Auton Robot (2011) 30: 307–331
DOI 10.1007/s10514-011-9221-8

Spectral registration of noisy sonar data for underwater 3D mapping

Heiko Bülow · Andreas Birk

Received: 29 April 2010 / Accepted: 24 January 2011 / Published online: 4 February 2011

Abstract 3D mapping is very challenging in the underwater domain, especially due to the lack of high resolution, low noise sensors. A new spectral registration method is presented that can determine the spatial 6 DOF transformation between pairs of very noisy 3D scans with only partial overlap. The approach is hence suited to cope with sonar as the predominant underwater sensor. The spectral registration method is based on Phase Only Matched Filtering (POMF) on non-trivially resampled spectra of the 3D data.

Two extensive sets of experiments are presented. First, evaluations with simulated data are done where the type and amount of noise can be controlled and the ground truth transformations between scans are known. Second, real world data from a Tritech Eclipse sonar is used. Concretely, 18 sonar scans of a large structure in form of a flood gate and a lock in the river Lesum in Bremen are used for 3D mapping. In doing so, the spectral registration method is compared to two other methods suited for noisy 3D registrations, namely Iterative Closest Point (ICP) and plane-based registration. It is shown that the spectral registration method performs very well in terms of the resulting 3D map as well as its runtimes.

Keywords 3D mapping · Underwater robotics · Spectral registration · Autonomous Underwater Vehicle (AUV) · Remotely Operated Vehicle (ROV)

H. Bülow · A. Birk (✉)
Dept. of Computer Science, Jacobs University Bremen, 28759 Bremen, Germany
e-mail: a.birk@jacobs-university.de
url: http://robotics.jacobs-university.de

H. Bülow
e-mail: h.buelow@jacobs-university.de

1 Introduction

Autonomous Underwater Vehicles (AUV) are so far mainly restricted to usage in open sea application where 2D maps are sufficient. But many marine application scenarios involve more complex environments where 3D information is essential. 2D mapping in these environments may be sufficient for aiding a remote operator or for most simple tasks (Ribas et al. 2006, 2007), but it is far from sufficient for any intelligent operations of AUVs. Also, proper 3D maps of the environment can obviously be helpful for the operators of Remotely Operated Vehicles (ROV), especially in complex inspection, construction, or maintenance missions that involve non-trivial structures as for example in oil- and gas-production and transport, underwater archeology, or harbor applications.

For mobile robotics in general, there has been a strong interest in generating 3D environment models (Pathak et al. 2010; Nuechter et al. 2006; Howard et al. 2004; Thrun et al. 2003; Hähnel et al. 2003; Davison and Kita 2001; Liu et al. 2001). But this work is limited to land robots as it is based on the availability of high resolution, low noise sensors in form of laser range finders (Wulf and Wagner 2003; Surmann et al. 2003; Wulf et al. 2004).

For underwater systems, one may argue that ground elevation as represented in classic bathymetric maps may be sufficient (Oskard et al. 1990; Newman and Durrant-Whyte 1998), which are nevertheless still quite challenging to generate (Majumder et al. 2001). Occasionally, bathymetric maps are even denoted as 3D maps (Negahdaripour and Madjidi 2003; Salvi et al. 2008; Madjidi and Nagahdaripour 2003; Roman and Singh 2006, 2007), which is inaccurate as bathymetric maps are actually 2.5D elevation maps, i.e., they can only represent a single continuous surface and not arbitrary 3D structures. One way out to remedy this problem

is to partition the environment into a collection of horizontal 2D slices (Lee et al. 2009). This may be suited to augment bathymetric maps, e.g., to handle overhanging cliffs at ocean ridges, but there would be an unlimited number of slices required to represent arbitrarily complex 3D structures.

Early work on real 3D Underwater Mapping can be found in Ende (2001) where an underwater cave is mapped. This work uses a ring of sonars on the robot and mainly concentrates on localizing the vehicle via an Inertial Navigation System (INS). Concretely, high end ring-laser gyros (RLG) for attitude sensing combined with accelerometers for translations are used. Despite the high cost of this solution, an INS will always experience quite some drift, especially in the double integration of the accelerometers, and lead to inaccurate position estimates that get worse and worse, i.e., the map quality deteriorates.

This can—at least partially—be remedied by using Simultaneous Localization and Mapping (SLAM). In Fairfield et al. (2007), there is again a complex 3D structure in form of an underwater cave mapped, namely the a flooded cenote (sinkhole) in Tamaulipas, Mexico. This work uses proper SLAM in form of a Rao-Blackwellized particle filter to generate an evidence grid representation. Nevertheless, an INS and a Doppler Velocity Log (DVL) are still required to aid the localization. Furthermore, the particle filter requires to maintain several hundred candidate maps, i.e., it is quite memory and compute intensive.

The benefits of SLAM are also demonstrated in Williams and Mahon (2004) where corals at the Great Barrier Reef are mapped. The vehicle's 2D path is estimated by 2D visual SLAM, which is used to compute a 2D mosaic as well as a rough 3D terrain model based on additional range information from a scanning sonar. This compares to the situation of bathymetric maps, i.e., not a proper 3D but a 2.5D model is generated as the vehicle is only localized in 2D and range information is supplemented at the estimated 2D poses.

An example for the generation of complex 3D models from underwater data is the recent work by Sedlazeck and Koeser (2009), where monocular video data and not motion sensor data is used. The work uses a structure from motion technique, i.e., the tracking and registration of features over several frames. The approach is used on deep-sea video footage from black smokers. But it is computationally expensive and hence used in an offline manner. Also, the monocular structure from motion methods are usually best suited for small areas, i.e., to generate 3D models of (complex) objects, and less for larger maps. Monocular vision is also used in the research presented in Nicosevici et al. (2009). An important feature of this work is that it uses sequential structure from motion, i.e., the 3D model is generated and updated whenever new sensor data becomes available. The results presented in Nicosevici et al. (2009) cover different scenarios including an underwater environment, which is quite challenging.

An approach to map larger underwater areas in 3D is presented in Pizarro et al. (2004). It also uses monocular video data, i.e., it can be considered as a variant of structure from motion but it uses in addition motion estimates, i.e., data from a compass, a depth sensor, and especially a Doppler velocity log (DVL). The approach uses a combination of techniques from computer vision, photogrammetry, and robotics, including visual features for determining correspondences across frames, bundle-adjustment for extracting 3D information, and navigation using quite precise motion sensors.

Results of 3D underwater mapping of a coral reef without any usage of inertial navigation sensors or similar are reported in Saez et al. (2006). The work uses stereo data, i.e., disparity images for range information, and visual features from the single monocular images for finding correspondences. This registration is embedded in a entropy minimization for SLAM. The registration method is quite fast, namely in the order of 150 to 200 msec, though global rectifications that are done every 100 frames take longer, namely up to 650 msec. This is both higher than the frame rate of the stereo camera, nevertheless it is close to be suited for online processing of the data.

All the above work has in common that either high end localization approaches are required or that visual data is employed, or both. But underwater motion sensors, like high quality Inertial Navigation Systems (INS) and Doppler Velocity Logs (DVL), are costly. And cameras can often not be used as there is very low visibility in many underwater environments, including important application scenarios like harbors or rivers. Also, vision is impaired in many scenarios, namely if there are moving objects like fish or an effect known as sea snow, which is generated from moving particles like plankton. The presence of many fish or sea snow obviously renders the tracking of supposedly static visual features extremely difficult.

An alternative to visual data is to use sonar for registration. There is for example already quite some work on 2D scan matching with sonar to generate 2D underwater maps (Pfingsthorn et al. 2010; Buelow et al. 2010; Mallios et al. 2009; Hernandez et al. 2009; Ribas et al. 2006, 2007). There has also been significant progress in 3D sensing and perception with sonar in the underwater domain in recent years (Murino and Trucco 2000; Lorenson and Kraus 2009). This includes multi-target tracking (Clark et al. 2005), the modeling of hydrothermal plumes (Santilli et al. 2004), the registration of sonar range views for scene reconstruction (Castellani et al. 2002, 2004; Murino et al. 2000), and object modeling and recognition (Palmese and Trucco 2008; Yu et al. 2007; Murino et al. 2007).

In this article, we address the problem of generating 3D underwater maps with sonar scans, i.e., with coarse 2.5D range data from forward looking devices. Concretely,

a spectral registration method is presented that can handle the very noisy data and can cope with large displacements between scans without motion estimates of the vehicle. Furthermore, the method is quite fast and has a fixed computation time, which is of interest to allow online processing with predictable update rates, e.g., for the control of AUV.

The rest of this article is structured as follows. In Sect. 2, the new spectral registration method is introduced. We present the method itself as well as its derivation from existing work on 2D spectral registration and 3D spectral processing, especially with respect to our intention to cope with small overlap and high amounts of noise. The discussion also includes the presentation of limits of the approach, especially in terms of the maximum simultaneous change between scans in yaw, pitch, and roll, which can be seen as a trade off to being able to handle small overlap and high amounts of noise. The performance of the spectral registration approach is first extensively evaluated in Sect. 3. Simulated data is used for this purpose as it has the advantage that the amount and the type of noise can be controlled and that the ground truth transformations between scans are known. In Sect. 4, experiments with real world data from a sonar, namely a Tritech Eclipse, are presented. Concretely, 18 sonar scans of a large structure in form of a flood gate and a lock in the river Lesum in Bremen are used as basis for 3D mapping. Each scan is a single, unprocessed sensor reading from the sonar device. The spectral registration method is compared to two other 3D registration methods, namely Iterative Closest Point (ICP) and plane-based registration. The comparison shows that the spectral registration method performs very well in terms of the resulting 3D map as well as its run-times. Section 5 concludes the article.

2 Spectral registration of 3D range data

2.1 Phase Only Matched Filtering (POMF)

The signal registration used here in all dimensions is based on Phase Only Matched Filtering (POMF) (Horner and Gianino 1984). This correlation approach makes use of the fact that two shifted signals having the same spectrum magnitude are carrying the shift information within their phase (1). This holds for N-dimensional signals and will be used for the rotational registration of descriptor data, as well as for the translation.

$$f(t-a) \bullet\!\!-\!\!\circ\, F(\omega)e^{i\omega a} \tag{1}$$

When both signals are periodically shifted the resulting inverse Fourier transformation of the phase difference of both spectra is actually an ideal Dirac pulse. This Dirac pulse indicates the underlying shift of both signals which have to be registered.

$$\delta(t-a) \bullet\!\!-\!\!\circ\, 1e^{i\omega a} \tag{2}$$

The resulting shifted Dirac pulse deteriorates with changing signal content of both signals. As long as the inverse transformation yields a clear detectable maximum this method can used for matching two signals.

2.2 Motivation of the algorithm

Compared to previous spectral registration algorithms (Makadia and Daniilidis 2003; Makadia et al. 2004, 2006; Kostelec and Rockmore 2003), the work presented here is fully based on Fourier transforms and yields peaks indicating a unique solution. The advantage is the full use of nearly all the 3D spectral information compared to the approaches from Lucchese et al. (2002), Keller et al. (2005, 2006) which use only small parts of the available spectral structures.

In a first step, the rotation will be determined. This is done by a resampling of points $\eta(\theta, \phi) \in \mathbb{S}^2$ on the unit sphere directly from the 3D spectrum. Here the structures on the unit sphere $\mathbb{S}^2$ are taken from different radii corresponding to different 3D frequencies and assembled to a 3D stack. The general advantage is that the process relies on phase matching of shifted structures, which yields a Dirac peak indicating a correct position even with large interfering structures. For the rotational orientation it exploits the fact that the information of 3D rotation is available within the magnitude of the 3D spectrum and is therefore decoupled from the overall translation.

First, this section analyzes the relation between the translation and rotation of the input voxel data and its corresponding effects within the 3D spectrum. Any point on $\mathbb{S}^2$ can be rotated according to Euler by an element of the rotation group $SO(3)$. The 3D rotation within a Cartesian coordinate system is then defined as a result of a multiplication of the three matrices $R(\alpha, \beta, \gamma) \in SO(3)$ corresponding to each axis, as shown in (3).

$$R(\alpha,\beta,\gamma) = \begin{pmatrix} \cos(\alpha)*\cos(\beta) & -\sin(\alpha)*\cos(\gamma)+\cos(\alpha)*\sin(\beta)*\sin(\gamma) & \sin(\alpha)*\sin(\gamma)+\cos(\alpha)*\sin(\beta)*\cos(\gamma) \\ \sin(\alpha)*\cos(\beta) & \cos(\alpha)*\cos(\gamma)+\sin(\alpha)*\sin(\beta)*\sin(\gamma) & -\cos(\alpha)*\sin(\gamma)+\sin(\alpha)*\sin(\beta)*\cos(\gamma) \\ -\sin(\beta) & \cos(\beta)*\sin(\gamma) & \cos(\beta)*\cos(\gamma) \end{pmatrix} \tag{3}$$

The 3D discrete Fourier transform on a Cartesian grid is given by (4). It can be shown by e.g. a coordinate transform that a rotation of a 3D structure by $R(\alpha, \beta, \gamma)$ orientates the magnitude of the corresponding spectrum in the same way, while a translational shift does not affect the spectrum magnitude. Having a relation between two 3D signals as in (5) with $\mathbf{x} = [x\ y\ z]^T$ and any translational shift $\mathbf{t}_s = [x_s\ y_s\ z_s]^T$, then the spectral relation is as given by (6) with $\mathbf{k} = [u\ v\ w]^T$. In terms of the magnitude this relation simplifies to (7). This relation allows a decoupling of translation and rotation for a registration process.

$$F(u, v, w) = \frac{1}{N^3} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \sum_{z=0}^{N-1} f(x, y, z) e^{-i2\pi(ux+vy+wz)} \tag{4}$$

$$r(\mathbf{x}) = s(R(\alpha, \beta, \gamma)\mathbf{x} - t_s) \tag{5}$$

$$R(\mathbf{k}) = S(R(\alpha, \beta, \gamma)\mathbf{k}) e^{i2\pi R(\alpha,\beta,\gamma)\mathbf{k}t_s} \tag{6}$$

$$|R(\mathbf{k})| = |S(R(\alpha, \beta, \gamma)\mathbf{k})| \tag{7}$$

2.3 Rotation on the sphere

The Fourier basis represents an evaluation for one- or higher dimensions on a Cartesian grid. The spherical harmonic basis is a similar tool but it is defined on the surface of a sphere. It hence provides a way to generalize the Fourier transform to any homogeneous space. The spherical harmonics of an image pair are related to each other through a shift theorem involving the irreducible representation of the rotation group (Driscoll and Healy 1994). A requirement for such spherical harmonics are orthogonal functions; the associated Legendre polynomials are given by (8).

$$P_l^m(x) = \frac{(-1)^m}{2^l l!} \sqrt{(1-x^2)^m} \frac{d^{l+m}(x^2-1)^l}{dx^{l+m}} \tag{8}$$

The function takes two integer arguments l and m, which are constrained by $l \in \mathbb{N}_0$ and $m \in [0, l]$. l is used as the band index to divide the class into bands of functions resulting in a total of $(l+1)l$ polynomials for a l-th band series. The $(2l+1)$ spherical harmonics of degree l are given in (9).

$$Y_l^m(\theta, \phi) = (-1)^m \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(\cos(\theta)) e^{-im\phi} \tag{9}$$

The spherical harmonic coefficients of a function $f(\theta, \phi) \in \mathbb{L}^2(\mathbb{S}^2)$ can be calculated by these orthonormal spherical harmonics as in (10) where any point on a sphere is given by $\eta(\theta, \phi) \in \mathbb{S}^2$.

$$\hat{f}_{lm} = \int_{\eta \in \mathbb{S}^2} f(\eta) Y_l^m(\eta)\, d\eta \tag{10}$$

For any $g \in SO(3)$ and function $f(\eta)$ with the operator Λ_g a rotation is defined as $\Lambda_g f(\eta) = f(g^{-1}\eta)$. The shift theorem gives the effect on the Fourier space of translations in the function space. A rotated spherical harmonic of degree l can be written as a linear combination of spherical harmonics of the same degree (see (11)), where the U_{ln}^m are unitary representations of the rotation group.

$$\Lambda_g Y_l^m(\eta) = \sum_{|n| \le l} U_{ln}^m(g) Y_l^m(\eta) \tag{11}$$

$$U_{ln}^m(g(\gamma, \beta, \alpha)) = e^{-im\gamma} P_{ln}^m(\cos(\beta)) e^{-in\alpha} \tag{12}$$

Makadia and Daniilidis (2003) derived the following relation (13) of the harmonic coefficients between a rotated function $\Lambda_g f(\eta)$ and the original function $f(\eta)$.

$$\Lambda_g \hat{f}_{lm} = \sum_{|p| \le l} \hat{f}_{lp} U_{pl}^m(g) \tag{13}$$

Equation (13) can be used to solve for the rotation. In Makadia and Daniilidis (2003) it was shown that nonlinear minimization techniques can be used to determine all Euler angles. The work of Driscoll and Healy (1994) dealt with the efficient calculation of Fourier transforms of functions defined on the 2D sphere $\mathbb{S}^2$, which is necessary for the computation of the spherical harmonic coefficients. The application background that motivates the work in Makadia and Daniilidis (2003) is the registration of spherical image data obtained from omnidirectional cameras. The results show that a registration is accurately possible in case full high resolution images are available. Experiments with data having only small interference of up to 15% immediatly show inacceptable deviations or the minimization algorithm did even not converge at all. This makes it unsuited for the type of registrations pursued here.

