an extension to elevation grids which allows for multiple height values to be stored per cell. Triangle meshes are another method for spatial representation popular in the computer graphics community. In order to reconstruct a mesh representation of the environment from range data, special care has to be taken in filtering and handling of uncertainty (Wiemann et al. [14]). Finally, occupancy grid mapping is one of the predominant modeling techniques for 2D and 3D environments in robotics applications. The modeling approach, originally introduced by Moravec and Elfes [7], represents the environment by partitioning space into a regular grid and updating the probability of occupancy for each cell. Occupancy grid models both free and occupied space and through the sequential update it is capable of adapting to dynamic changes in the environment. It is the only of the above mentioned approaches that can do that and thus is the only one suited for mapping in dynamic environments. In this paper we will use Octomap [15], an implementation of 3D occupancy grid for a comparison.

B. Normal Distributions Transform (NDT)

The Normal Distribution Transform representation was first introduced by Biber and Strasser [2] for 2D scan matching. NDT is a grid based representation, much like the well known occupancy grid map [7], but capable of obtaining similar accuracy while using a much larger cell size [10]. The basic idea is to first accumulate sensor measurements into grid cells and then use them to compute a sample mean and a covariance for each cell:

$$\mu_i = \frac{1}{n}\sum_{k=1}^{n} x_k,\; P_i = \frac{1}{n-1}\sum_{k=1}^{n}(x_k-\mu)(x_k-\mu)^t \tag{1}$$

where $\{x_k\}_{k=1}^{n}$ is the set of n points that fall within the boundary of the considered cell. Thus, the NDT map is a set of Normal Distributions that describe the probability of a point being measured at a particular physical location.

In the context of updating a 3D NDT map Takeuchi and Tsubouchi [12] propose an approach for adding points into the NDT cells, without the need for storing the points. According to Chan et al. [4], [3] this approach is prone to numerical errors and “should almost never be used”. Moreover, the proposed approach requires storing intermediate sum values for mean and covariance, and thus requires either additional computations or memory to access normal distribution parameters compared to our approach. Furthermore, the approach of Takeuchi and Tsubouchi [12] does not implement an occupancy update.

Yguel and Aycard [16] propose a method for updating the distributions in each cell using error-refinement (ER). They also propose to use an occupancy update method depending on whether a cell was observed empty or occupied. The proposed approach[16], however, only tracks the occupancy probability of cells containing a Gaussian component, thus disregarding modeling of free space. In the subsequent sections we propose an approach that also models free space and demonstrate that the distribution update method proposed in this work is more accurate than the error-refinement based approach introduced in [16].

III. NDT Occupancy Maps (NDT-OM)

NDT Occupancy Map (NDT-OM) is an extended NDT map, which enables recursive updates of sequential measurements and models the occupancy of the cell. Formally, a cell c_i in NDT-OM is represented with parameters $c_i = \{\mu_i,\, P_i,\, N_i,\, p(m_i|z_{1:t})\}$, where μ_i and P_i, are the parameters of the estimated Gaussian component, N_i is the number of points used in estimation of normal distribution parameters so far, and $p(m_i|z_{1:t})$ is the probability of the cell being occupied. The minimum amount of parameters required to store by an NDT-OM cell is 11 (mean, upper diagonal of covariance, number of points and occupancy probability). Throughout this paper we will assume that an NDT-OM is composed of even sized cells, within a regular grid of a given resolution.

A. Online update of NDT-OM

We will now investigate how a single cell in the NDT grid can be updated with new observations in an efficient manner. The objective is to update the mean and covariance over the sequence of observations. Direct application of Eq. (1) to compute mean and covariance requires to store and process all points. This is not a feasible solution for long-term applications, since both memory requirement and computation time grow rapidly as new measurements are added. We will therefore first derive the equations for recursive update of mean and covariance. The approach is a multivariate extension Chan et al. [3].