In Makadia et al. (2006) the spherical and rotational harmonic transform is used in a similar application, namely two Extended Gaussian Images are correlated for the registration of 3D point clouds (a primer on Gaussian images is Horn (1984)). The determination of the three Euler angles is carried out in an improved way, based on the work of Kostelec and Rockmore (2003). It is called *FFTs on the rotation group* and consists of a correlation approach based on the Wigner D-function and again on spherical harmonic coefficients. The data of the related experiments was generated by projections of images onto the sphere. The determination of the Euler angles using the direct correlation is more efficient as the previously used minimization technique and the presented results are promising but again require reasonable high-resolution images for a successful registration. The resampled sphere layers from the 3D spectrum used in our work do not allow a high resolution representation due to limitations concerning memory consumption and computation time with respect to real-time applicability. Furthermore such spherical spectral structures as used

in Makadia et al. (2006) are considerably corrupted by interference.

We hence developed a technique, which has two main advantages for a fast and robust registration of Euler angles from spherical information. It is based on the assumption that the use of the POMF registration is more robust with respect to misleading intensities of the matched structures compared to correlation techniques. The second reason, which requires a Fourier transform on a discrete Cartesian grid, is the need to process not only one spherical layer but a complete stack of layers in one step. A resampling scheme which allows direct phase matching of the 3D spectral structures can be expected to yield much better registration results, in particular a robust Dirac peak, than an accumulation of several layers and a subsequent corresponding correlation in 2D. The reasons will be evident later in Sect. 2.8.

2.4 Rotational resampling

The rotational information devolves to the spectrum independent from an additional translation within the structure of a sphere or its layers, respectively. The crucial point is therefore how to resample the 3D spectral grid to a Cartesian structure where the three angles can be determined. This is comparable to the polar-logarithmic resampling scheme as adopted in the Fourier Mellin Transformation (FMT) (Chen et al. 1994) in order to convert rotation and scaling to signal shifts. The goal is to obtain structures where the desired parameters can be found as signal shifts which can then be determined with phase correlation.

The general idea is to resample layers of a hemisphere at different radii on $\mathbb{S}^2$. Such a resampled layer of a hemisphere is intrinsically not a 2D rectangular matrix which is the goal for an effcient matching and registration of rotation parameters. For this reason a two-stage algorithm is applied to deal with inevitable structural distortions and the determination of angle parameters is separated. The yaw-angle can be determined over the entire range because of its rotational appearance within the unwrapped structure.

In a next step the 3D spectrum will be re-rotated according to the determined yaw angle. As a result the 3D spectrum contains afterwards only roll and pitch as a tilt within the 3D structure. The problematic part is mapping the hemisphere to a square structure for roll and pitch registration which could not be solved for a full registration in the interval $[-90°, +90°]$. This approximation has its limits, namely up to $\pm 35°$ for roll and pitch, as also motivated in a bit more detail later on in Sect. 2.7 and as also supported by experimental results. We consider this no limitation for most 3D mapping applications. Please note that the $\pm 35°$ for roll and pitch correspond to the relative changes between two subsequent 6 DOF poses where the mobile robot has taken a 3D scan, i.e., already quite fast and extreme motions in roll and pitch which are typically avoided. The yaw registration is possible within the range $[-90°, +90°]$, which also corresponds to relatively fast robot motions.

2.5 Determination of yaw

In a first step only the yaw-angle is determined which is matchable as a rotation within an unwrapped matrix. The desired structure is generated by traversing this plane in spherical coordinates (see Fig. 1).

The following pseudo-code describes the resampling of the matrix structure for yaw determination from a hemisphere of the 3D spectrum:

Rotational resampling from sphere:

```
begin
  N := Matrixsize
  Ncentre := N/2
  normf := N/π
  for x := 1 to N do
    for y := 1 to N do
      vx := x − Ncentre
      vy := y − Ncentre
      φ := atan2(vy, vx)
      γ := sqrt(vx ∗ vx + vy ∗ vy)/normf
      xp := radius ∗ sin(θ) ∗ cos(ψ) + centre3Dgrid
      yp := radius ∗ sin(θ) ∗ sin(ψ) + centre3Dgrid
      zp := radius ∗ cos(θ) + centre3Dgrid
      frame(x, y) := FFT3VOXELGRID(xp, yp, zp)
    od
  od
end
```

The two loops running over x and y define the resampled matrix. Using the position (x, y) within the matrix the

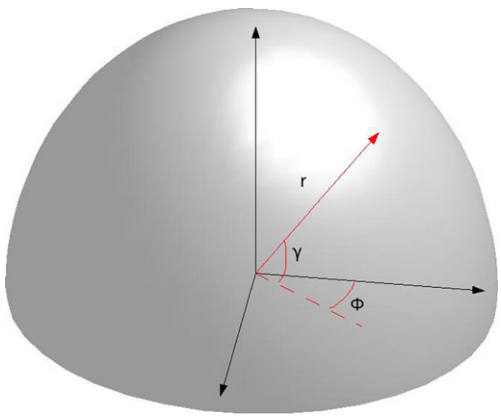


Fig. 1 The rotational resampling

values vx and vy are used to determine the angles ϕ and γ which describe in combination with a certain *radius* a hemisphere. A factor *normf* is used to squeeze the resampled hemisphere into the desired matrix size N. After the addition of the centre of the 3D spectral voxel grid *centre3Dgrid*, the corresponding vector $[xp\ yp\ zp]$ is used to access the voxel grid *FFT3VOXELGRID*, which contains the spectral data. An example of a resulting projection is depicted in Fig. 5(c).

Roll and pitch are present as an undesirable interference in this matrix. Roll and pitch are shifting the matrix in x and y direction. Hence, we have two unwrapped matrices which are translated and rotated against each other. This rotation can be decoupled from translation and will be determined in the same way as the FMT does. The basic principle of a 2D spectral registration like the FMT makes use of the fact that the translation of a signal can be decoupled from other transformations like rotation and scaling. Here only the rotation will be determined which reduces the 2D parameter space to 1D. The rotational registration will be carried out by a polar resampling. For turning this rotation into a signal shift the magnitude of the signals spectrum is resampled into polar coordinates (see (14)).

$$u_{mk} = \frac{m\frac{N}{2}}{M}\cos\left(\frac{\pi k}{K}\right) + \frac{N}{2} + 1$$

$$v_{mk} = \frac{m\frac{N}{2}}{M}\sin\left(\frac{\pi k}{K}\right) + \frac{N}{2} + 1$$

$$m = 0, \ldots, M-1; k = 0, \ldots, K-1 \tag{14}$$

In a next step this shift can be determined with the POMF as described in Sect. 2.1. This requires corresponding window functions to avoid spectral artifacts (Oppenheim and Schafer 1989; Chen et al. 1994) as for the FMT implementation. Figure 3 shows a clear peak of the rotational registration applying the POMF process on the descriptors in Fig. 2. The rotation is clearly visible as a signal shift in x direction. Experiments showed that low and high frequencies rather degrade the registration process than providing useful information. Hence, the radial axis has less information than the angular axis resulting into a non-square descriptor size. Experiments with different sets of input scans showed that using 11% to 45% of the radial axis of the 2D spectrum yields optimal registration results.

The angle can be determined from the resulting Dirac peak xd on the grid from the inverse Fourier transform as follows:

$$yaw = (xd/NPL) * \pi$$

with NPL is the size of the square grid where the zero frequency is supposed to be in the middle $NPL/2$.

Because of the circular unwrapping which was necessary for the determination of yaw, it is not suitable for the subsequent determination of roll and pitch. The translation caused

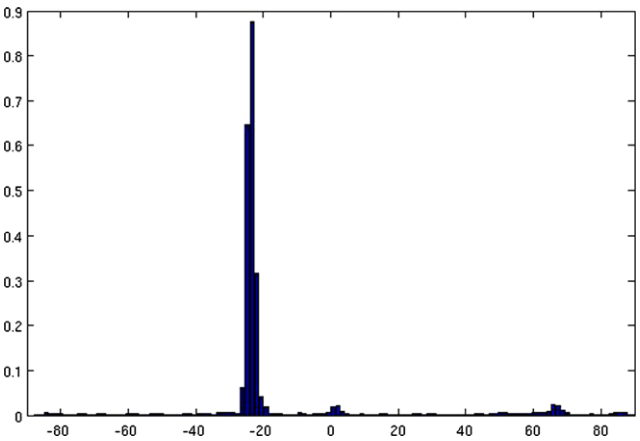


Fig. 3 A rotation peak from polar resampled Fourier descriptors

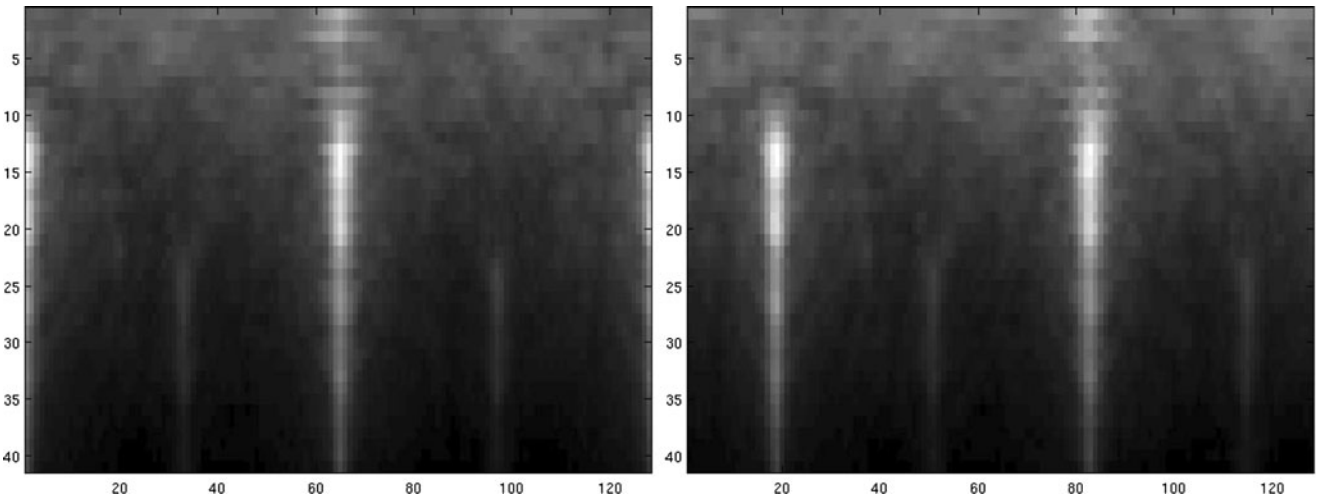


Fig. 2 Fourier descriptors from rotational resampled sphere layers

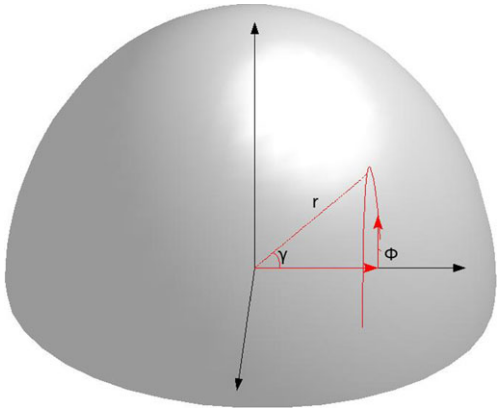


Fig. 4 Rectangular resampling

by roll and pitch is considerably distorted because the layer surface is resampled to a square shape.

2.6 Determination of roll and pitch

Now roll and pitch are determined within one registration process. The following steps are necessary for the isolation of the roll and pitch angles. The previously determined yaw angle is used to re-rotate the voxel data of the magnitude of the 3D spectrum which is then the basis for a new unwrap process. Note that if the determination of yaw was not successful the subsequent registration steps will definitely fail. Figure 4 shows the rectangular resampling in comparison to Fig. 1.

The sampling vector covers the area of each layer rectangularly, which means that the outer parts of the hemisphere are so to say squeezed into the same number of columns or rows respectively, as is the center part. A rotated vector traverses the x-axis on the hemisphere of the 3D spectrum in the same way as the resulting unwrapped matrix. The unwrapped data from the rotated vector within a range of 180° is then filled along the y-axis. The resulting unwrapped matrix contains roll and pitch solely as a translation in x and y direction. Due to this resampling scheme it is obvious that there are much more voxels to be resampled at the middle of the resampled hemisphere than at the edges. This effect is clearly visible with the example data shown in Fig. 5, namely for each of both edges of the traversed resampling axis. In order to keep these distortions effects as low as possible for the registration process, the same resampling process is carried out turning the resampling axis by 90°, compare Figs. 5(a) and 5(b). It turned out that the registration process became much more stable using both matrices, especially when the roll and pitch tilt is higher than 20° for both directions. Roll and pitch can be determined from the resulting Dirac peak (xd, yd) on the grid in 2D from the inverse Fourier transform as follows:

$$(roll, pitch) = ((xd, yd)/N_{rect}) * \pi;$$

with N_{rect} is the size of the square grid where the zero frequency is supposed to be in the middle $N_{rect}/2$.