Consider two sets of measurements $\{x_i\}_{i=1}^{m}$ and $\{y_i\}_{i=1}^{n}$. We want to compute the mean and covariance of the combined set of measurements. The recursive formula for the mean follows immediately the mean estimate in Eq. 1:

$$\mu_{x\oplus y} = \frac{1}{m+n}\left(\sum_{i=1}^{m} x_i + \sum_{i=1}^{n} y_i\right) = \frac{1}{m+n}\left(m\mu_x + n\mu_y\right). \tag{2}$$

Let $S_x = \sum_{i=1}^{m}(x_i-\mu_x)(x_i-\mu_x)^T$ and $S_y = \sum_{i=1}^{n}(y_i - \mu_y)(y_i-\mu_y)^T$. Then, the covariance estimate for the combined set is

$$P_{x\oplus y} = \frac{1}{n+m-1} S_{x\oplus y}, \tag{3}$$

where

$$S_{x\oplus y} = \sum_{i=1}^{m}(x_i - \mu_{x\oplus y})(x_i - \mu_{x\oplus y})^T + \\ \sum_{i=1}^{n}(y_i - \mu_{x\oplus y})(y_i - \mu_{x\oplus y})^T. \tag{4}$$

Here we first note that

$$\begin{aligned}
&\sum_{i=1}^{m}(x_i - \mu_{x\oplus y})(x_i - \mu_{x\oplus y})^T \\
&= \sum_{i=1}^{m}(x_i - \mu_x + \mu_x - \mu_{x\oplus y})(x_i - \mu_x + \mu_x - \mu_{x\oplus y})^T \\
&= \sum_{i=1}^{m}(x_i - \mu_x)(x_i - \mu_x)^T + m(\mu_x - \mu_{x\oplus y})(\mu_x - \mu_{x\oplus y})^T \\
&= S_x + m(\mu_x - \mu_{x\oplus y})(\mu_x - \mu_{x\oplus y})^T. \qquad (5)
\end{aligned}$$

The second step follows since the terms

$$\sum_{i=1}^{m}(x_i - \mu_x)(\mu_x - \mu_{x\oplus y})^T \quad \text{and} \quad \sum_{i=1}^{m}(\mu_x - \mu_{x\oplus y})(x_i - \mu_x)^T$$

are equal to zero. Similarly for the summation involving y_i terms:

$$\begin{aligned}
&\sum_{i=1}^{n}(y_i - \mu_{x,y})(y_i - \mu_{x\oplus y})^T \qquad (6) \\
&\quad = S_y + n(\mu_y - \mu_{x\oplus y})(\mu_y - \mu_{x\oplus y})^T. \qquad (7)
\end{aligned}$$

Eq. (4) together with Eqs. (2), (5) and (6) could already be used for recursive updating of the covariance. However, the formula can be further simplified by noting that

$$\begin{aligned}
&m(\mu_x - \mu_{x\oplus y})(\mu_x - \mu_{x\oplus y})^T + n(\mu_y - \mu_{x\oplus y})(\mu_y - \mu_{x\oplus y})^T \\
&= m\mu_x\mu_x^T + n\mu_y\mu_y^T - \mu_{x\oplus y}(m\mu_x + n\mu_y)^T - \\
&\quad (m\mu_x + n\mu_y)\mu_{x\oplus y}^T + (m+n)\mu_{x\oplus y}\mu_{x\oplus y}^T \qquad (8) \\
&= m\mu_x\mu_x^T + n\mu_y\mu_y^T - \frac{1}{m+n}(m\mu_x + n\mu_y)(m\mu_x + n\mu_y)^T \\
&= \left(m - \frac{m^2}{m+n}\right)\mu_x\mu_x^T + \left(n - \frac{n^2}{m+n}\right)\mu_y\mu_y^T - \\
&\quad \frac{mn}{m+n}\mu_x\mu_y^T - \frac{mn}{m+n}\mu_y\mu_x^T \qquad (9) \\
&= \frac{mn}{m+n}(\mu_x - \mu_y)(\mu_x - \mu_y)^T, \qquad (10)
\end{aligned}$$

where the second step follows by inserting Eq. (2) for $\mu_{x\oplus y}$. Combining Eqs. (4)-(10) we have

$$S_{x\oplus y} = S_x + S_y + \frac{mn}{m+n}(\mu_x - \mu_y)(\mu_x - \mu_y)^T. \tag{11}$$