The following pseudo-code describes the composition of the sampling vector in Fig. 4 and the resulting resampling scheme:

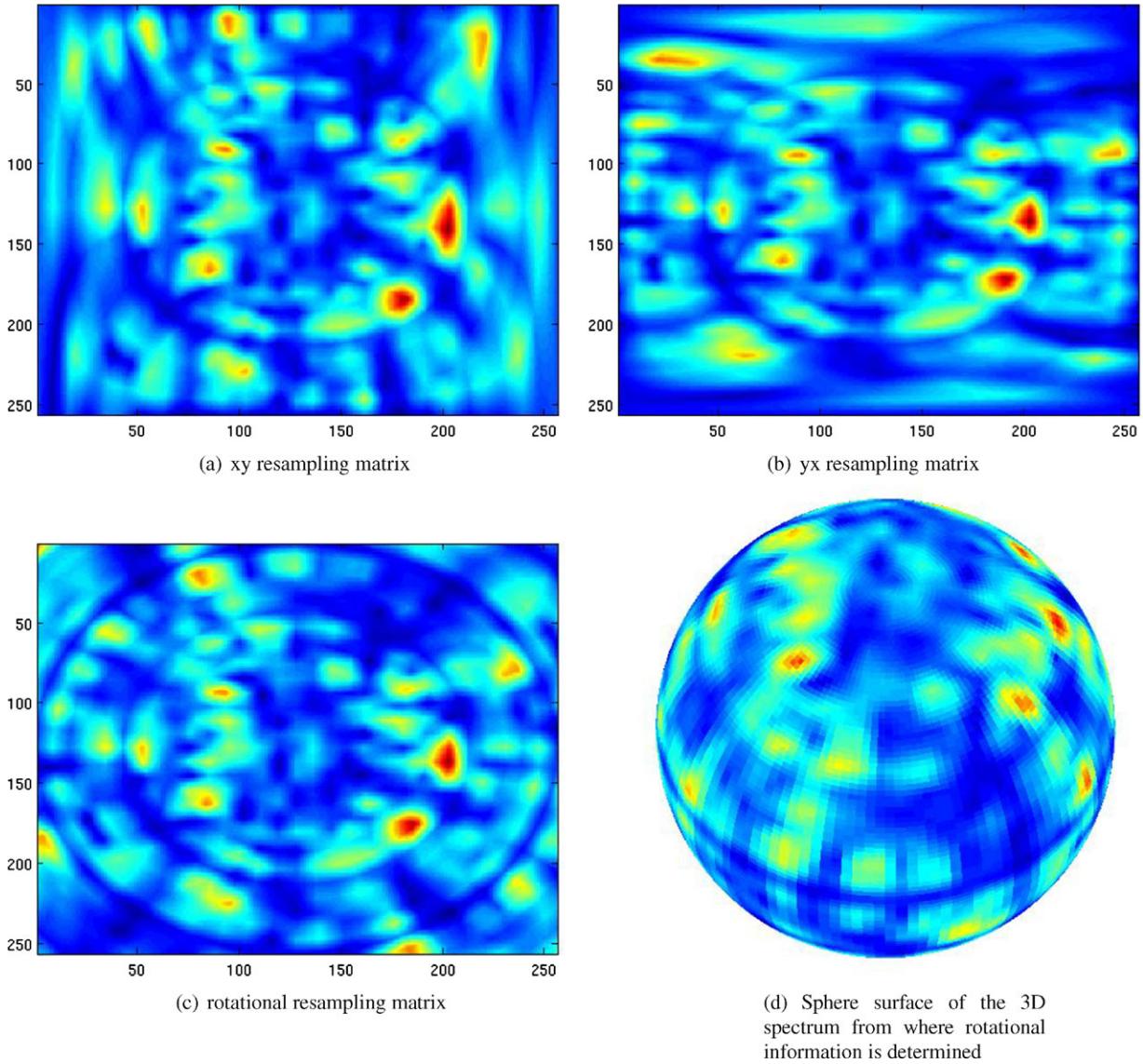
Rectangular resampling from sphere:

```
begin
  N := Matrixsize
  Ncentre := N/2
  for x := 1 to N do
    γ := −(π/2) ∗ ((x − Ncentre)/Ncentre)
    [vhx; vhy; vhz] := [1 0 0; 0 cos(γ) −sin(γ); 0 sin(γ) cos(γ)] [0; 0; 1]
    [vrx vry vrz] = [0 0 vhz]
    for y := 1 to N do
      φ = (π/2) ∗ ((y − Ncentre)/Ncentre)
      [vfx; vfy; vfz] := [cos(φ) 0 sin(φ); 0 1 0; −sin(φ) 0 cos(φ)] [vrx; vry; vrz]
      vfy = vhy
      [vfx vfy vfz] := [vfx vfy vfz] ∗ radius
      xp := vfx + centre3Dgrid
      yp := vfy + centre3Dgrid
      zp := vfz + centre3Dgrid
      frame(y, x) := FFT3VOXELGRID(zp, xp, yp)
    od
  od
end
```

The pseudo-code describes only the xy-resampling. For the yx-resampling the procedure is repeated rotated by 90°, as depicted in Fig. 5. Analogue to the described rotational resampling, two loops with x and y define the matrix of the projection result. The current x position is then used to determine γ by which a unit vector lying on the z-axis is rotated around the y-axis. The resulting vhz is then rotated around the x-axis by ϕ. In a next step the resulting resampling vector is composed by replacing vfy with the previous vhy. After the multiplication with the *radius* and the addition of the centre of the 3D spectral voxel grid *centre3Dgrid*, the resulting vector $[xp\ yp\ zp]$ is again used to access the voxel grid *FFT3VOXELGRID*. The described heuristic with an equidistant division of the angles γ and ϕ allows a rectangular projection of the hemisphere.

2.7 Approximation of the 3D rotational registration

The described process of resampling is actually a projection of the spherical layer which leads inevitable to deviations from the true roll/pitch rotation compared to the minimization approach in Makadia et al. (2004). Figure 6 demon-



(a) xy resampling matrix

(b) yx resampling matrix

(c) rotational resampling matrix

(d) Sphere surface of the 3D spectrum from where rotational information is determined

Fig. 5 Example data illustrating the resampling of structures from the 3D spectrum

strates the problem. Any point **a** which can lie on the resampled hemisphere which is rotated by roll and subsequently rotated by the pitch angle will be affected by a mutual influence. The projection on the z-plane of this vector is denoted as (xd', yd') which represents the corresponding registered pitch and roll. After rotations around roll and pitch from (xd, yd) such projections will not lie on the projection axis of **a** but slightly shifted to (xd', yd'), called yaw-deviation in Fig. 6, hence moving within the resampled plane from the x-axis (roll-angle) to the y-axis (pitch-angle) as indicated by the bent arrow. Since the content of the hull of a hemisphere is resampled, a vivid illustration would be a plane rotated as described beforehand and later on projected to the z-plane. The first rotation around roll would squeeze the pattern on that plane along the x-axis and a second rotation around pitch would not only squeeze it along the y-axis but furthermore rotate that pattern away from the y-axis while the orientation on the x-axis would remain constant.

The following depictions of the registration process clarify these effects. The described distortions inherent to the registration process are neglectable for smaller angles up to 15°, for larger angles this effect becomes more evident. Figure 7 shows examples of the corresponding registration peaks for roll and pitch from 5° to the maximum angle of 35° where this registration approach has its limits. It is clearly visible that the peaks gets broader becoming a bent shape due to the geometric ambiguity. The same effect of broadening the peak applies for the 1D registration for the yaw angle as Fig. 8 shows, though there higher angles up to ±90° are feasible. As mentioned, we consider this to be

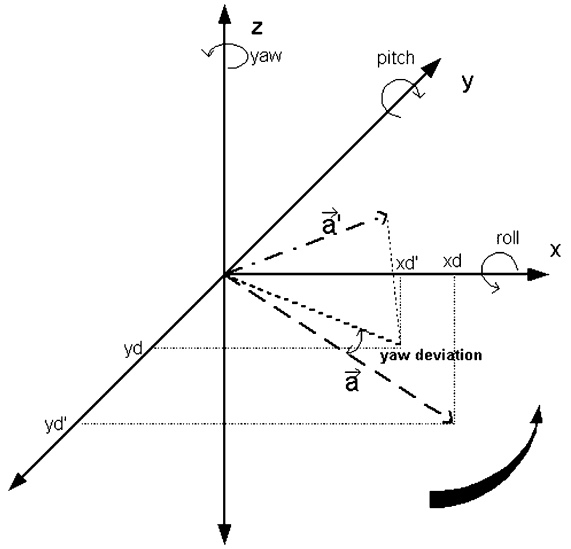


Fig. 6 Registration deviation caused by the resampling approximation

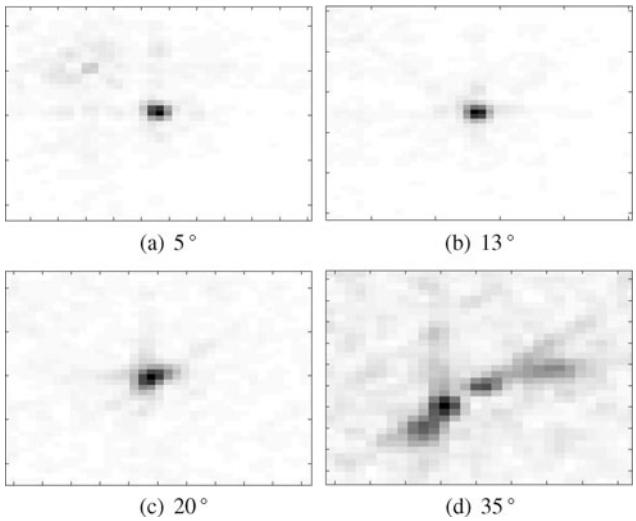


Fig. 7 Examples of 2D registration peaks for different roll and pitch angles

still very reasonable values for 3D mapping as mobile robots rarely undergo extreme full 6 DOF motions, especially with respect to changes in roll and pitch. Also, these limits are a trade off needed to be able to register 3D data that has only partial overlap and that is strongly affected by noise, which are both two important aspects for 3D mapping in general and for the sonar based maps presented here in particular.

Another undesired effect as already presented in Fig. 6 is the increasing deviation of yaw which is also visible in Fig. 8 from left to the right. The effect is illustrated with a simulated test object that is rotated along the different axes. The registration results for different combinations of rotations—each with the maximum possible tilt of roll/pitch—are compared to the ground truth values. In case only one angle of roll/pitch is rotated or none of them, the yaw registration is always exact for the whole range as the example in Fig. 9(a) shows. In case both roll and pitch angles are rotated, a constant offset which depends on roll/pitch is present (Fig. 9(b)). Concerning roll and pitch, the corresponding deviations depend on the rotation sequence which is assumed by the registration process. The pitch rotation as the second step is supposed to shift the matching structures on the x-axis but also rotates the structure and furthermore causes a shift on the y-axis. Figure 9(c) shows the undesired shift on the y-axis as the corresponding deviation for the registered roll angle.

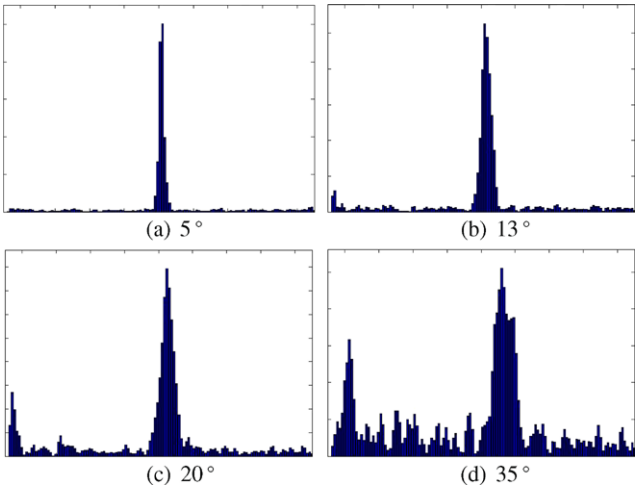
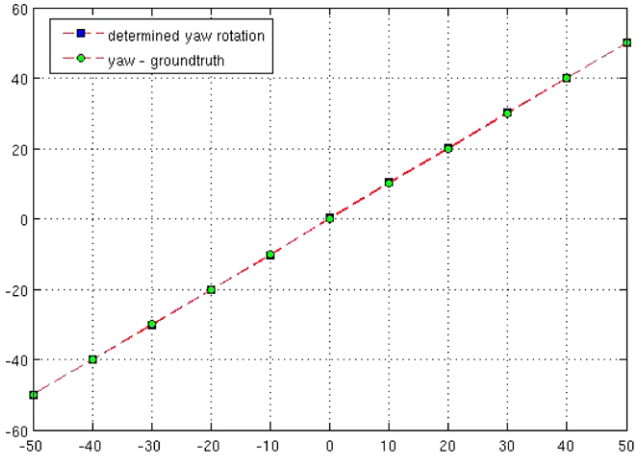


Fig. 8 Examples of 1D registration peaks for different yaw angles

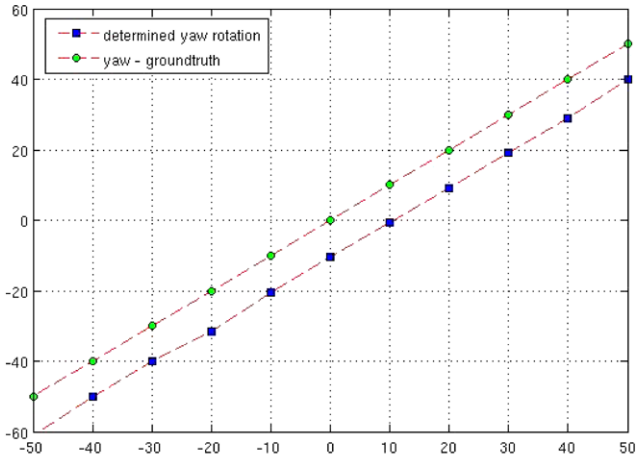
Please note that the resulting errors of these effects are relatively small and acceptable for the practical applications that we have in mind. Our spectral method is only suited for coarse registrations anyway as there always will be some minor errors due to the discretization effects of the grids. Nevertheless, our spectral method is already suited for 3D mapping by registration only as illustrated with 18 real world sonar scans in Sect. 4 where it outperforms alternative registration techniques like ICP. Also, it can be expected that significantly larger sequences of scans can be used for 3D mapping if the spectral registration is embedded in proper Simultaneous Localization and Mapping (SLAM) where these errors can be optimized away when loops occur. This requires the computation of uncertainty estimations in the registrations. The possibility to extend our previous work on uncertainty estimation for 2D spectral registration (Pfingsthorn et al. 2010) to 3D is planned for future work.

2.8 Multilayer parameter registration

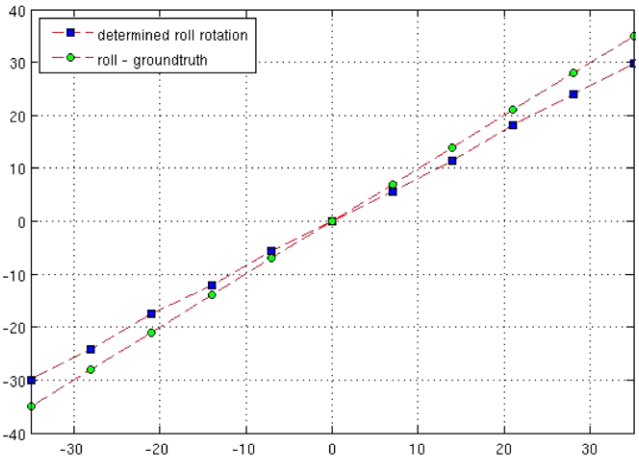
The spectral structure at one certain radius works fine for registration when noise-free, completely overlapping—so to say absolutely perfect—data is used, as experiments with simulated data show where an ideal 3D scan in combination with a rotated and translated copy of exactly the same data



(a) yaw = roll = 35 °, pitch = 0 °



(b) yaw = roll = 35 °, pitch = 35 °

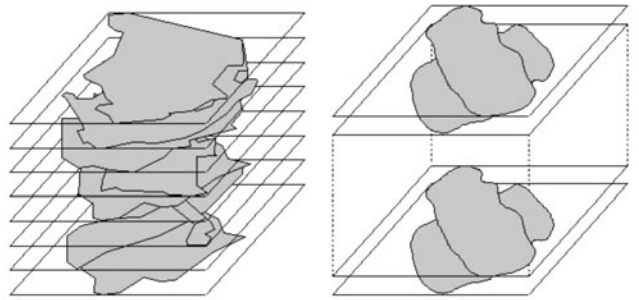


(c) roll = yaw = 0 °, pitch = 35 °

Fig. 9 Examples of possible angle registration deviations in their mutual dependence

are used. As soon as interference and occlusion affects the data, the registration immediately becomes unstable.

But the rotational information is present within the entire 3D structure and therefore at all radii. Resampling the



(a) yaw - one layer (preprocessed Fourier descriptors) from each radius

(b) roll/pitch - two rotated layers from each radius

Fig. 10 An illustration of the pile of layers from different radii

structures as described in Sects. 2.5 and 2.6 for a whole radius range improves the registration process considerably. As one can expect and as is confirmed through tests this process only becomes more stable within a certain range. Having a cubic size N, the optimal range for the resampled hemisphere turned out to be from $(0.2 \ldots 0.8) * (N/2)$. Frequencies too low decrease the registration peak because of the insufficient available voxel data which can be resampled. Note this effect of having a restricted number of voxels adds to the unavoidable distortions during the resampling process. Higher frequencies are obviously rather decreasing the information content caused by occluding and interfering structures.

An example illustration of the resampled 3D structures is shown in Fig. 10. It consists of one resampled pile for yaw (Fig. 10(a)) and one resampled for roll/pitch (see Fig. 10(b)), i.e., the resampled 3D structures contain the desired information in x and y while having the intentionally redundant information on the z axis. Note that the rectangular resampling at each radius for roll and pitch registration is done with an additional 90° rotated counterpart, see Fig. 5. Figure 2 shows a 2D structure which is the result of an accumulation of the described multilayer set.

2.9 3D registration of resampled structures

For a complete 3D registration, i.e., the processing of the multiple layers, the resampled layers are itself arranged into a 3D grid, as described in Sect. 2.8. Afterwards, the shift registration by the POMF is carried out in 3D in order to employ all information (Fig. 10) for a stable registration. Note that the first 3D stack (Fig. 10(a)) consists of preprocessed Fourier descriptors (see Sect. 2.5). The second stack of resampled layers for roll and pitch (Fig. 10(b)) does not need a preprocessing because structures contain the desired parameters as x and y translation. Hence the POMF process can directly be applied. Since the structures for roll/pitch and the descriptors for yaw are resampled structures and not pixelwise shifted signals, a postprocessing by an interpolation fil-

ter as described in Buelow et al. (2009) for an application of the FMT improves the stability of the registration.

2.10 Determination of translation

Once the 3D rotation is correctly determined, the subsequent registration of the full 3D translation is actually straightforward. According to the determined angles (roll, pitch and yaw), the voxel data of the first scan is re-rotated and the corresponding 3D spectrum with the full phase information (see (4)) is calculated. Note that the 3D rotation of the scan data and a subsequent calculation of the spectrum is the better alternative compared to a direct rotation of the spectrum which requires interpolation of complex data. The 3D registration is then done by a 3D POMF registration for translation. In case the data is correctly aligned and that there is sufficient overlap between both scans, it will yield a clear peak according to (2) in the same way as already used for the multilayer angle registration.

2.11 Overview on the complete process

The complete process can be summarized as follows:

1. resample hemispheres (rotational unwrap) on different radii from the magnitude of the 3D spectrums from two corresponding 3D grids
2. determine the yaw angle by a rotational registration (polar resampled POMF process) from the resampled structures
3. re-rotate the 3D spectrum by the determined yaw angle in order to align the corresponding 3D spectrums according to yaw
4. resample hemispheres (rectangular unwrap) on different radii from the magnitude of the first 3D spectrum and the re-rotated 3D spectrum
5. determine roll and pitch angle by a translational registration (3D POMF) from the resampled structures
6. re-rotate the 3D spectrum by the determined roll and pitch angle in order to align the corresponding 3D spectrums according to 3D translation
7. determine the 3D shift of the first 3D spectrum and the according to roll, pitch and yaw re-rotated 3D spectrum by a 3D POMF process

3 Experiments and results with simulated data

Before experiments and results with real world data are presented, the robustness of our registration approach is tested with simulated data where ground truth transformations are known and noise can be controlled. Concretely, four different datasets are generated, each with a specific noise corruption on a simulated scene. Please note that for real 3D sonars, the type and amount of structural errors and noise strongly depends on the concrete device as well as on the environment. We use three different noise models, which are explained below, to conduct different, independent tests with each of them. The registration is carried out on a $256 \times 256 \times 256$-grid. The side-length of each voxel corresponds to one meter in this simulation. The algorithm is currently implemented in MATLAB. One of its advantages compared to alternative methods—which are also discussed in Sect. 4—is that it has a fixed computation time. So, all registrations presented in this section do require exactly 60.6 seconds per scan pair in the current MATLAB version.

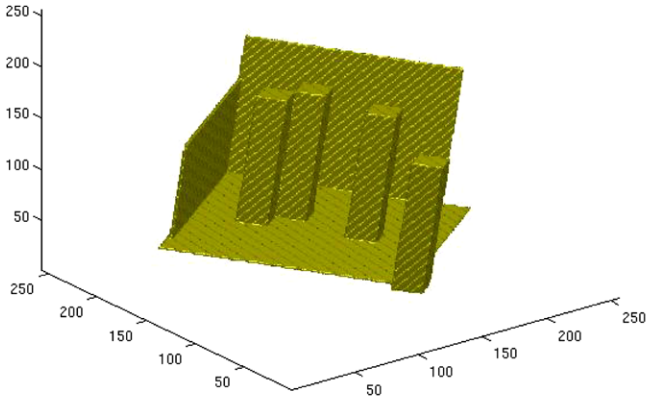
In scenario one and two the noise is extended to an envelope around the structure in two different margins, namely 5 and 10 voxel. In scenario three the noise is added to the entire registration area. In each case, the noise is uniform within the different regions. Two different levels of noise are used, namely 1/25 and 1/100. This means that 1 out of 25, respectively 100 voxel is flipped, i.e., an occupied voxel becomes void or vice versa. Please note that 1/25, respectively 1/100 roughly corresponds to one swapped voxel per $3 \times 3 \times 3$, respectively $5 \times 5 \times 4$ piece of volume. Figure 11 shows examples of the simulated data with a voxel swap of 1/100, i.e., the less noisier of the two cases. In total, there are seven different types of scans:

D_0: noise-free
D_1, D_2: noise on the structure and in a 5 voxel envelope; swap ratio 1/100 (D_1), respectively 1/25 (D_2)
D_3, D_4: noise on the structure and in a 10 voxel envelope; swap ratio 1/100 (D_3), respectively 1/25 (D_4)
D_5, D_6: noise all over; swap ratio 1/100 (D_5), respectively 1/25 (D_6)

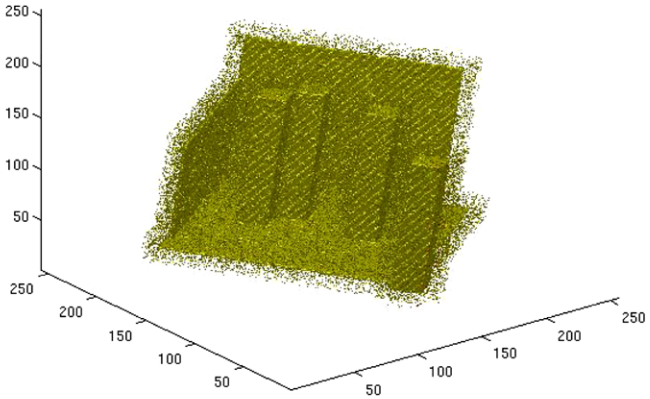
In a first set of experiments, the performance on registering translated data is evaluated. In order to assess the registration results, corresponding scan pairs are generated by shifting the artificial structure concurrently in different steps on the x-, y- and z-axis. In this sequence the step size is one, two and three for x, y an z which is linearly increased within a loop. Hence with this increasing shift, the structures are moving out of the registration frame, i.e., the overlap between these virtual scans is more and more decreased.

First, the translation experiment is done without any noise added, i.e., D_0 is used. Figure 12 shows the results without any noise, the last scan which still works has an Euclidean distance of 150 voxels, respectively meters for the settings in this simulation. Figure 13 shows the corresponding virtual scan. Please note the small amount of overlap of this scan with the original scene.

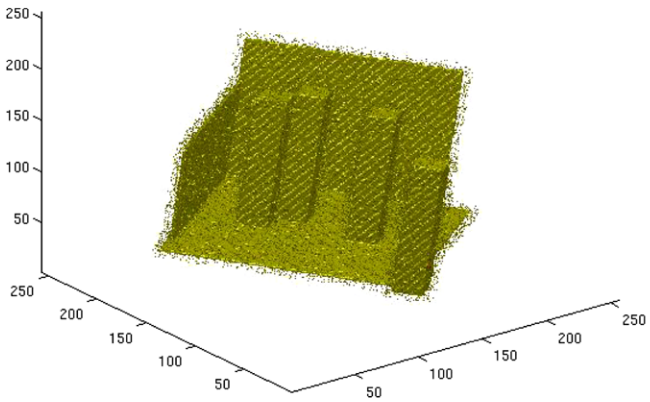
This experiments shows that small overlaps can still be registered. A crucial question is of coarse how this is affected by noise. Hence, the same test is done for the different structures of uniform noise at a swap level of 1/100, i.e.,



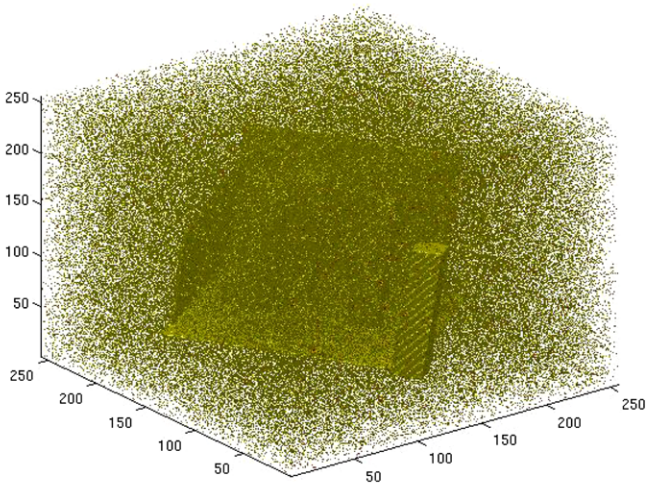
(a) Simulated structure used for evaluating the registration method.



(c) Structure + enveloping region (10 Voxel) corrupted by noise.



(b) Structure + enveloping region (5 Voxel) corrupted by noise.



(d) Entire scene corrupted by uniform noise.

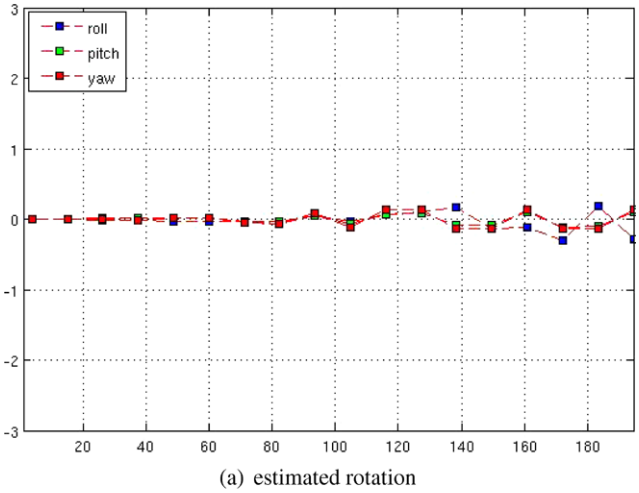
Fig. 11 Simulated data with different forms of uniform noise

with the scans sets D_1, D_3, and D_5. As mentioned, the swap ratio corresponds to flipping one voxel per $5 \times 5 \times 4$ piece of volume, which is already quite noisy as Fig. 11 shows.

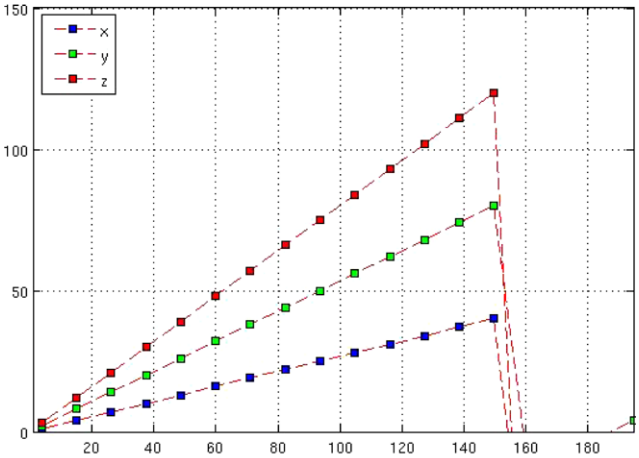
The results in Fig. 14 show that under the presence of quite some noise, the registration is very stable over a long range even if the overlap becomes quite small due to the translation. As can be expected, the registration becomes more unstable in case the entire area is corrupted by noise. The registration results for the data with noise corruption in the structure plus an additional enveloping region show that this kind of noise has nearly no effect on the registration results. Since sonar data is relatively noisy around structures it emphasizes especially the applicability of the proposed method.

The same set of experiments is repeated for noise generated with a swap rate of 1/25 voxel, i.e., with the scans sets D_2, D_4, and D_6. Note that this corresponds to flipping one voxel per roughly $3 \times 3 \times 3$ piece of volume, i.e., that this is a severe corruption of the original data. As can be expected, the registration works with more noise only for smaller shifts as Fig. 15 shows. Nevertheless, when the noise is restricted to envelopes around the main structures, which is not a completely unrealistic assumption, the results are not too far off from the noise-free case and almost perfect values are computed until a translation of 140 voxels, respectively meters. Even in the extreme case of uniform noise all over the scans that are to be registered, there are still surprisingly good results up to about 110 meters of translation.

A similar series of experiments is done with rotational displacements. The series has a fixed tilt for roll and pitch of both 10° with a continuous rotation of yaw starting from 8° up to 88°. Please note that the values for roll and pitch are relative values between scans, i.e., they do not correspond to the absolute orientation in the world frame but to the change in orientation of the sensor/robot poses between scans. This also holds for yaw.