Equations (2) and (11) together with Eq. (3) provides exact sample mean and covariance of the combined sample sets. For creating NDT maps, this means that we do not need to store point clouds after updating the cells. Essentially, the memory requirements do now scale only with the size of the environment and not with the operation time. Additionally, a direct observation is that the mean and covariance can be updated by knowing the number of points used in estimation so far. This is convenient from implementation point of view since one only needs to store the number of points as an extra parameter for each cell. In the remainder of this paper the cell update method described above will be referred to as the Recursive Covariance Update (RCU) method.

B. NDT-OM as multi-resolution maps

A notable property of the RCU is that it can be used to efficiently create lower resolution maps from an existing high resolution map. Since Eqs for (2) and (11) hold for two arbitrary sized sample sets, one simply needs to combine all normal distributions that fall into the lower resolution cell with RCU in order to create a lower resolution map. It is thus sufficient to maintain one global NDT-OM with a high resolution, which can be transformed into suitable lower resolutions when needed (see Fig. 4d) e.g. for registration or planning purposes.

C. Occupancy update for NDT-OM

Up until now, the NDT update assumes a static environment. However, in dynamic environment some cells for which a Gaussian was estimated at some point in time will become empty at a later time. This type of state-change is not handled by the NDT update; a Gaussian will remain unchanged in the cell even if the measurements repeatedly indicate that the cell is no longer occupied. We propose to add an occupancy update step for NDT maps. This introduces two new steps to the NDT update: 1) instead of just adding a measured point into the map, ray-tracing is applied to collect evidence about empty cells; and 2) a probabilistic occupancy estimate is maintained for all cells.

The ray-tracing step evaluates all cells that lie between the sensor origin and the measured point. A counter is increased for each cell along the ray that does not contain the end point. The cell occupancy update uses the log-odds formulation for updating occupancy grid maps [8]. An important difference to standard occupancy mapping is due to the fact that NDT models a spatial structure *within* a grid cell, while standard occupancy mapping assumes the cells to be homogeneous. This means that cells, which were "hit" by some measurements will be considered occupied even if rays repeatedly go through these cells. Since, loosely speaking, the Gaussian distributions do not cover the whole cell, single measurements carry only little evidence about cells being empty. Accordingly, the occupancy belief in NDT-OM is updated more conservatively after a cell has been observed as empty than in standard occupancy mapping. However, assuming that over time we observe cells uniformly, the log-odds value $l(m_k|z_{1:t})$ for the cell m_k, given measurements z_t becomes

$$l(m_k|z_{1:t}) = l(m_k|z_{1:t-1}) + \begin{cases} N_p log(\frac{p_o}{1-p_o}),\ if\ N_p > 0 \\ N_e log(\frac{p_e}{1-p_e}),\ if\ N_p = 0 \end{cases}, \tag{12}$$

where N_p is the number of points in z_t that fall into the cell, N_e is the number of times the cell is observed empty while adding z_t, p_o is the probability of occupancy when measured occupied, and p_e is the probability of occupancy when observed free. In our implementation we use heuristic values of $p_o = 0.6$ and $p_e = 0.49$.

IV. Tests and Results

A. Map quality test for recursive NDT update

Map quality test was carried out to experimentally validate the recursive update of NDT as introduced in Section III-A. The selected map quality metric is based on L2-distance for normal distributions [9]. This metric provides a similarity between two NDT maps. Furthermore, it has been successfully applied to scan-to-scan registration [9], and thus also provides information about the suitability of a map for localization. We compare Recursive NDT (updated using RCU (Section III-A), without occupancy update), NDT-OM and the Error refinement (ER) approach presented in [16].