(a) estimated rotation



(b) estimated translation

Fig. 12 Translation experiment using D_1 without noise

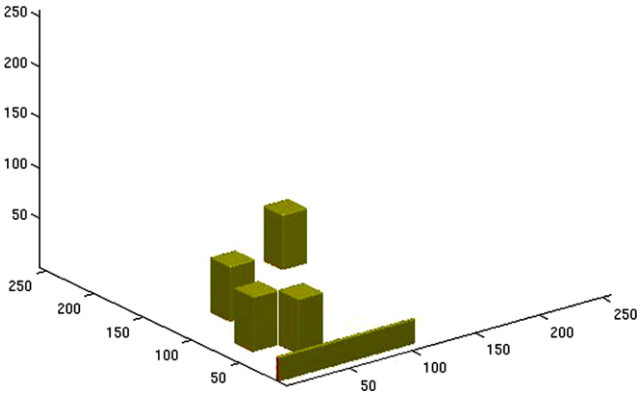


Fig. 13 This virtual scan can still be registered with the scene shown in Fig. 11(a) despite the very small overlap

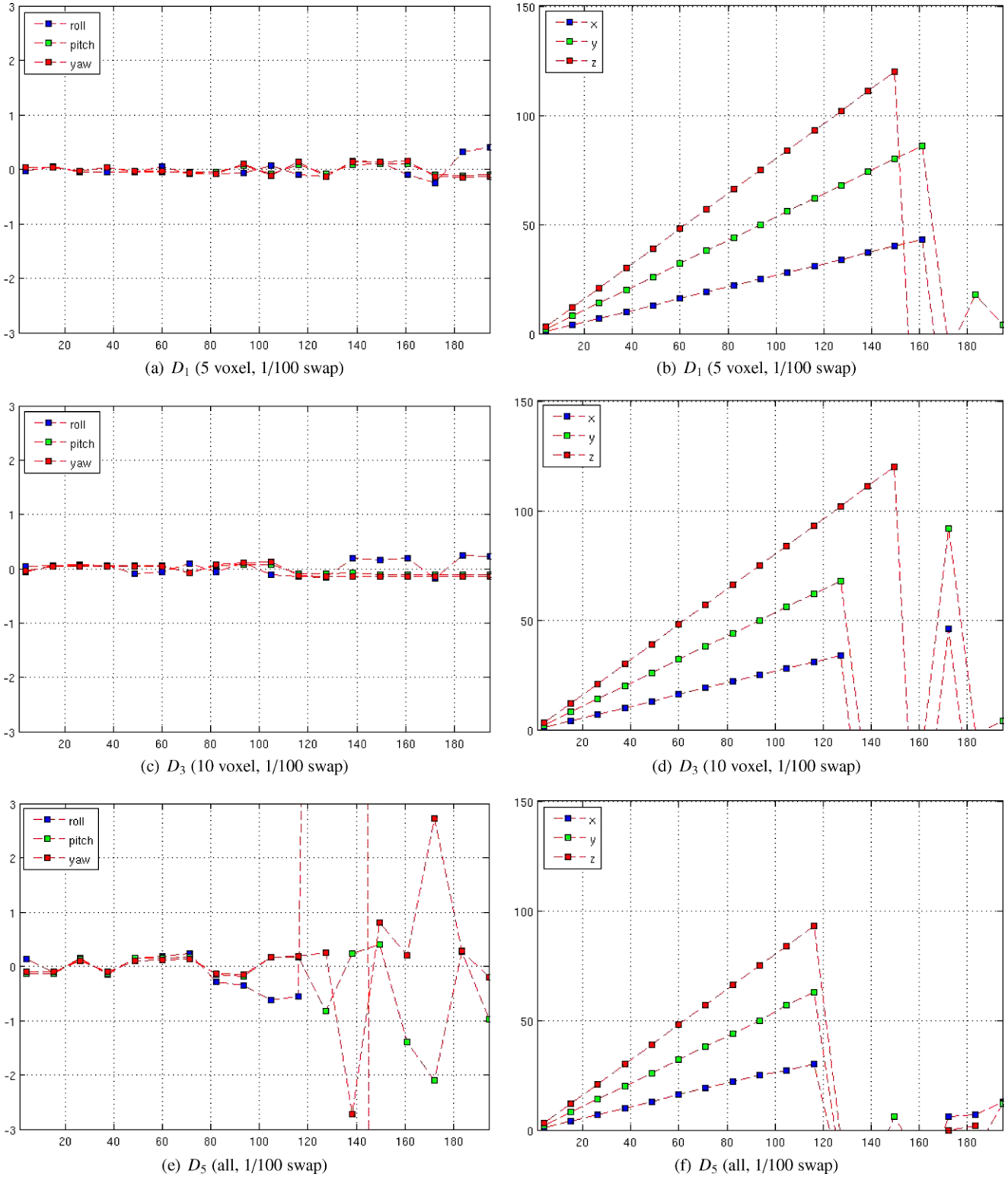
Figure 16 shows the results in the noise free case. The small deviations with increasing yaw are due to the approximation inherent to the algorithm as discussed before.

The results with severe noise corruption, namely a swap ratio of 1/25, shown in Fig. 17 indicate that the registration is very robust. Only in the extreme case of uniform noise all over the registration grid some minor deviations from ground truth can be seen.

Finally, our method is tested with Gaussian noise along virtual beams from the sensor. This means that given a sensor pose, ray-tracing to the structure is used. The effect of this is that also radial distortions along the sensor beam are present. The variance σ^2 of the Gaussian noise is proportional to the length of the beam, i.e., the noise increases with the distance. The maximum variance is set in two experiments to $\sigma_1^2 = 1$, respectively $\sigma_2^2 = 2$ grid-voxels. This corresponds to a variance of roughly 1 meter, respectively 2 meters for the settings used with the real sensor presented in Sect. 4. This exceeds the nominal noise of the real sensor, but it also has to be noted that the structural errors and the noise in the real data are much more complex—especially with respect to their dependency on the layout of the environment and on the acoustic parameters of different environment parts. Figure 18 shows the simulated scene for σ_1^2 from the perspective of the sensor, which is simulated with a 120×90 degrees field of view. Please note that this visualization shows the data already entered into the registration grid.

For the sake of completeness, the simulated sensor is simultaneously translated and rotated in these two sets of experiments, i.e., there are full 6 degree of freedom transformations between the scans. Table 1 shows the parameters for the 10 continuously increasing transformations that are used. The reference scan S_r corresponds to the original scene plus noise as shown in Fig. 18. Each new scan S_i—with increasing distance to S_r and its own noise added—is than registered with S_r. Please note that the overlap between the scans becomes quite small. Figure 19 shows the scan S_8 for which registration with a noise variance of $\sigma^2 = 1$ still succeeds.

The results of the registrations are shown in Figs. 20 and 21 for the two different variances. With the variance of $\sigma_1^2 = 1$, the registration succeeds up to scan S_8. This scan has undergone a spatial transformation of a simultaneous rotation by $(54°, 18°, 9°)$ and a translation of (36, 72, 108) voxels. Please note that this translation corresponds to a Euclidean distance of about 135 voxels. Even the very high noise of $\sigma_2^2 = 2$ still allows quite some motion between scans. The registration still succeeds with scan S_6, has undergone a spatial transformation of a simultaneous rotation by $(42°, 14°, 7°)$ and a translation of (28, 56, 84) voxels, i.e., a Euclidean distance of 104 voxels.



(a) D_1 (5 voxel, 1/100 swap)

(b) D_1 (5 voxel, 1/100 swap)

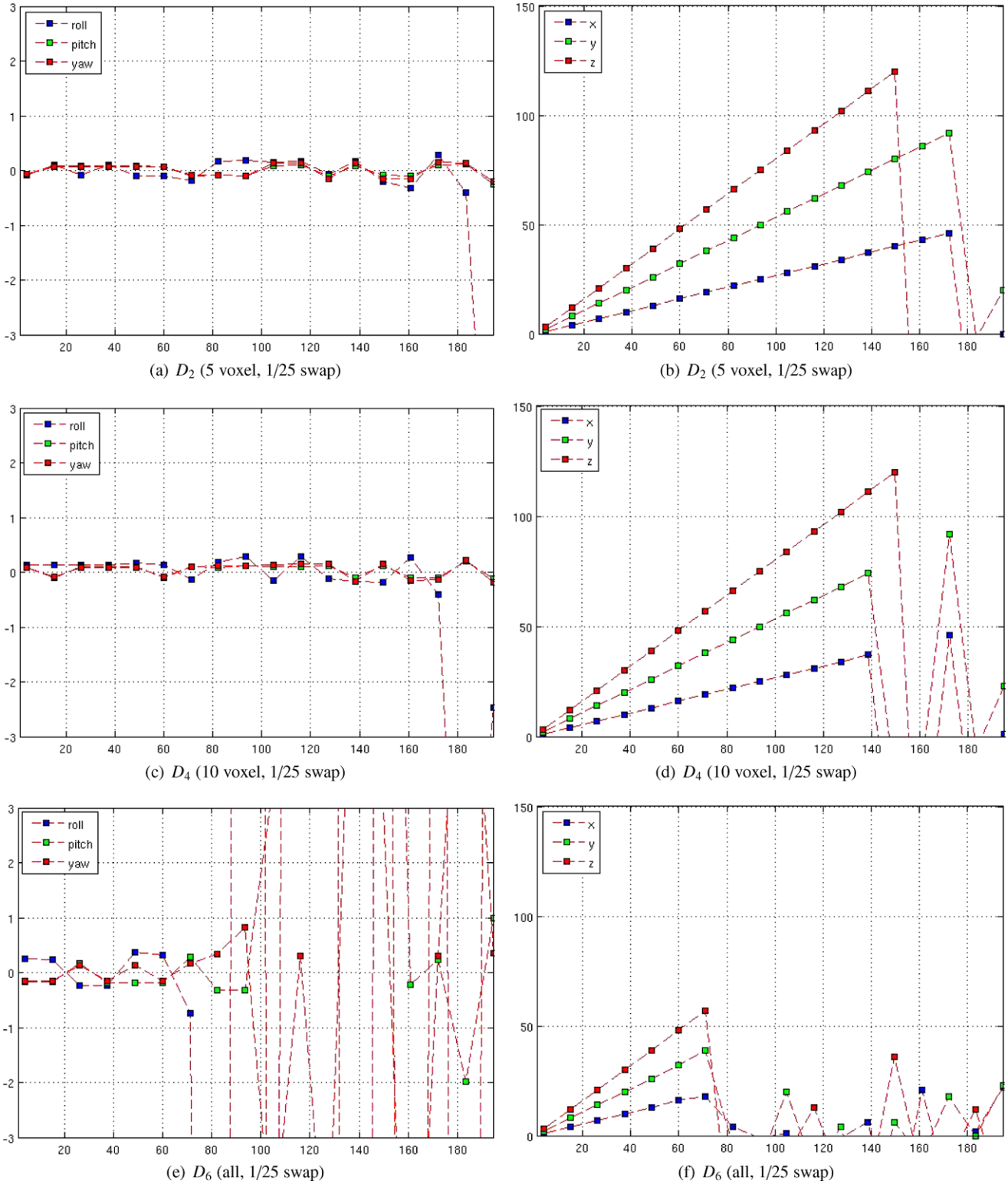
(c) D_3 (10 voxel, 1/100 swap)

(d) D_3 (10 voxel, 1/100 swap)

(e) D_5 (all, 1/100 swap)

(f) D_5 (all, 1/100 swap)

Fig. 14 The translation experiment with different types of noise at a swap rate of 1/100. The resulting rotation (*left*) and translation (*right*) values are shown for different amounts of ground truth translation. All angles are in degrees; all translations are in voxels, respectively meters as the side-length of each voxel correspond to one meter in this simulation

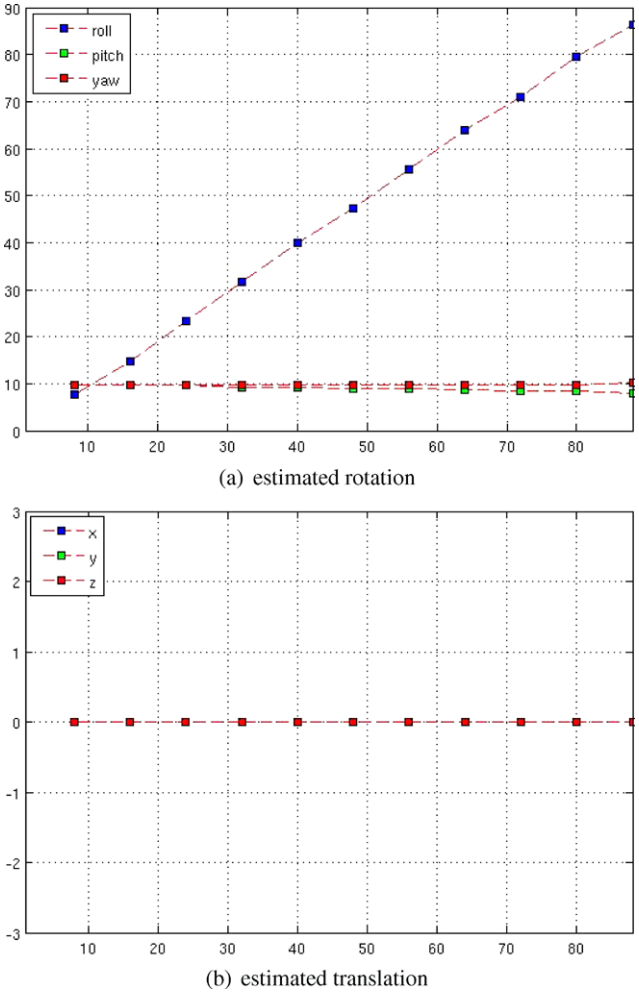


(a) D_2 (5 voxel, 1/25 swap) (b) D_2 (5 voxel, 1/25 swap)

(c) D_4 (10 voxel, 1/25 swap) (d) D_4 (10 voxel, 1/25 swap)

(e) D_6 (all, 1/25 swap) (f) D_6 (all, 1/25 swap)

Fig. 15 The translation experiment with different types of noise at a swap rate of 1/25. The resulting rotation (*left*) and translation (*right*) values are shown for different amounts of ground truth translation. All angles are in degrees; all translations are in meters



(a) estimated rotation

(b) estimated translation

Fig. 16 Rotation test on D_0 without noise

4 Experiments and results with real world data

4.1 The tritech ecplipse sonar

In the following section experiments with real world data are presented. The device used in the experiments presented here is a Tritech Eclipse sonar. It is a multibeam sonar with time-delay beamforming and electronic beam steering. Its core acoustic sensing parameters are:

- Operating Frequency: 240 kHz
- Beam Width: 120°
- Number of Beams: 256
- Acoustic Angular Resolution: 1.5°
- Effective Angular Resolution: 0.5°
- Depth/Range Resolution: 2.5 cm
- Maximum Range: 120 m
- Minimum Focus Distance: 0.4 m
- Scan Rate: 140 Hz at 5 m, 7 Hz at 100 m

The scan time depends on the maximum range of the device as well as the horizontal/vertical field of view and resolution, which can all be set in software. In our experience, the device is working best over longer distances, i.e., the maximum range being set somewhere in the range of 40 to 120 meters. In the experiments presented here, the maximum range was set to the physical limit of 120 meters.