For testing we used a publicly available data set (rgbd_dataset_freiburg1_rpy-2hz-with-pointclouds.bag) from the TUM RGB-D SLAM data sets [11]. The data set contains 47 point clouds collected by a hand-held Kinect depth camera as well as synchronized ground truth sensor poses from a motion capture system. This set was selected because it provides an example of a situation where the same static area is observed repeatedly. The set is also sufficiently limited in size so that we can compute a batch-map using all the points according to Eq. (1) without memory overflows. Please note, that we will only compare the accuracy of NDT representations in this paper. A comparison against alternative representations including NDT, occupancy grid and triangle mesh can be found in [10].

1) Comparison against the batch map: This evaluation measures the consistency of the map when compared against the batch map. The evaluation was done by incrementally inserting one point cloud at a time to each map. After each point cloud insertion also a batch map is computed using the accumulated measurements so far. After each insertion the maps are compared against the batch map, by calculating the mean L2-distance. The mean L2-distance is obtained by the sum of L2-distances for all cells that contain a Gaussian in the test map, and dividing the result with number of Gaussians in the batch map. The maximum correspondence using the mean L2-distance is obtained with value one.

The result is illustrated in Fig. 1. We first see that the recursive update of NDT accurately reproduces the normal distributions of the batch map. A very small difference, hardly visible in Fig. 1, is because a Gaussian is not computed if the number of points is less than 3 in a cell. This happens more often in individual update steps than in the batch map. The ER approach is clearly the least consistent when compared against the batch map. NDT-OM differs slightly from the batch map. This is due to sensor noise, which can cause Gaussian distributions being estimated for cells that are not occupied. However, these erroneous distributions vanish due to the occupancy update step over the sequence of measurements and this results into a lower mean L2-distance (simply because there is a lower number of elements in the test map). The difference between NDT-OM and the batch map is indeed desirable.

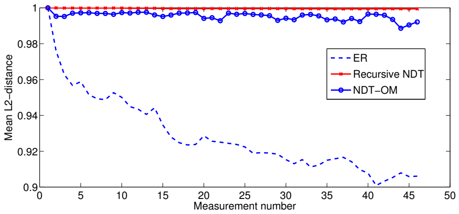

Fig. 1. ER, Recursive NDT and NDT-OM compared against the batch map. The higher the mean L2 distance the better correspondence.

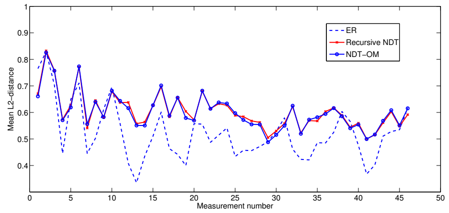

Fig. 2. Representation self-test. Individual point clouds are compared against the ER, Recursive NDT and NDT-OM representations constructed using all measurements. Higher mean L2-distance means better similarity between the representation and the measurement.

2) Representation self-test: This test measures the capability to represent observations, when revisiting a mapped region. The test was conducted by first creating a map using the whole data set for each approach. Then, the individual point clouds in the data set were tested against the map independently. Since the environment is static the map representation should explain all measurements well. Again, the mean L2-distance was used as metric. This time the comparison is done against a local NDT map, which is constructed from a single point cloud. The result is illustrated in Fig. 2. The result shows, that both Recursive NDT and NDT-OM perform equally well outperforming ER. We conclude that a map built using ER does not represent the measurements as well as maps obtained with NDT-OM.

B. Mapping in large-scale dynamic environment

1) Test setup: The target environment for the large-scale tests is a milk production plant, where Laser Guided Vehicles (LGVs) transfer the milk cages from production to warehouse. The size of the target environment is approximately 150×150 meters. A part of this environment is illustrated in Fig. 4e. The data was collected on top of an LGV in production use and consists of Velodyne HDL-32 3D lidar measurements collected at 10Hz together with localization data from an independent navigation system at 15Hz. The HDL-32 sensor measures up to 70m and produces up to 700,000 measurement points per second. In practice this number is lower, and was found to vary from 200,000 to 500,000 in our tests due to the partially blocked view by the LGV and distances over the measurement range. The test data used cover 10min of LGV operation, during which the vehicle traveled 255m. We use the pose from the independent navigation system and thus address the problem of mapping with known poses. All the tests are performed using a laptop computer, with Intel Core i7 CPU Q 720 @ 1.60GHz on a single core.

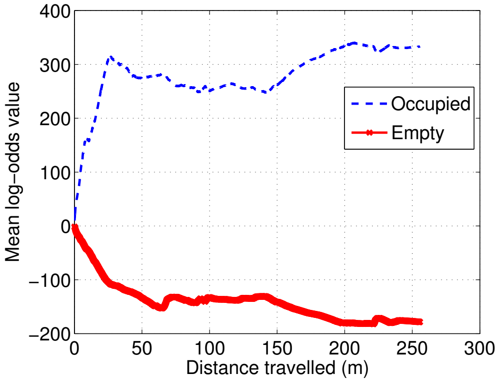

Fig. 3. Evolution of mean log-odd values for empty and occupied cells during mapping

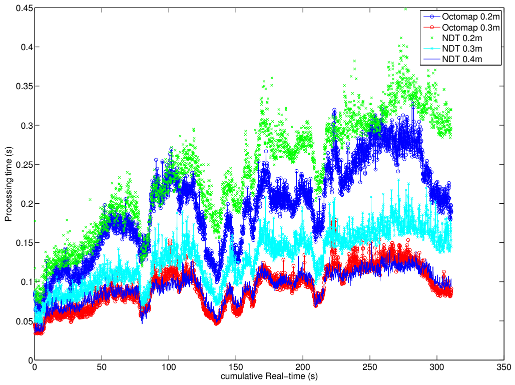

Fig. 5. Time comparison between NDT-OM and Octomap with different resolutions.

2) Mapping in a large-scale dynamic environment: The mapping test was performed by incrementally adding scans every 0.05m of distance traveled using the pose given by the vehicle. Fig. 3 illustrates the evolution of the mean log-odds values for both occupied cells and empty cells while mapping. The graph shows that the confidence over the static cells grows while the environment is re-observed. The curves are decreasing/increasing similar to what would be expected for occupancy maps, however, not monotonically, which is because new regions are discovered during mapping.

Figure 4a shows a mapping result obtained with NDT-OM. The ellipsoids in the figure represent the 1-σ contour of the estimated normal distributions. Figure 4b illustrates the result without occupancy update. The highlighted parts in 4b are obtained by finding "dynamic" cells that are empty in NDT-OM (see Figure 4c), but represented with a Gaussian in the map without occupancy update. In total the number of conflicting cells was 15281 (approx. 25% of the NDT-OM cells). This demonstrates the need for the occupancy update extension of NDT proposed in this paper in dynamic environments. NDT-OM represents the parts that were static during mapping well, but without occupancy update the map would quickly get inconsistent.

Figure 4d shows an example of a low resolution map created from the map illustrated in Figure 4a. In this case the high-resolution map had 0.3m resolution with 61728 normal distributions and the low-resolution map had 0.9m resolution with 11359 Gaussians. Comparing Figure 4a and Figure 4d, we can observe that the finer details (e.g. on the right side) are not very visible any more in low-resolution map. However, the clear surfaces such as the floor and the wall on the left are still accurately represented.