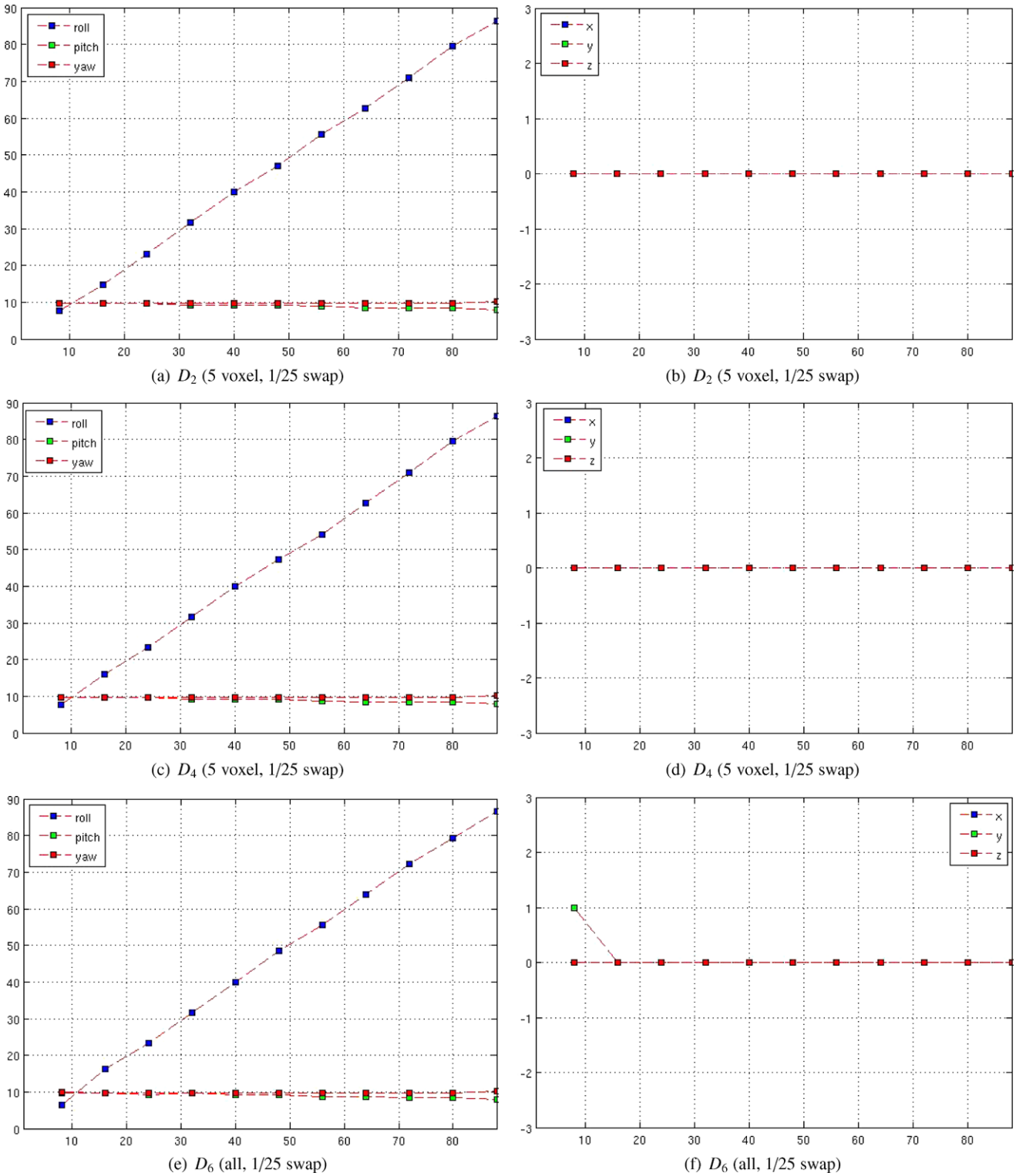
Table 1 Rotation and translation parameters for the full 6 degree of freedom transformations between simulated scans with Gaussian noise along the virtual sensor beams added

Scan Nr.	x	y	z	yaw	roll	pitch
0	4	8	12	6°	2°	1°
1	8	16	24	12°	4°	2°
2	12	24	36	18°	6°	3°
3	16	32	48	24°	8°	4°
4	20	40	60	30°	10°	5°
5	24	48	72	36°	12°	6°
6	28	56	84	42°	14°	7°
7	32	64	96	48°	16°	8°
8	36	72	108	54°	18°	9°
9	40	80	120	60°	20°	10°

The device was used to generate 18 scans of the Lesumer Sperrwerk, a river flood gate in the north of Bremen, Germany (Fig. 22). The sonar data is quite noisy and error-prone. Hence, a pre-filtering using a threshold on the intensity values was done, i.e., readings with a weak echo got discarded. In addition of a reduction in noise and the overall amount of data, it lead to a significant reduction of the field of view of the sonar to about 90° opening angle—instead of 120°—as the center is most illuminated by sound; an effect which is also described in the device's manual. The data is despite this simple pre-processing still quite noisy. Two example point clouds from the scans are shown in Fig. 24. The scans have varying amount of overlap, ranging from about 90 to 50 percent between consecutive scans.

The 18 scans are registered with three different methods for comparison as described in the following sub-sections. The two methods chosen for comparison are the Iterative Closest Point (ICP) algorithm and the recently introduced Plane-based Registration, which are two methods that are also considered to be robust with respect to noise and that can cope with only partial overlap, i.e., criteria that are extremely important for 3D mapping.

Please note that we are interested in evaluating the results of the relative spatial transformations given by the registrations, i.e., the structure of the map. Top views on the resulting 3D maps are therefore compared with an image from Google Earth as a representation of the ground truth structure. As we are interested in assessing the quality of the overall structure of each 3D map, the top view images and the one from Google earth are manually co-aligned. Concretely, the place of the deployment of the system is used



(a) D_2 (5 voxel, 1/25 swap)

(b) D_2 (5 voxel, 1/25 swap)

(c) D_4 (5 voxel, 1/25 swap)

(d) D_4 (5 voxel, 1/25 swap)

(e) D_6 (all, 1/25 swap)

(f) D_6 (all, 1/25 swap)

Fig. 17 The rotation experiment with different types of noise at a swap rate of 1/25. The resulting rotation (*left*) and translation (*right*) values are shown for different amounts of ground truth yaw. The scans are always tilted by 10° in both roll and pitch to generate changes in these angles, too. All angles are in degrees; all translations are in meters

to provide a geo-reference for only the very first scan. The image scales are based on the meter-scale given in Google earth and the range values in meters of the Tritech Eclipse.

4.2 Iterative Closest Point (ICP)

In a first experiment, the Iterative Closest Point (ICP) algorithm is used to register the data (Besl and McKay 1992). As the name suggests, the algorithm is based on the heuristic that nearest neighbors correspond and hence "moves" them "towards each other" in each iteration of the registration process. ICP and its variants is a very popular basis for 3D mapping. It is for example stated in Weingarten (2006) that "up to now, all approaches successfully applied to 3D SLAM are based on the ICP algorithm".

The 3D map resulting from ICP-registrations is shown in Fig. 25. It reasonably corresponds to the ground truth structure though there are several clear deficiencies in map; see also Fig. 26. First of all, the points of the different scans are not well aligned and there is a high amount of widely dispersed points. There is a large amount of noise in the map and the underlying structures are not only vaguely recognizable. Second, the 3D map is significantly distorted along one axis. The map part C is far from the corresponding river shore. This could in theory be adjusted by using a different scaling of the top view image, assuming that the Tritech range values are not properly calibrated. But then either map part A or B would not correspond to the related structures anymore as the ratios of the relative distances between A, B and C are fixed within the map. Third, there are several clearly misaligned wall structures, especially the map parts marked as d, e, and f in Fig. 26. It has to be noted that ICP is known to perform better if good initial guesses can

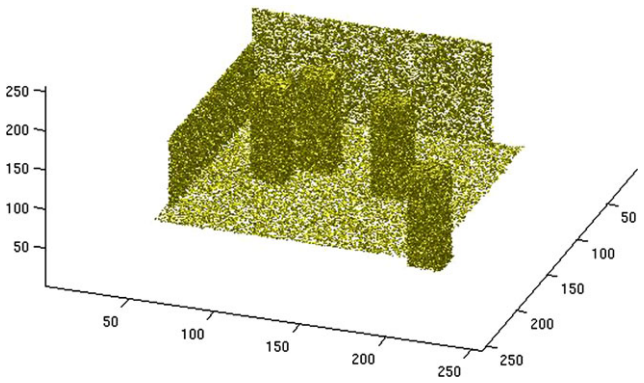


Fig. 18 Simulated data with Gaussian noise along virtual beams from the sensor. This scene corresponds to the reference scan S_r. It is shown from the perspective of the simulated sensor

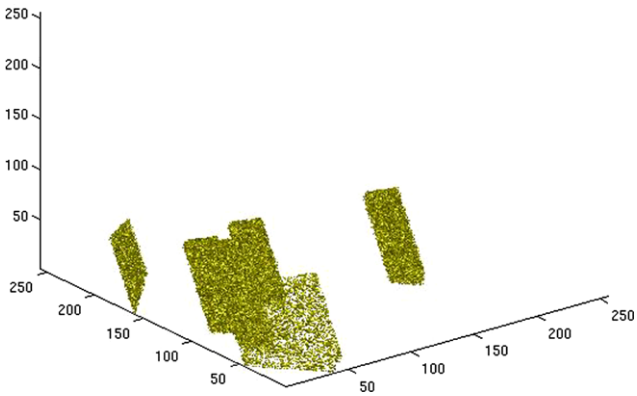
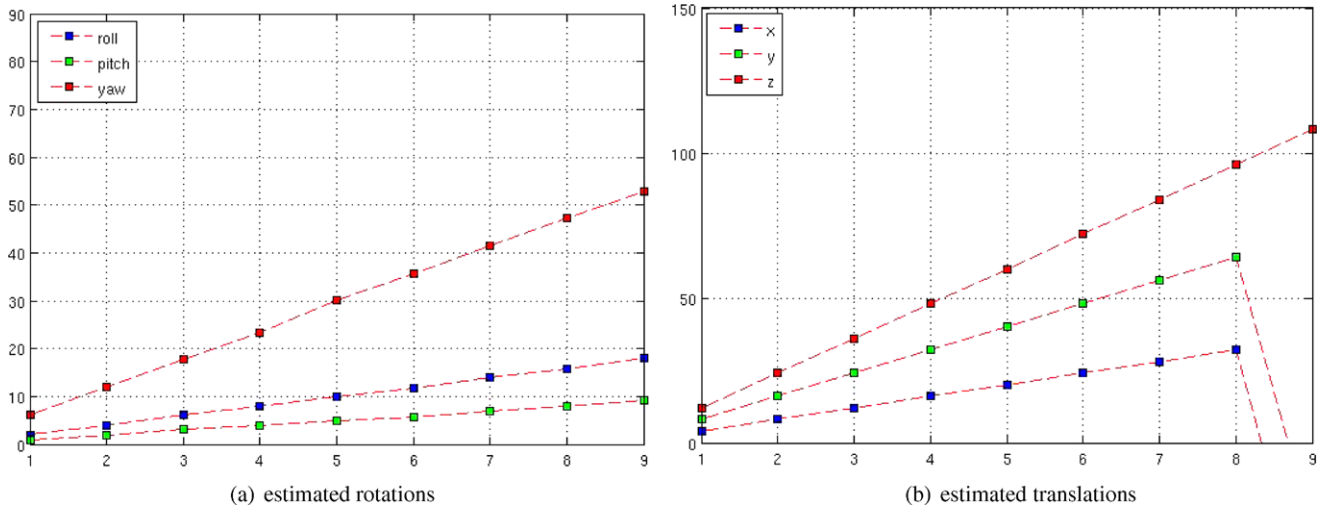
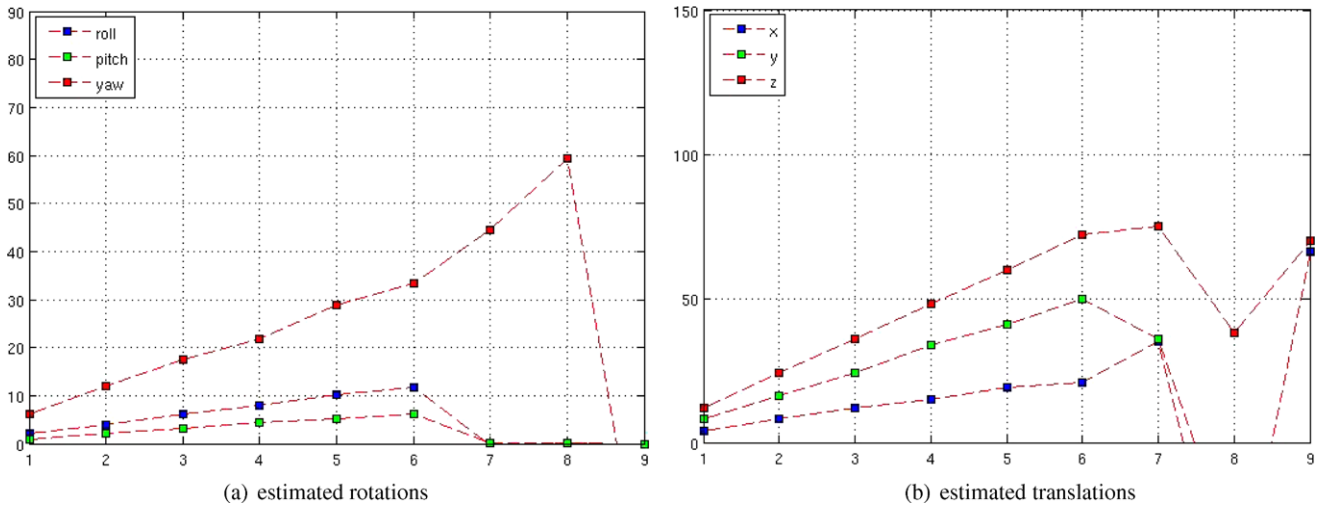


Fig. 19 Scan S_8 from the Gaussian noise data-set with full 6 degree of freedom transformations. Despite the very small overlap and the high amount of noise, it can still be registered with the reference scan S_r shown in Fig. 3



(a) estimated rotations (b) estimated translations

Fig. 20 The transformation parameters estimated by the registration of the scans S_i with the reference scan S_r under Gaussian noise with variance $\sigma_1^2 = 1$. The registrations succeed up to scan S_8, i.e., a rotation with (54°, 18°, 9°) plus a translation with (36, 72, 108) voxels, i.e., a Euclidean distance of 135 voxels, respectively meters



(a) estimated rotations (b) estimated translations

Fig. 21 The transformation parameters estimated by the registration of the scans S_i with the reference scan S_r under Gaussian noise with variance $\sigma_2^2 = 2$. The registrations succeed up to scan S_6, i.e., a rotation with (42°, 14°, 7°) plus a translation with (28, 56, 84) voxels, i.e., a Euclidean distance of 104 voxels

Fig. 22 An overview of the Lesumer Sperrwerk as seen from the river's surface

be provided as starting conditions; this could for example be done by having special navigation sensors like a Doppler Velocity Log (DVL) on the vehicle or by deriving accurate hydrodynamic models.

Table 2 shows the run-times for each registration on a PC with a Core 2 Duo CPU with a 1.8 GHz clock under Ubuntu Linux. The ICP is implemented in C++ in a standard version using a kd-tree. The average run-time is about 21 second. As usual for ICP, there is a high variance in the run-times. The minimum is about 1.2 seconds whereas the maximum for a registration is about 47 seconds.