3) Performance comparison: The performance comparison was conducted using different NDT-OM resolutions and with the factory data test. As baseline we used Octomap [15], a popular implementation of 3D occupancy grid mapping. Velodyne HDL-32 produces measurements at 10Hz, so in order to reach real-time performance the measurements must be processed faster than 100ms/measurement. Figure 5 summarizes our tests. In order to reach real-time a resolution of 0.4m is required for an NDT-OM in our setup, whereas Octomap requires 0.3m. Comparing NDT-OM and Octomap at equal resolutions, we can conclude that NDT-OM is little slower, which was expected because it requires the update of normal distributions in addition to the occupancy update. However, NDT is more expressive at the same resolution. In [10] it was shown that in some environments an NDT with 0.8m resolution (mean update time 44ms) is equivalent to an occupancy grid with 0.1m resolution (mean update time 818ms). In fact, 0.3m resolution for an occupancy map is considered very coarse, while the same resolution can be considered high for an NDT. Thus, we can conclude that NDT-OM is capable of real-time 3D mapping with representative accuracy.

V. Conclusion and Discussion

In this paper we proposed extensions for the Normal Distributions Transform based approaches to account for occupancy updates and continuous online map building. A novel approach, Normal Distributions Transform Occupancy Map (NDT-OM) is presented and demonstrated to be a viable option for real-time 3D mapping in large-scale, dynamic environments. We showed that the NDT-OM can be efficiently updated online with new measurements without deterioration in accuracy, compared to a direct NDT estimation approach. The proposed approach guarantees that the memory requirements of the model depend on the size of the environment mapped and not on the length of the traveled trajectory or amount of 3D range data acquired. The cell update approach also natively enables the use of NDT-OM as a multi-resolution map, which is an important feature for global operations, such as planning and localization.

The main focus of this paper was to formulate NDT-OM and demonstrate its use for mapping in large-scale dynamic environments. We have not yet discussed in detail all of the implications and future applications of the proposed approach. However there is a wide range of applications

The version of Octomap used for comparison was 1.4.3 from ros-fuerte-octomap package.

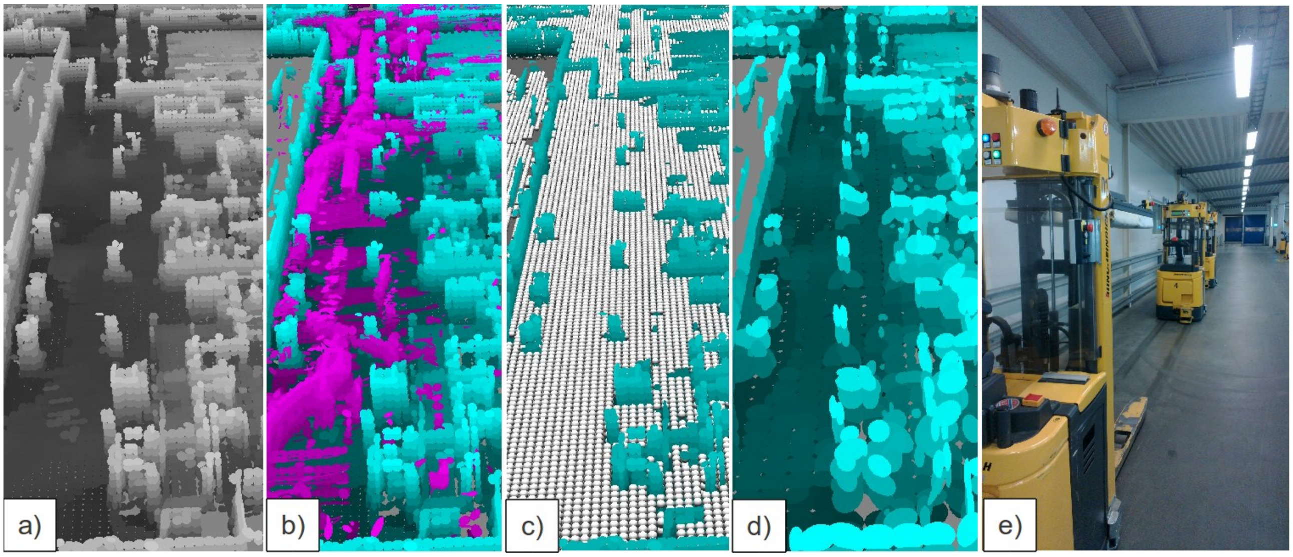

Fig. 4. Mapping result in milk production plant. The figures show the corridor in production are and part of the warehouse as illustrated in e). Ellipsoids represent the 1-σ contour of the estimated normal distributions a) NDT-OM, b) Recursive NDT, without occupancy update, the conflicting cells (with respect to NDT-OM) are highlighted, c) empty cells estimated by NDT-OM, illustrated as white spheres (plotted up to 1.5m height for clarity), d) a low-resolution map created from a), and e) a picture of the area and vehicles.

for which accurate real-time modeling of dynamic 3D environments is needed. In this paper we had an external localization system that provided an accurate position; this is an assumption that in general cannot be made. Recent work on NDT registration [9] has shown that NDT to NDT registration is a fast, accurate and robust solution. Combining NDT-OM with NDT to NDT registration allows model – to – observation registration, which can be used as an accurate localization/SLAM system.

All the results of this paper are integrated as part of the oru-ros-pkg open source package.

References

[1] J. Bares, M. Hebert, T. Kanade, E. Krotkov, T. Mitchell, R. Simmons, and W. Whittaker. Ambler: an autonomous rover for planetary exploration. *Computer*, 22(6):18 –26, june 1989.

[2] P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS 2003)*, pages 2743–2748, 2003.

[3] T.F. Chan, G.H. Golub, and R.J. LeVeque. *Updating formulae and a pairwise algorithm for computing sample variances*. Department of Computer Science, Stanford University, 1979.

[4] T.F. Chan, G.H. Golub, and R.J. LeVeque. Algorithms for computing the sample variance: Analysis and recommendations. *American Statistician*, pages 242–247, 1983.

[5] T. Lang, C. Plagemann, and W. Burgard. Adaptive Non-Stationary Kernel Regression for Terrain Modeling. In *Robotics: Science and Systems (RSS)*, Atlanta, Georgia, USA, June 2007.

[6] M. Magnusson, A. Lilienthal, and T. Duckett. Scan registration for autonomous mining vehicles using 3D-NDT. *Journal of Field Robotics*, 24(10):803–827, 2007.

[7] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, volume 2, pages 116 – 121, mar 1985.

[8] H.P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI magazine*, 9(2):61, 1988.

[9] Todor Stoyanov, Martin Magnusson, Henrik Andreasson, and Achim J Lilienthal. Fast and accurate scan registration through minimization of the distance between compact 3d ndt representations. *The International Journal of Robotics Research*, 31(12):1377–1393, 2012.

[10] Todor Stoyanov, Martin Magnusson, and Achim J. Lilienthal. Comparative Evaluation of the Consistency of Three-Dimensional Spatial Representations used in Autonomous Robot Navigation. *Journal of Field Robotics*, 2013. to appear.

[11] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[12] E. Takeuchi and T. Tsubouchi. A 3-D Scan Matching using Improved 3-D Normal Distributions Transform for Mobile Robotic Mapping. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3068 –3073, oct. 2006.

[13] R. Triebel, P. Pfaff, and W. Burgard. Multi-level Surface Maps for Outdoor Terrain Mapping and Loop Closing. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. IROS 2006.*, pages 2276 – 2282, 2006.

[14] T. Wiemann, A. Nüchter, K. Lingemann, S. Stiene, and J. Hertzberg. Automatic construction of polygonal maps from point cloud data. In *Proc. 8th IEEE Intl. Workshop on Safety, Security, and Rescue Robotics (SSRR-2010), Bremen, Germany*, 2010.

[15] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, Anchorage, AK, USA, May 2010. Software available at octomap.sf.net.

[16] M. Yguel and O. Aycard. 3D Mapping of Outdoor Environment Using Clustering Techniques. In *Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on*, pages 403 –408, nov. 2011.

Software available at http://code.google.com/p/oru-ros-pkg/