4.3 Plane-based registration

A second registration method tested on the data is the plane-based registration, which was recently co-developed by one of the authors of this article (Pathak et al. 2010). The main idea of plane-based registration is to also exploit information about "global" structures in 3D range data, namely large pla-

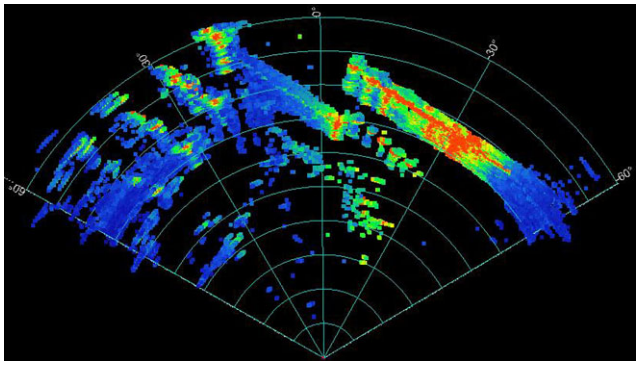
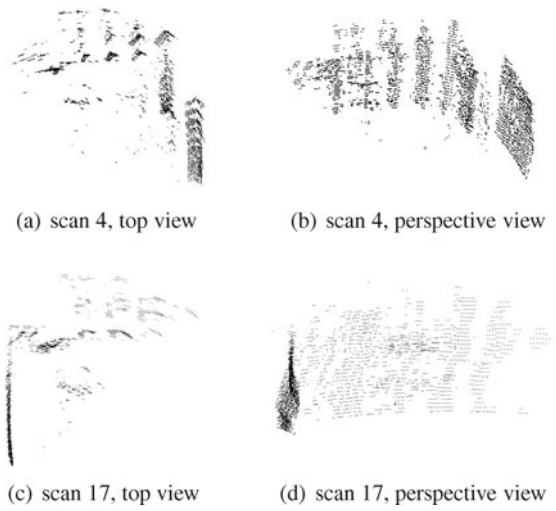


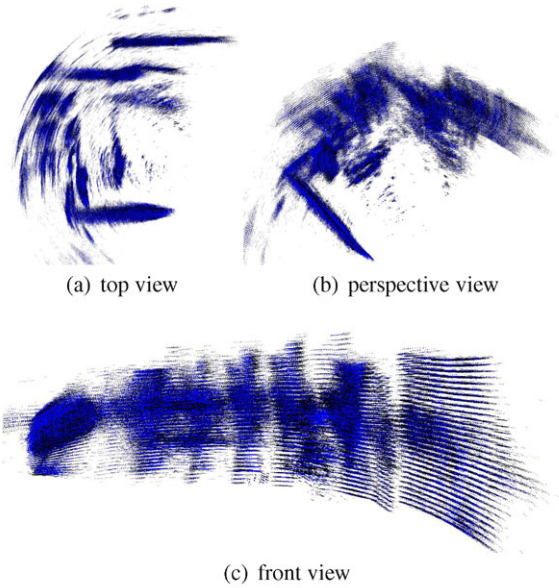
Fig. 23 A scan of the Tritech Eclipse as shown in its user interface. The colors indicate the intensity of the received echo



(a) scan 4, top view (b) scan 4, perspective view

(c) scan 17, top view (d) scan 17, perspective view

Fig. 24 Two examples of the sonar scans as point clouds. As can be seen, the data is quite noisy



(a) top view (b) perspective view

(c) front view

(d) overlay with Google earth

Fig. 25 The 3D map generated by ICP. The scans are not well registered as can be seen due to the high amount of dispersed points and several clearly misaligned wall structures

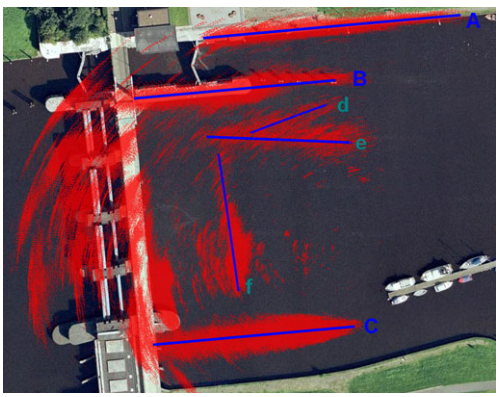


Fig. 26 Several of the deficiencies in the ICP-based 3D-map marked for easier reference; please refer to the main text for a discussion

nar patches. The method is based on a fast fitting of planes into the 3D range data and the extraction of the proper related uncertainties (Pathak et al. 2010).

Given two plane sets from two scans, the registration method then maximizes the overall geometric consistency within a search-space to determine correspondences between planes. The search-space is pruned using criteria such as overlap, and size-similarity. In doing so, only the plane parameter covariance matrix is employed, i.e., there is no need to use any point-cloud information. An important aspect of this registration method is that the covariance matrix of the estimated spatial transformations between scans is computed, which identifies the principal uncertainty directions. This information is required for using the registration in Simultaneous Localization and Mapping (SLAM), as shown in Pathak et al. (2010) where the method is used to map response scenario with a land robot.

Figure 27 shows two views of the 3D map generated by registering the 18 scans with the plane-based approach. As shown in Fig. 28, the map has a good correspondence with

Table 2 The run-times of the ICP registrations

Scan pair	Time (sec)
0 → 1	47.36
1 → 2	43.44
2 → 3	28.17
3 → 4	1.95
4 → 5	37.89
5 → 6	12.00
6 → 7	26.27
7 → 8	2.21
8 → 9	15.34
9 → 10	17.75
10 → 11	16.99
11 → 12	15.70
12 → 13	1.27
13 → 14	26.22
14 → 15	2.56
15 → 16	41.31
16 → 17	26.45
Average	21.403

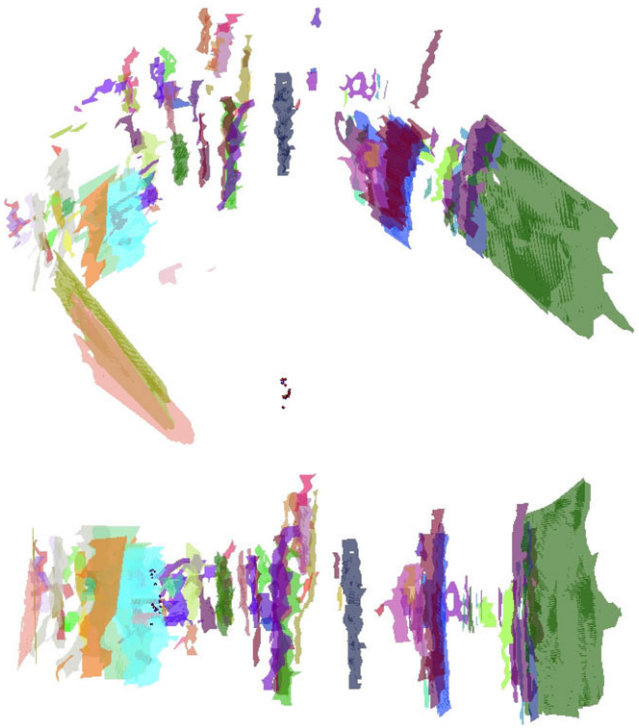


Fig. 27 Perspective views of the 3D map generated by plane-based registration from the 18 registered scans. A comparison with ground truth is shown in Fig. 28

the ground truth structures, though the three central gates are only sparsely covered—see also the area marked with *A* in

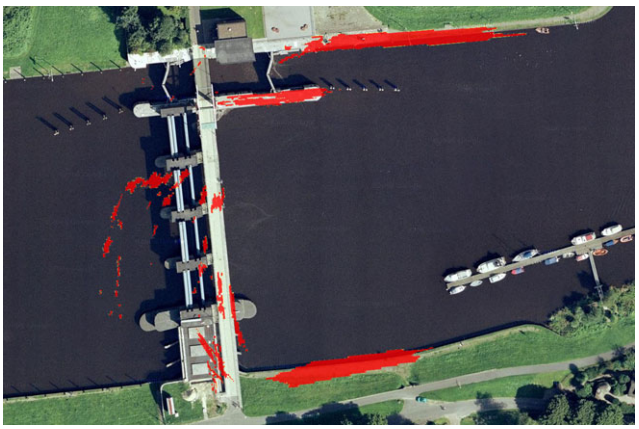


Fig. 28 An overlay of a topview of the plane-based 3D map of the Sperrwerk with an image from Google maps. It can be seen that the map captures the real structure quite well. Only the three gate structures in the center are only sparsely covered

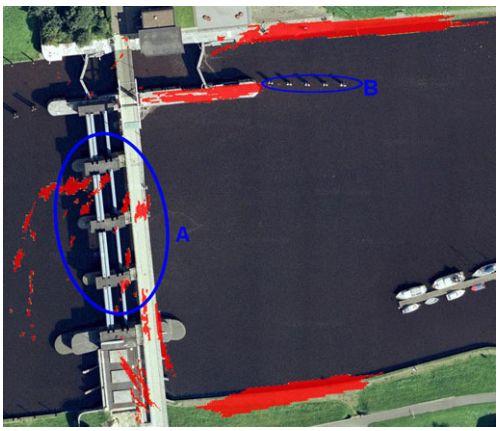


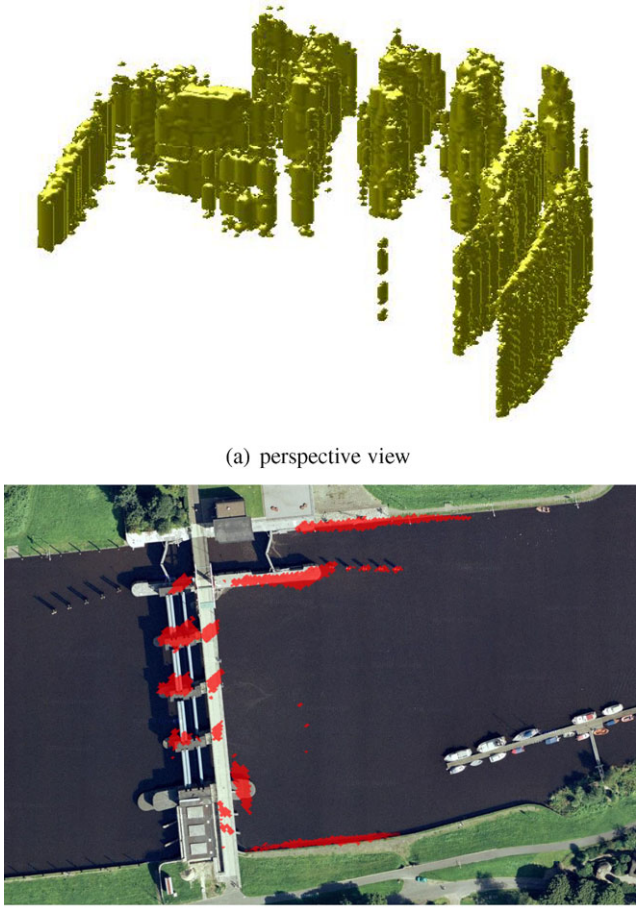
Fig. 29 Two environment parts that are not or only sparsely covered in the 3D map generated by planed-based registration

Fig. 29. The representation based on large surface patches can be very interesting for robotics applications (Birk et al. 2010); it leads among others to a high compression of the map data while maintaining a high accuracy (Vaskevicius et al. 2010). But using only large planar patches for representation has also its limits, e.g., the poles in the water in front of the gate—area *B* in Fig. 29—are not included in the 3D map.

The plane extraction takes about 0.9 to 1.4 seconds and the polygonization of the planes—e.g. for visualization or path-planning—takes 0.87 to 1.5 seconds. The actual registration, i.e., the plane matching takes 8 to 56 seconds with an average of 31 seconds on a standard PC with a AMD Turion 64 X2 procesor and 1 Gb of RAM.

4.4 Spectral registration

Finally, there is the 3D map generated with our spectral registration algorithm, which is shown in Fig. 30. The re-



(a) perspective view

(b) overlay with Google earth

Fig. 30 The 3D map generated by spectral registration

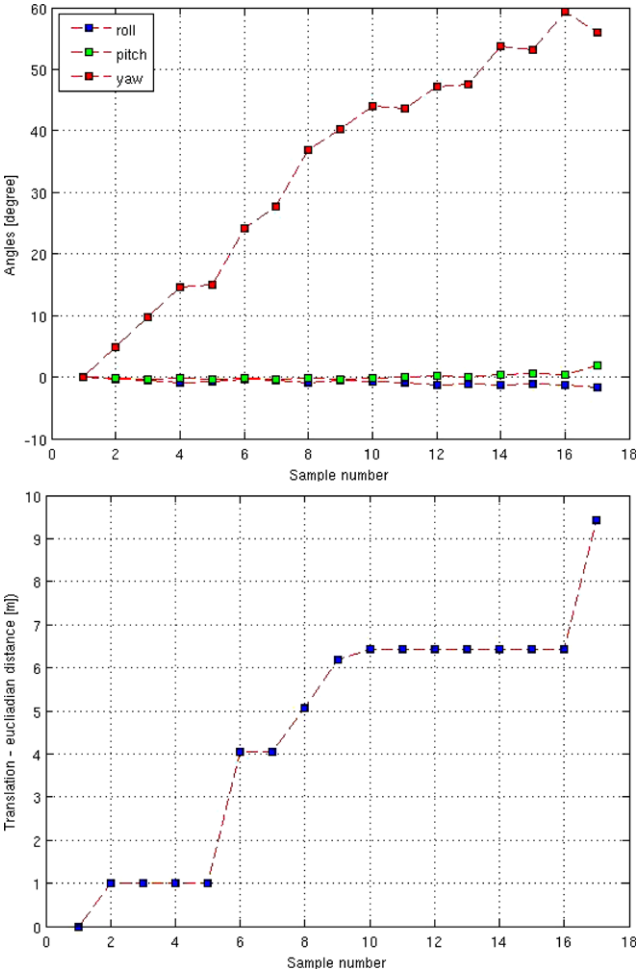


Fig. 31 The motion estimates from the spectral registration

lated motion estimates are shown in Fig. 31. The underwater structure is well represented. The map has a very good correspondence with ground truth, which is at least as good as in the case of the plane-based registration and clearly better than ICP. The plane-based representation has the advantage that it is more memory efficient than the point clouds or grid representations, but it also depends to some extent on the presence of planar structures in the environment. This obviously does not hold for the spectral registration presented here.

The computation time using a $256 \times 256 \times 256$-grid resolution is always exactly 60.6 seconds. When using a $128 \times 128 \times 128$-grid resolution, which leads to a stronger discretization of the map while providing similar qualitative results for both the Lesum 3D map as well as the simulation data reported in Sect. 3, the computation time is only 10.5 seconds. Please note that spectral registration is currently implemented in MATLAB, whereas the plane-based registration and ICP are implemented in C/C++.

Furthermore, it has to be noted that the computation time for the spectral registration is fixed whereas ICP and plane-based registration have a high variance of their run-times. So, the runtime of the spectral registration is absolutely deterministic and a priori known. This is for example of interest for online mapping where a clearly determined amount of computation time can be allocated between scans to generate the 3D map.

5 Conclusions and future work

A registration method that is well suited to generate 3D maps from noisy sonar data was presented. This 3D registration is a spectral method based on Phase Only Matched Filtering (POMF) on non-trivially resampled spectra. The approach can cope with significant 6 degree of freedom transformations between scans, i.e., large translations—leading to small overlap between scans—and substantial rotations—namely changes in yaw of up to $\pm 90°$ combined with changes in both roll and pitch of up to $\pm 35°$. The registration is at the same time very robust against noise on the 3D scans.

The 3D registration method is evaluated in different sets of experiments. First, simulated data is used, which has the

advantage that the type and the amount of noise can be controlled and that the ground truth transformations between scans are exactly known. It is shown that the method can deal with large spatial distances between scans and that it is very robust against noise. The spectral 3D registration method is then evaluated in a second set of experiments with real world data. Concretely, 18 scans of the Lesumer Sperrwerk—a large structure in form of a flood gate and a lock in the river Lesum in Bremen—with a Tritech Eclipse sonar are used. The spectral registration method is compared to two other methods suited for noisy 3D registrations, namely Iterative Closest Point (ICP) and plane-based registration. The spectral registration method performs very well in terms of the resulting map as well as its run-times.

The 3D registration method presented in this article can cope with noisy data and large spatial transformations, i.e., small overlap, between scans. Though being quite precise, there are always at least the effects of the discretization used in our algorithm. The method is hence suited to generate 3D maps, but there are obvious limits as the relative localization errors accumulate. A possible remedy is the use of SLAM. This requires the computation of uncertainty estimations in the registrations. The possibility to extend our previous work on uncertainty estimation for 2D spectral registration (Pfingsthorn et al. 2010) to 3D is left for future work.

Acknowledgements The authors thank Narunas Vaskevicius for running the ICP registrations as a comparison basis.

The research leading to the results presented here has received funding from the European Community's Seventh Framework Programme (EU FP7) under grant agreement n. 231378 "Cooperative Cognitive Control for Autonomous Underwater Vehicles (Co3-AUVs)", http://www.Co3-AUVs.eu.

References

Besl, P. J., & McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *14*(2), 239–256.

Birk, A., Pathak, K., Vaskevicius, N., Pfingsthorn, M., Poppinga, J., & Schwertfeger, S. (2010). Surface representations for 3D mapping: a case for a paradigm shift. KI—German Journal on *Artificial Intelligence*.

Buelow, H., Birk, A., & Unnithan, V. (2009). Online generation of an underwater photo map with improved Fourier Mellin based registration. In *International OCEANS Conference*. New York: IEEE Press.

Buelow, H., Pfingsthorn, M., & Birk, A. (2010). Using robust spectral registration for scan matching of sonar range data. In *7th symposium on intelligent autonomous vehicles (IAV), IFAC*.

Castellani, U., Fusiello, A., Murino, V., Papaleo, L., Puppo, E., Repetto, S., & Pittore, M. (2004). Efficient on-line mosaicing from 3D acoustical images. In *OCEANS '04* (Vol. 2, pp. 670–677).

Castellani, U., Fusiello, A., & Murino, V. (2002). Registration of multiple acoustic range views for underwater scene reconstruction. *Computer Vision and Image Understanding*, *87*, 78–89.

Chen, Q., Defrise, M., & Deconinck, F. (1994). Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *16*, 1156–1168.

Clark, D., Bell, J., de Saint-Pern, Y., & Petillot, Y. (2005). PHD filter multi-target tracking in 3D sonar. In *Oceans 2005* (pp. 265–270).

Davison, J., & Kita, N. (2001) 3D simultaneous localisation and map-building using active vision for a robot moving on undulating terrain. In *IEEE conference on computer vision and pattern recognition*, Hawaii, Dec 8–14, 2001.

Driscoll, J., & Healy, D. (1994). Computing Fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics*, *15*, 202–250.

Ende, B. A. A. (2001). 3D mapping of underwater caves. *IEEE Computer Graphics and Applications*, *21*, 14–20.

Fairfield, N., Kantor, G. A., & Wettergreen, D. (2007). Real-time SLAM with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics*, *24*(1–2), 3–21.

Hähnel, D., Burgard, W., & Thrun, S. (2003). Learning compact 3D models of indoor and outdoor environments with a mobile robot. *Robotics and Autonomous Systems*, *44*(1), 15–27.

Hernandez, E., Ridao, P., Ribas, D., & Mallios, A. (2009). Probabilistic sonar scan matching for an AUV. In *Intelligent robots and systems, 2009. IROS 2009. IEEE/RSJ international conference on* (pp. 255–260).

Horn, B. (1984). Extended Gaussian images. *Proceedings of the IEEE*, 1671–1686.

Horner, J. L., & Gianino, P. D. (1984). Phase-only matched filtering. *Applied Optics*, *23*, 812–816.

Howard, A., Wolf, D. F., & Sukhatme, G. S. (2004). Towards 3D mapping in large urban environments. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Sendai, Japan.

Keller, Y., Shkolnisky, Y., & Averbuch, A. (2005). Algebraically accurate volume registration using Euler's theorem and the 3d pseudo-polar fft. In *Proc. IEEE conf. comput. vision pattern recognit*.

Keller, Y., Shkolnisky, Y., & Averbuch, A. (2006). Volume registration using the 3-d pseudopolar Fourier transform. *IEEE Transactions on Signal Processing*, *54*(11), 4323–4331.

Kostelec, P., & Rockmore, D. (2003). *Ffts on the rotation group*. Working Papers Series, Santa Fe Institute.

Lee, T. S., Choi, J. S., Lee, J. H., & Lee, B. H. (2009). 3-D terrain covering and map building algorithm for an AUV. In *Proceedings of the 2009 IEEE/RSJ international conference on intelligent robots and systems* (pp. 4420–4425). St. Louis: IEEE Press.

Liu, Y., Emery, R., Chakrabarti, D., Burgard, W., & Thrun, S. (2001). Using EM to learn 3D models of indoor environments with mobile robots. In *18th conf. on machine learning*, Williams College, 2001.

Lorenson, A., & Kraus, D. (2009). 3D-Sonar image formation and shape recognition techniques. In *OCEANS 2009* (pp. 1–6).

Lucchese, L., Doretto, G., & Cortelazzo, G. (2002). A frequency domain technique for range data registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 1468–1484.

Madjidi, H., & Nagahdaripour, S. (2003). 3-D photo-mosaicking of benthic environments. In S. Nagahdaripour (Ed.), *OCEANS 2003. Proceedings* (Vol. 4, pp. 2317–2318).

Majumder, S., Rosenblatt, J., Scheding, S., & Durrant-Whyte, H. (2001). Map building and localization for underwater navigation. In *Lecture notes in control and information sciences: Vol. 271. Experimental robotics VII* (pp. 511–520). Berlin: Springer.

Makadia, A., & Daniilidis, K. (2003). Direct 3d-rotation estimation from spherical images via a generalized shift theorem. In *Computer vision and pattern recognition, Proceedings. IEEE computer society conference on* (Vol. 2, pp. 217–224).

Makadia, A., Patterson, A., & Daniilidis, K. (2006). Fully automatic registration of 3D point clouds. In *IEEE comp. soc. conf. on computer vision and pattern recognition (CVPR)* (Vol. 1, pp. 1297–1304).

Makadia, A., Sorgi, L., & Daniilidis, K. (2004). Rotation estimation from spherical images. In *Proc. int. conf. on pattern recognition*, Cambridge.

Mallios, A., Ridao, P., Hernandez, E., Ribas, D., Maurelli, F., & Petillot, Y. (2009). Pose-based SLAM with probabilistic scan matching algorithm using a mechanical scanned imaging sonar. In *OCEANS 2009* (pp. 1–6).

Murino, V., Fusiello, A., Iuretigh, N., & Puppo, E. (2000). 3D mosaicing for environment reconstruction. In *Pattern recognition, 2000. Proceedings. 15th international conference on* (Vol. 3, pp. 358–362).

Murino, V., Palmese, M., & Trucco, A. (2007). Processing tools for acoustic 3D images. In P. Blondel & A. Caiti (Eds.), *Springer praxis books: Part II. Buried waste in the seabead—acoustic imaging and bio-toxicity* (pp. 39–64). Berlin: Springer.

Murino, V., & Trucco, A. (2000). Three-dimensional image generation and processing in underwater acoustic vision. *Proceedings of the IEEE*, *88*(12), 1903–1948.

Negahdaripour, S., & Madjidi, H. (2003). Stereovision imaging on submersible platforms for 3-D mapping of benthic habitats and seafloor structures. *Oceanic Engineering, IEEE Journal of*, *28*(4), 625–650.

Newman, P., & Durrant-Whyte, H. (1998). Using sonar in terrain-aided underwater navigation. In H. Durrant-Whyte (Ed.), *Robotics and automation, 1998. Proceedings. 1998 IEEE international conference on* (Vol. 1, pp. 440–445).

Nicosevici, T., Gracias, N., Negahdaripour, S., & Garcia, R. (2009). Efficient three-dimensional scene modeling and mosaicing. *Journal of Field Robotics*, *26*(10), 759–788.

Nuechter, A., Lingemann, K., Hertzberg, J., & Surmann, H. (2006). 6D SLAM—Mapping outdoor environments. In *IEEE international workshop on safety, security, and rescue robotics (SSRR)*. New York: IEEE Press.

Oppenheim, A. V., & Schafer, R. W. (1989). *Discrete-time signal processing*. *Prentice Hall signal processing series*. Prentice Hall: Englewood Cliffs.

Oskard, D., Hong, T. H., & Shaffer, C. (1990). Real-time algorithms and data structures for underwater mapping. *Systems, Man and Cybernetics, IEEE Transactions on*, *20*(6), 1469–1475.

Palmese, M., & Trucco, A. (2008). From 3-D sonar images to augmented reality models for objects buried on the seafloor. *Instrumentation and Measurement, IEEE Transactions on*, *57*(4), 820–828.

Pathak, K., Birk, A., Vaskevicius, N., Pfingsthorn, M., Schwertfeger, S., & Poppinga, J. (2010). Online 3D SLAM by registration of large planar surface segments and closed form pose-graph relaxation. *Journal of Field Robotics, Special Issue on 3D Mapping*, *27*(1), 52–84.

Pathak, K., Birk, A., Vaskevicius, N., & Poppinga, J. (2010). Fast registration based on noisy planes with unknown correspondences for 3D mapping. *IEEE Transactions on Robotics*, *26*(3), 424–441.

Pathak, K., Vaskevicius, N., & Birk, A. (2010). Uncertainty analysis for optimum plane extraction from noisy 3D range-sensor pointclouds. *Intelligent Service Robotics*, *3*, 37–48.

Pfingsthorn, M., Schwertfeger, S., Buelow, H., & Birk, A. (2010). Maximum likelihood mapping with spectral image registration. In *IEEE international conference on robotics and automation (ICRA)*. New York: IEEE Press.

Pizarro, O., Eustice, R., & Singh, H. (2004). Large area 3D reconstructions from underwater surveys. In *OCEANS, MTS/IEEE conference and exhibition*, Kobe, Japan (pp. 678–687).

Ribas, D., Ridao, P., Domingo Tardos, J., & Neira, J. (2007). Underwater SLAM in a marina environment. In P. Ridao (Ed.), *Intelligent robots and systems, 2007. IROS 2007. IEEE/RSJ International Conference on* (pp. 1455–1460).

Ribas, D., Ridao, P., Neira, J., & Tardos, J. D. (2006). SLAM using an imaging sonar for partially structured underwater environments. In P. Ridao (Ed.), *Intelligent robots and systems, 2006 IEEE/RSJ international conference on* (pp. 5040–5045).

Roman, C., & Singh, H. (2006). Consistency based error evaluation for deep sea bathymetric mapping with robotic vehicles. In *Robotics and automation, 2006. ICRA 2006. Proceedings 2006 IEEE international conference on* (pp. 3568–3574).

Roman, C., & Singh, H. (2007). A self-consistent bathymetric mapping algorithm. *Journal of Field Robotics*, *24*(1–2), 23–50.

Saez, J., Hogue, A., Escolano, F., & Jenkin, M. (2006). Underwater 3D SLAM through entropy minimization. In *Robotics and automation, 2006. ICRA 2006. Proceedings 2006 IEEE international conference on* (pp. 3562–3567).

Salvi, J., Petillot, Y., Thomas, S., & Aulinas, J. (2008). Visual SLAM for underwater vehicles using video velocity log and natural landmarks. In *OCEANS 2008* (pp. 1–6).

Santilli, K., Bemis, K., Silver, D., Dastur, J., & Rona, P. (2004). Generating realistic images from hydrothermal plume data. In *Visualization, 2004* (pp. 91–98). New York: IEEE.

Sedlazeck, A., Koeser, K., & Koch, R. (2009). 3D reconstruction based on underwater video from ROV Kiel 6000 considering underwater imaging conditions. In *IEEE OCEANS Conference '09*, Bremen, Germany.

Surmann, H., Nuechter, A., & Hertzberg, J. (2003). An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, *45*(3–4), 181–198.

Thrun, S., Haehnel, D., Ferguson, D., Montemerlo, M., Triebel, R., Burgard, W., Baker, C., Omohundro, Z., Thayer, S., & Whittaker, W. (2003). A system for volumetric robotic mapping of abandoned mines. In *Proc. IEEE international conference on robotics and automation (ICRA)*, Taipei, Taiwan.

Vaskevicius, N., Birk, A., Pathak, K., Schwertfeger, S., & Rathnam, R. (2010). Efficient representation in 3D environment modeling for planetary robotic exploration. *Advanced Robotics*, *24*(8–9).

Weingarten, J. (2006). *Feature-based 3D SLAM*. PhD thesis, EPFL, Lausanne, Switzerland.

Williams, S., & Mahon, I. (2004). Simultaneous localisation and mapping on the Great Barrier Reef. In I. Mahon (Ed.), *Robotics and automation, 2004. Proceedings. ICRA '04. 2004 IEEE international conference on* (Vol. 2, pp. 1771–1776).

Wulf, O., Brenneke, C., & Wagner, B. (2004). Colored 2D maps for robot navigation with 3D sensor data. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (Vol. 3, pp. 2991–2996). New York: IEEE Press.

Wulf, O., & Wagner, B. (2003). Fast 3D-scanning methods for laser measurement systems. In *International conference on control systems and computer science (CSCS14)*.

Yu, S. C., Kim, T. W., Marani, G., & Choi, S. (2007). Real-time 3D sonar image recognition for underwater vehicles. In *Underwater technology and workshop on scientific use of submarine cables and related technologies, 2007. Symposium on* (pp. 142–146).

Heiko Bülow received his diploma (FH) from the University of Applied Scienes in Braunschweig/Wolfenbuettel and the M.Phil degree from the University of Glamorgan. Currently he is pursuing the Ph.D. at the Jacobs University in Bremen. He is working at the Jacobs Robotics Lab, developing algorithms for rescue and other autonomous robots. His research interests include multidimensional signal processing, pattern recognition and computer vision.

Andreas Birk is an associate professor in Electrical Engineering and Computer Science (EECS) at Jacobs University Bremen. He started at Jacobs University in Fall 2001, building up the robotics group at this newly founded private university. Before he joined Jacobs, he held a research-mandate from the Flemish Society for Applied Research (IWT) and was a visiting Professor at the Vrije Universiteit Brussel (VUB) where he did his research in the Artificial Intelligence Lab (AI-lab). In 1995 he received his doctorate from the Universität des Saarlandes, Saarbrücken. Andreas Birk's research focuses on autonomous systems. On the engineering side, he is working on the design and construction of complete systems. This includes the design and construction of embedded hardware and mechatronics as well as software development up to full autonomy. On the basic research side, he is interested in a constructive understanding of intelligence.